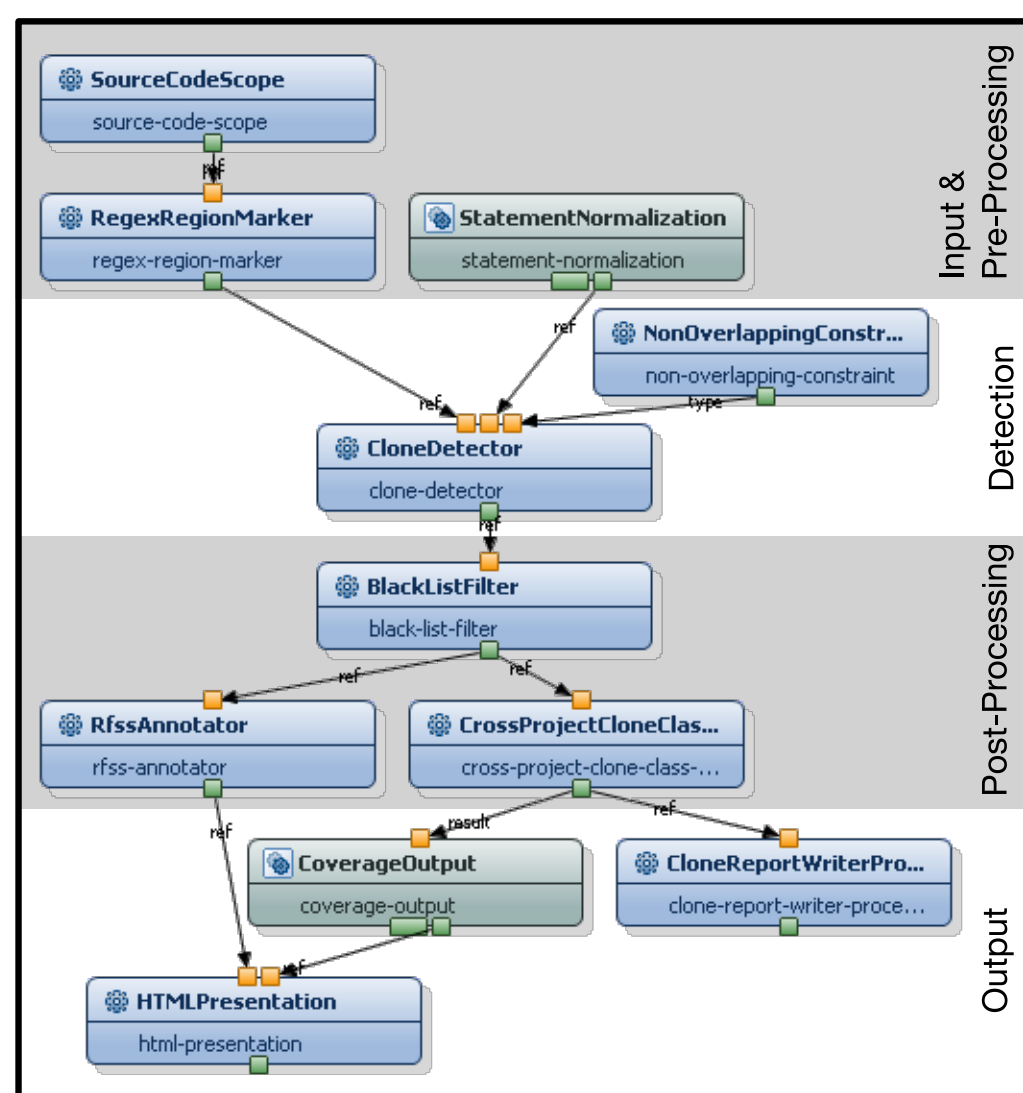


CloneDetective - A Workbench for Clone Detection Research

Elmar Juergens, Florian Deissenboeck, Benjamin Hummel

Technische Universität München
{juergens,deissenb,hummelb}@in.tum.de

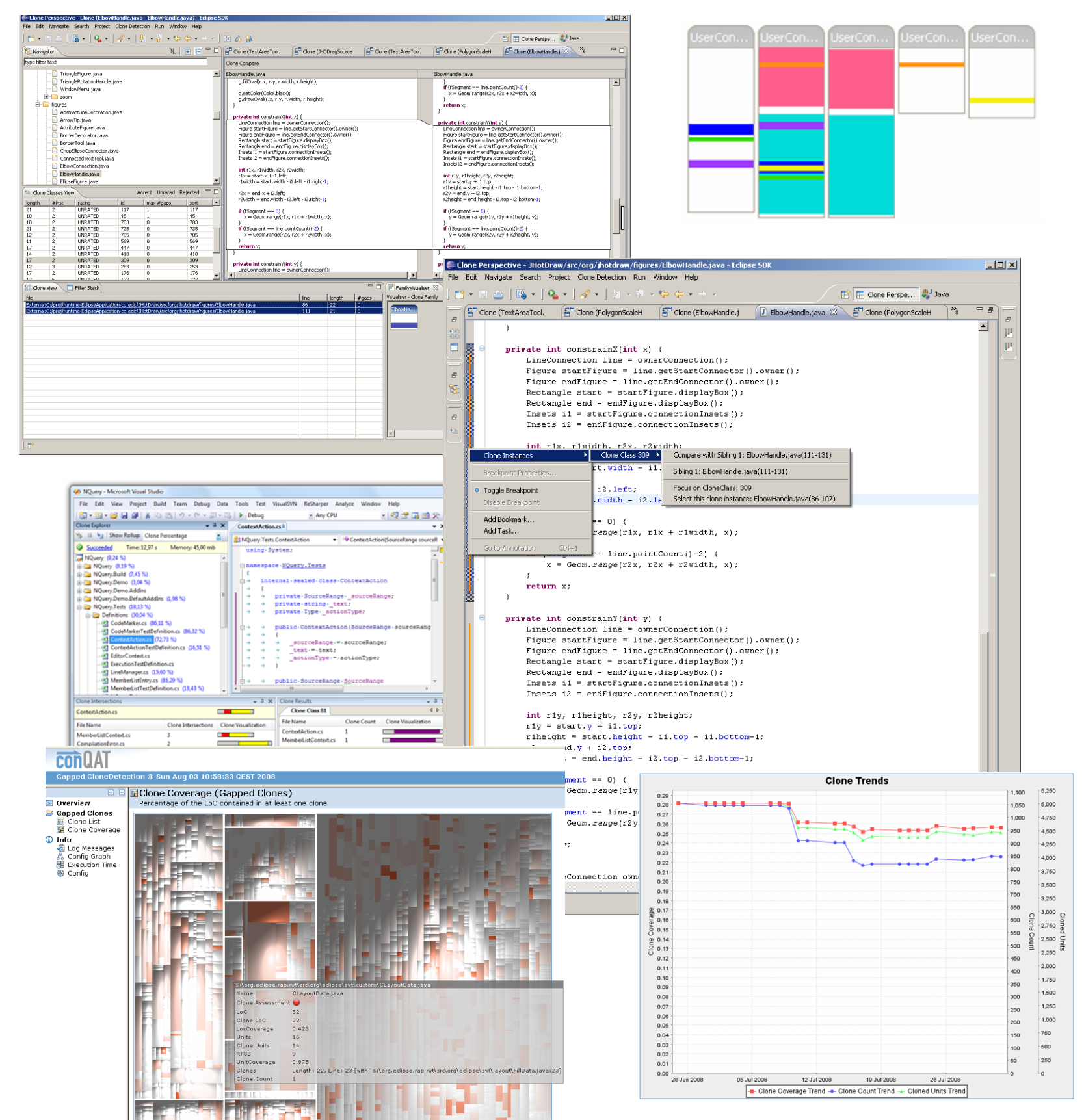
Flexible Configuration and Extension



ConQAT tool chain supports

- graphical setup of clone detection pipeline
- adding custom detection steps via extension mechanism

Tool Infrastructure



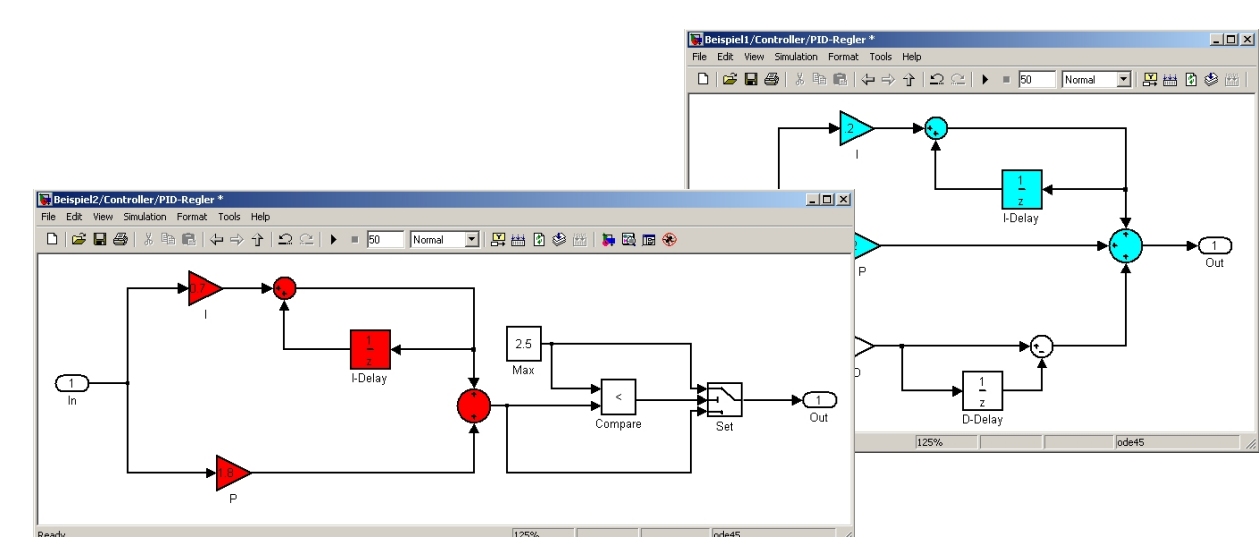
- Viewer to review clones
- Fully integrated into Eclipse and Visual Studio
- Dashboard integration

Features

```
// 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();
    }
}
```

- Scalable token-based detection
- Detection of gapped clones
- Supports ABAP, Ada, C/C++, C#, COBOL, Java, PL/I, VB
- Flexible normalization of code
- Exclusion of generated code
- Filtering and visualization of detection results

Model-based Clone Detection



- Supports detection of clones in Matlab/Simulink models
- Interface for extension towards similar data-flow models