Model Clone Detection in Practice

Florian Deissenboeck, Benjamin Hummel
Elmar Juergens, Michael Pfaehler

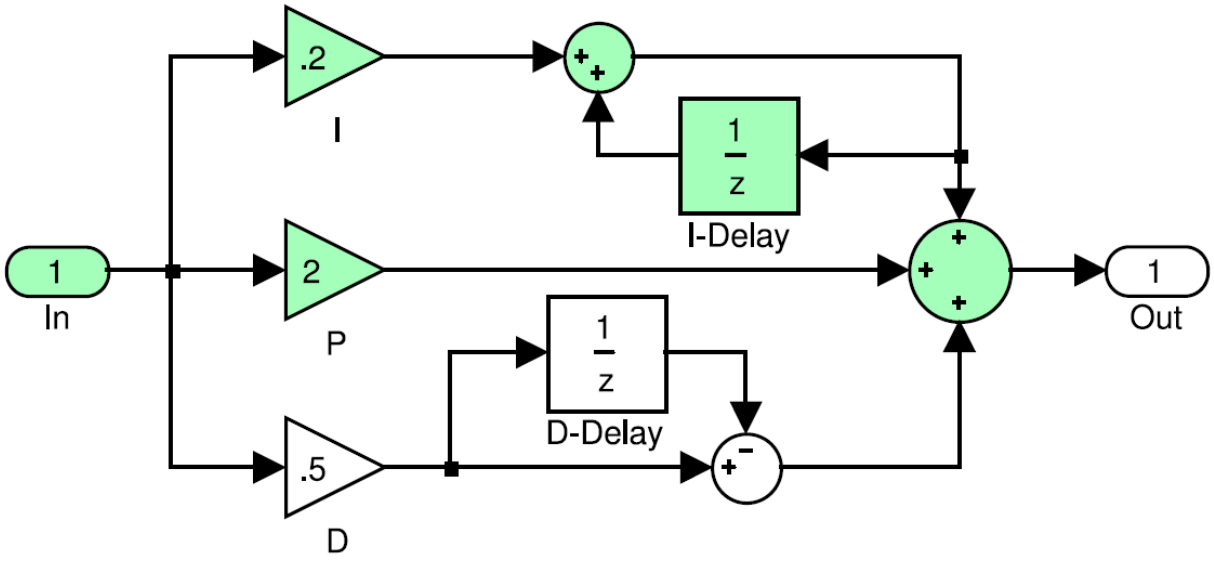
Technische Universität München

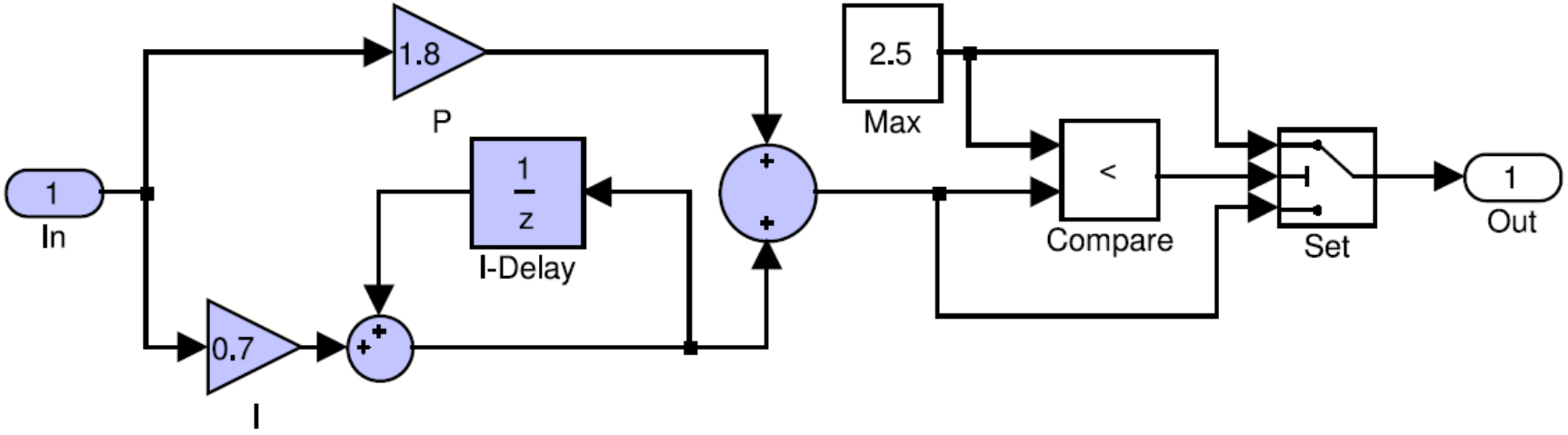
Bernhard Schaetz

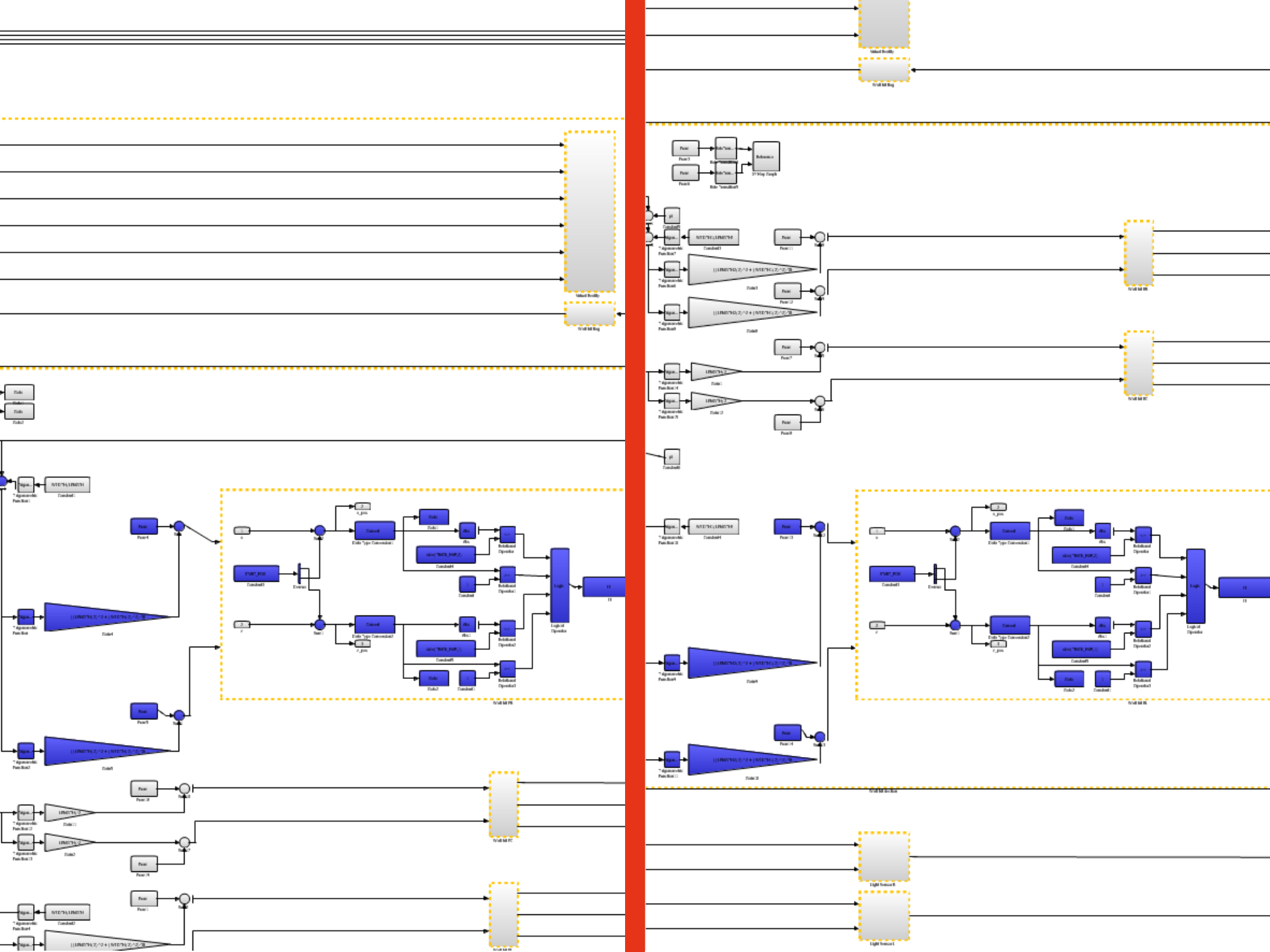
fortiss gGmbH

```
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
   boolean found = false;
   StringBuffer res = new StringBuffer();
   if (elements != null) {
      Index.getInstance().setCurrentRenderer(
            FlatReferenceRenderer.getInstance());
      for (int i = 0; i < elements.length; i++) {
         ModelElement el = elements[i];
         res.append(showElementLink(el)).append(HTML.LINE_BREAK);
         found = true;
      }
      Index.getInstance().resetCurrentRenderer();
   }
   if (!found && nomsg != null && nomsg.length() > 0) {
      res.append(HTML.italics(nomsg));
   }
   return res.toString();
}
```

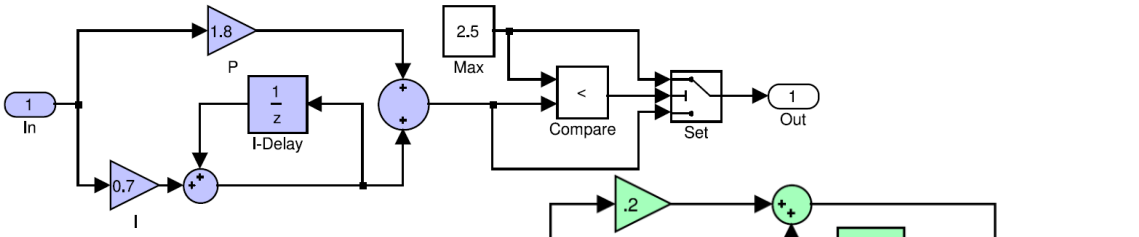

```
// Utilities for arrays of elements
public String showElements(ModelElement[] elements, String nomsg) {
   boolean found = false;
   StringBuffer res = new StringBuffer();
   if (elements != null) {
      Index.getInstance().setCurrentRenderer(
            FlatReferenceRenderer.getInstance());
      for (int i = 0; i < elements.length; i++) {
         ModelElement el = elements[i];
         res.append(showElementLink(el)).append(HTML.LINE_BREAK);
         found = true;
      }
      Index.getInstance().resetCurrentRenderer();
   }
   if (!found && nomsg.length() > 0) {
      res.append(HTML.italics(nomsg));
   }
   return res.toString();
}
```


.2
I
1
z
I-Delay
1
In
2
P
1
Out
1
z
D-Delay
.5
D


1.8
P
2.5
Max
1
In
1
z
I-Delay
<
Compare
1
Out
Set
0.7
I

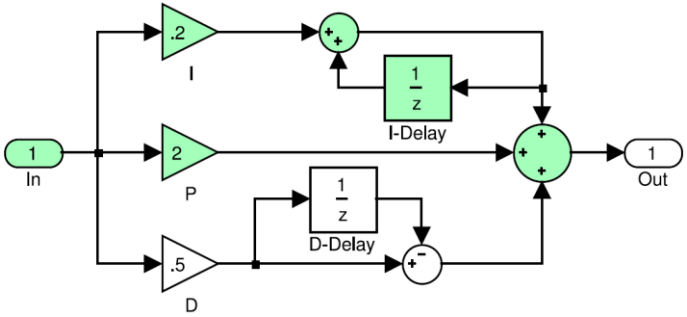


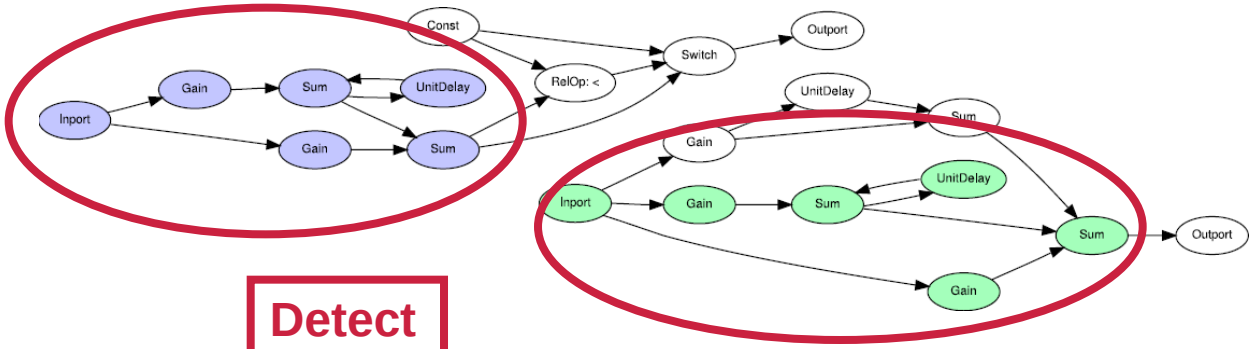
Detection Steps



Load Model

Normalize





Detect

Report

Existing Work

Deissenboeck et. al. [ICSE'08] — **First detector and MAN case study**

Pham et. al. [ICSE'09]
Nguyen et. al. [FASE'09] } **Different detection algorithm**

Huhn et. al. [MBEES'10] — **Application to SCADE models**

Challenges in Practice

Scalability

Relevance

Inspection

Scalability

Publicly available (Matlab Central)

Model	# Nodes	# Edges
SIM	428	415
MUL	475	576
SEM	1,741	2,029
ECW	2,312	2,274
MPC	369	395

“Real-World” (BMW)

Algorithmic Tricks

Cloned Subsystem Removal (CSR)

[Pham+09]

Splitting at High-Degree Nodes (HDN)

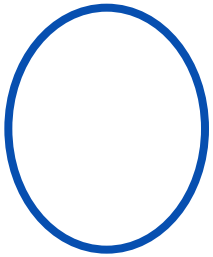
[Pham+09]

Reduced Branching Heuristic (RBH)

[DHJ+08]

Runtime Results

	CSR	HDN	RBH
eScan [Pham+09]			
eScan [Pham+09]	☑		
eScan [Pham+09]		☑	
eScan [Pham+09]	☑	☑	
ConQAT [DHJ+08]			☑
ConQAT [DHJ+08]	☑		☑



“…more than half of the clones found are obviously clones according to our definition but would not be considered relevant by a developer.”

MAN case study [DHJ+08]

Improving Relevance

Different notion of clone?
(not isomophism)

Different normalization?

Domain-specific metrics for ranking/filtering clones!

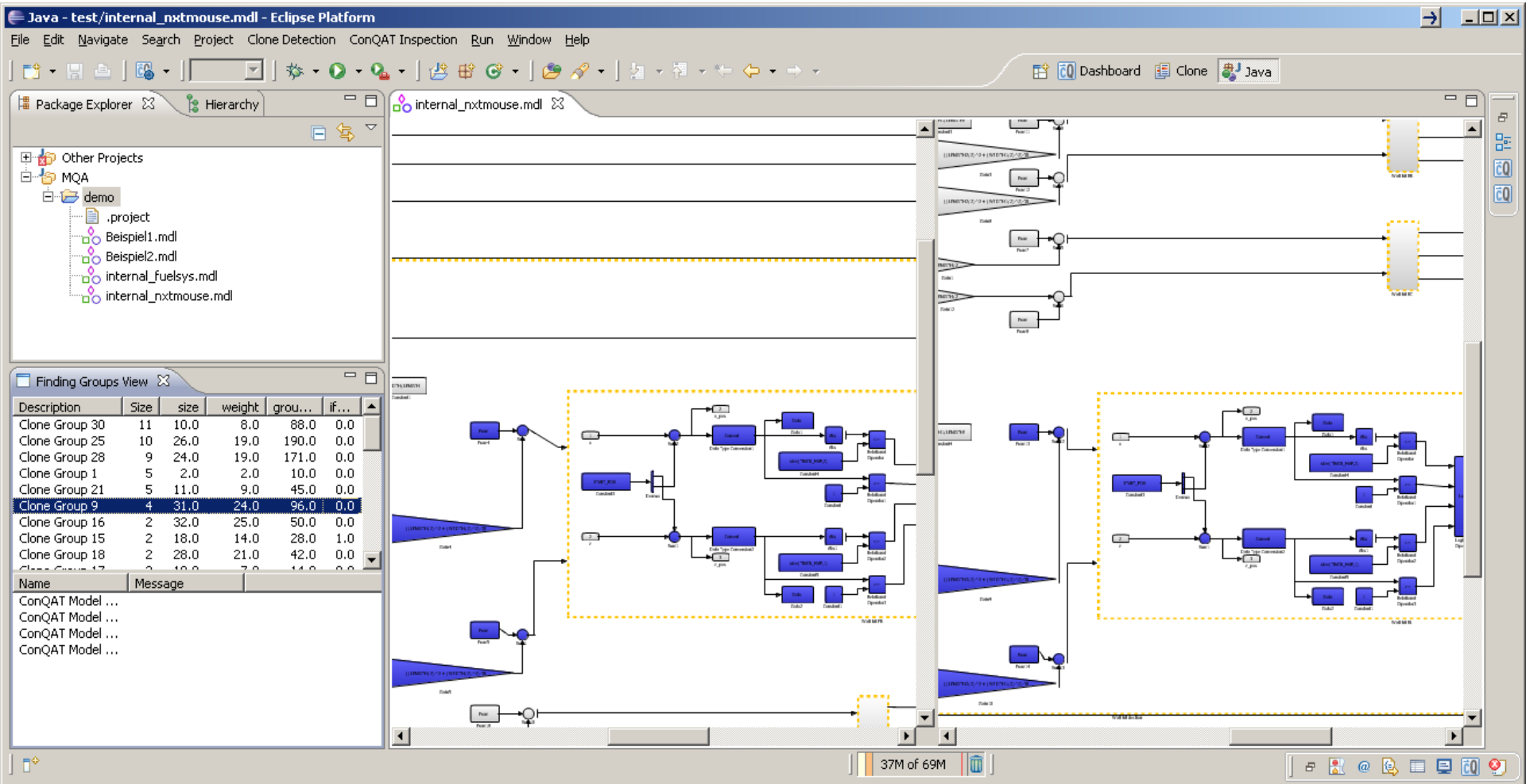
Metrics

Deviation for ranking 21 clones
(worst 220, average 146)

	Developer 1	Developer 2
Node Size	139	163
Clone Weight	157	183
Relative Weight	110	90
Interface/Weight	86	90
Interface/Node Size	123	123

Deviation between developers: 74

Inspection: Tool Support



Summary

Ingrediences for practical application:

(not only for model clones)

- **Scalability**
- **Relevance**
- **Inspection/Tooling**

Working on all three of them,
but none is „solved“ yet :-)

What should be the clones?