Traces

Overture Workshop

Newcastle 2009

Peter Gorm Larsen & Kenneth Lausdahl

Agenda

- Traces
- Motivation
- Goal
- Approach
- Specification overview
- Integration in Eclipse
- Demo

Trace definition

- Named trace
  - Choice definition
    - a.Push(1) | a.Reset()
  - Sequence definition
    - a.Push(1) ; a.Reset()
  - Trace apply
    - a.Push(1)
  - Repeat pattern
    - a.Push(1) {1,4}

RepeatPattern				
a*	a+	a?	a{x}	a{x,y}
0	1	0	x	x
*	*	1	x	y

Traces

```
class Stack
end Stack

class UseStack

instance variables

  s : Stack := new Stack();

traces

  PushBeforePop :   s.Reset();
                    s.Push(6){1,4};
                    s.Pop()

end UseStack
```

Motivation

- Improve the testing in VDM
- Introduce Combinatorial Testing

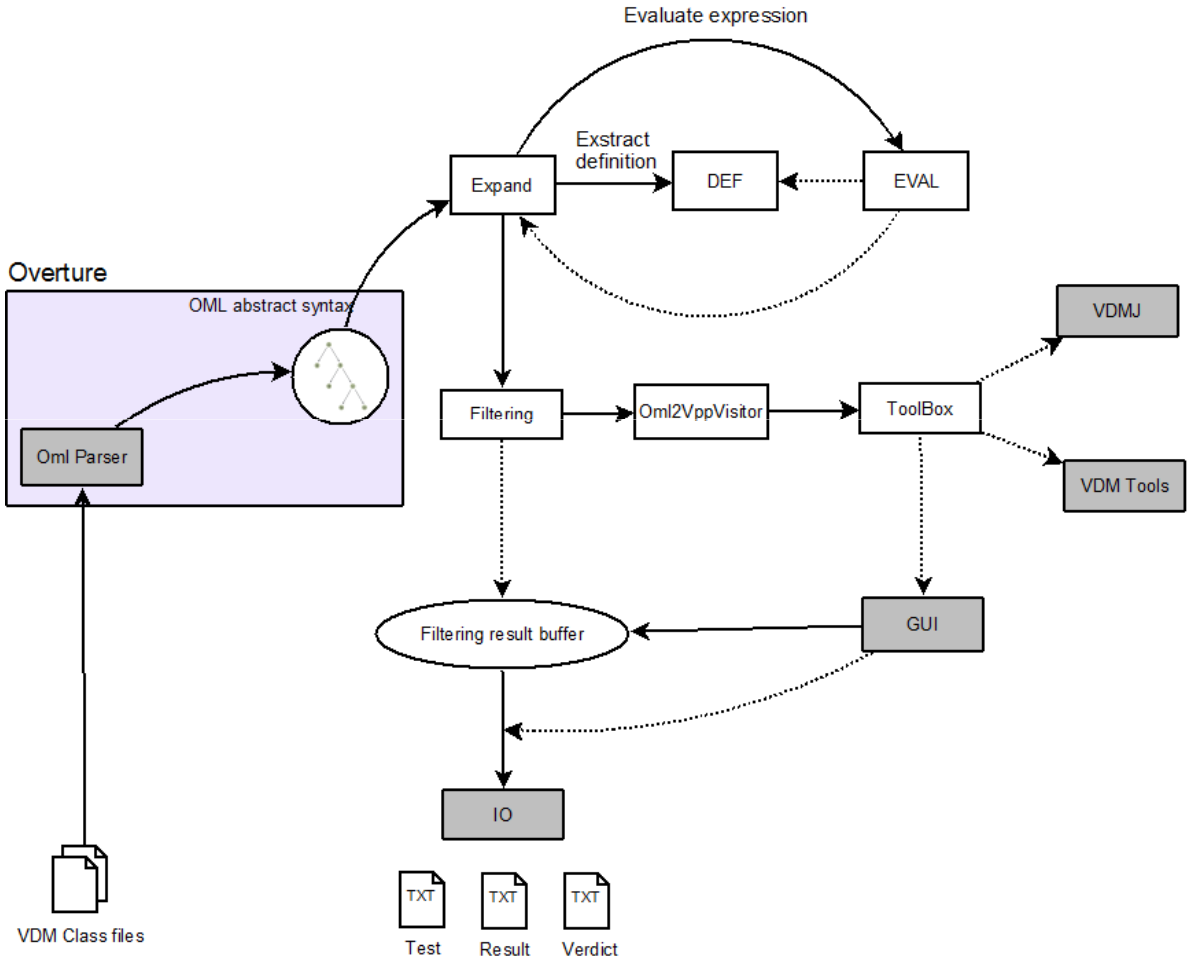
Goal

- Create a CT test tools to facilitate VDM modeling
- Minimize the computation when expanding test cases
  - Limit the amount of test cases
- Integrate to tool with Eclipse for easy use

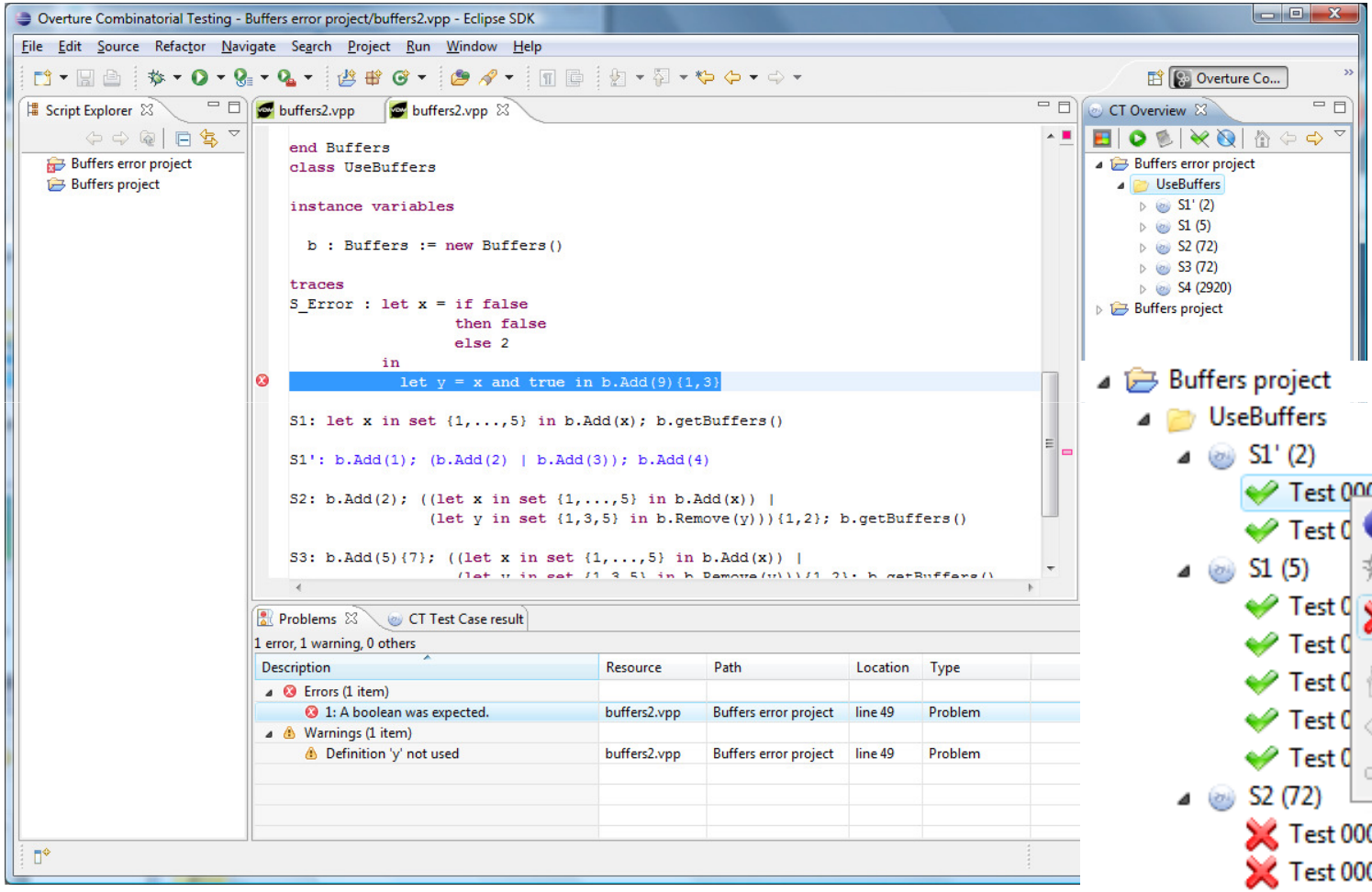
Approach

- Reuse of overture tool components
  - AST
  - Parser
  - VDMJ for evaluation
- Specification of CT in VDM++
- Code generation to Java
- Creation of Eclipse specific code
  - Views
- Combining Eclipse plug-ins and core components

Specification Overview



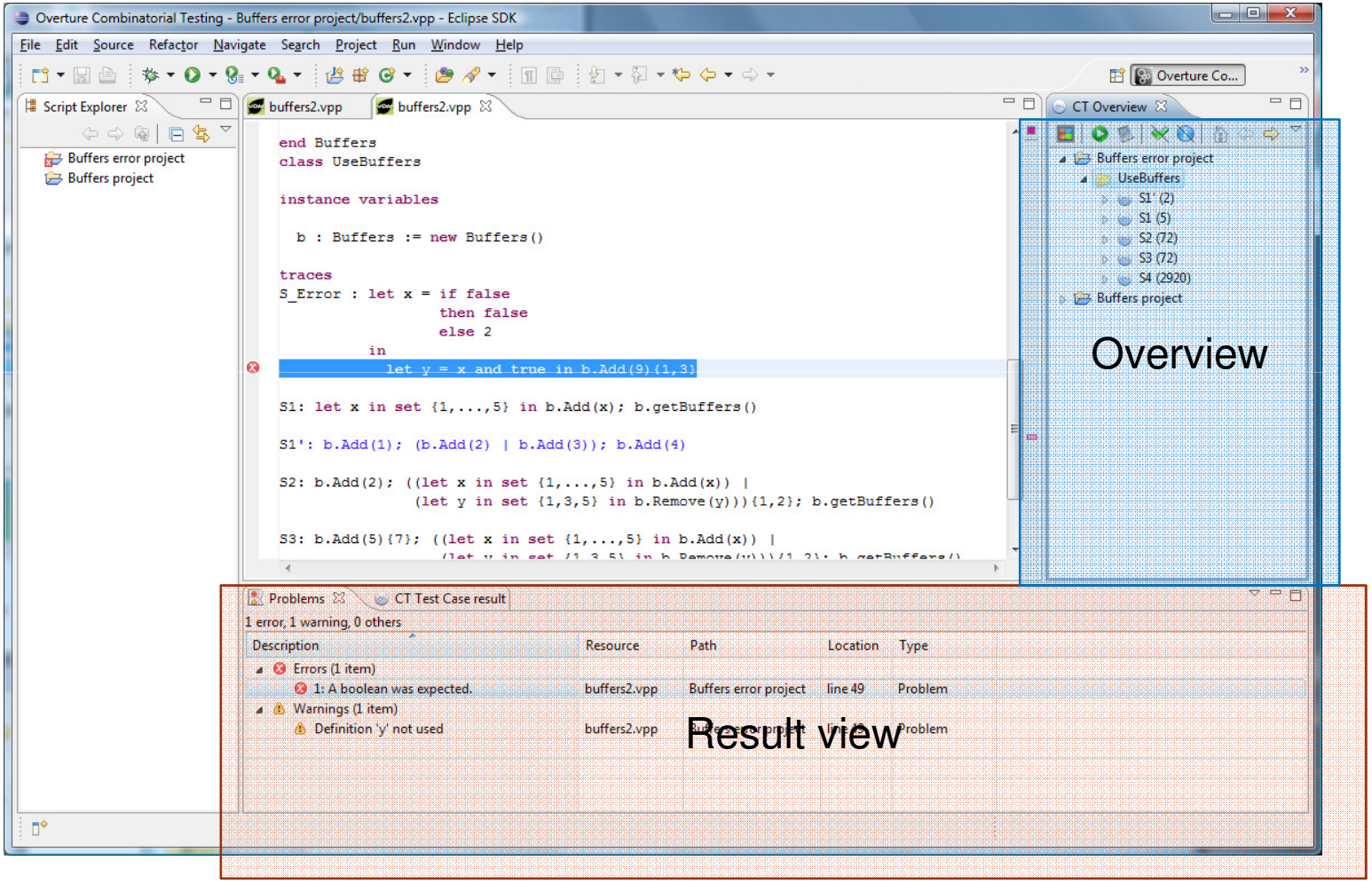
Integration in Eclipse



Eclipse plug-ins

- Two Eclipse Views have been build
  - CT Overview with projects
  - CT Results
- Integration with two external tools
  - VDMJ
  - VDM Tools
- Two Eclipse plug-in projects
  - One for wrapping the core functionality
  - One for extending the Eclipse GUI with the views

Eclipse views



Eclipse plug-in projects

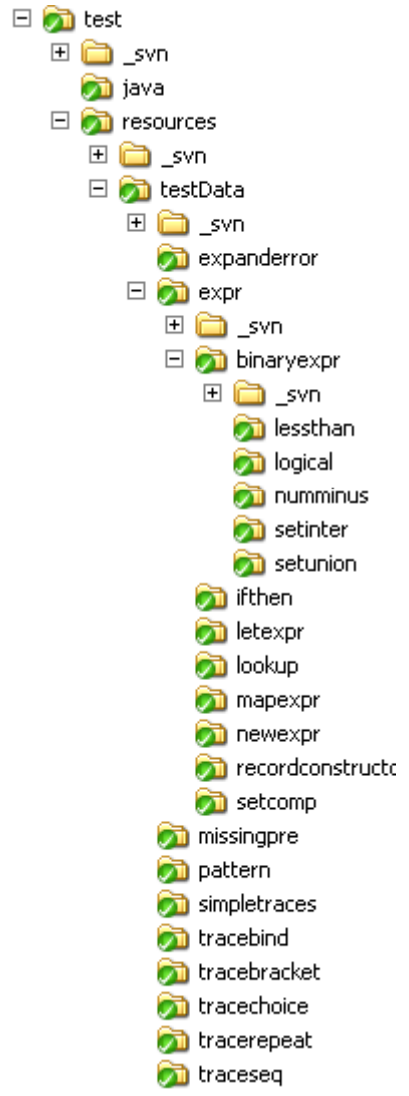
- Core component
  - org.overturetool.traces
- Eclipse plug-ins
  - org.overturetool.eclipse.plugins.traces.core
    - Wrapper for the core component
  - org.overturetool.eclipse.plugins.traces
    - Eclipse GUI

Demo

- Buffers

Test of the CT Component

- Script (for both windows and Linux)
- java -classpath "lib\StdLib-1.0.0.jar;lib\parser-1.0.0.jar;overtureUtility.jar;lib\umltrans-1.0.0.jar;lib\ast-1.0.0.jar" MainClass -vdmjEx -vdmjDir "vdmj" -project "..\..\java\CT_vdmtools_project.prj" -vdmjSpec "vdmj" -argumentFile "vdmj\vdmArgument.bat"



Test cases and expected results

```
begin Stack
…
end Stack
class UseStack

instance variables

  s : Stack := new Stack();

traces

  PushBeforePop : s.Reset() | s.Push(6)

end UseStack
```

```
{"UseStack|PushBeforePop|1" |-> ["s.Reset()"],
 "UseStack|PushBeforePop|2" |-> ["s.Push(6)"]}
```

Update site

- http://mt.lausdahl.com/CT/updatesite