TRex

An Eclipse-Based Tool for TTCN-3 Editing, Refactoring and Metrics

Paul Baker\textsuperscript{1}, Dominic Evans\textsuperscript{1}, Jens Grabowski\textsuperscript{2}, Helmut Neukirchen\textsuperscript{2}, Benjamin Zeiss\textsuperscript{2}

\textsuperscript{1}Motorola Labs, Viables Industrial Estate, Basingstoke, UK
\textsuperscript{2}Software Engineering for Distributed Systems Group, University of Göttingen
Outline

• Motivation

• Metrics

• Refactoring

• TRex Tool

• Summary / Outlook
Motivation

• Huge legacy test suites at Motorola:
  – Migration to TTCN-3
  – Automatic conversion of a UMTS test suite:
    • 60,000 lines of TTCN-3 code
    • Hard to read, use, re-use, and maintain.

• Current TTCN-3 tools:
  – Editors, Compiler, Test-Execution
  – But: No support for improving test suites!
Approach

- Assess test suites,
- Detect issues,
- Restructure test suites.

\[ \rightarrow \text{Metrics, further Analysis} \]

\[ \rightarrow \text{Refactoring} \]
Outline

• Motivation

• **Metrics**

• Refactoring

• TRex Tool

• Summary / Outlook
Metrics

“You cannot control what you cannot measure.”
De Marco: Controlling software projects.
Yourdon Press, 1982

• Software metrics:
  – Size metrics,
  – Complexity metrics,
  – Object-oriented metrics.
TTCN-3 Metrics: Template Coupling

- Measure dependency between test data and test behaviour.
  - High coupling: Changing test data requires changing test behaviour
  - Low coupling: Changing test data requires no changes to behaviour.

**Low template coupling**

1

**High template coupling**

\[
\sum_{i=1}^{n} \text{score (stmt (i ))}
\]

\[
\text{Template Coupling} := \frac{n}{n}
\]

**score (stmt (i )) :=**

1, if stmt (i ) references a template without parameters.
2, if stmt (i ) references a template with parameters.
3, if stmt (i ) uses an inline template.
Rule-Based Issue Detection

• Metrics-based:
  – \textit{Number of references to a template} = 0
    \Rightarrow \text{Remove template}
  – \textit{Number of references to a template} = 1
    \Rightarrow \text{Consider inlining the template}

• Further analysis:
  – Identical actual parameter value
    \Rightarrow \text{Inline template parameter}
  – Several templates only differ at the same element
    \Rightarrow \text{Parameterize template}
Outline

- Motivation
- Metrics
- Refactoring
- TRex Tool
- Summary / Outlook
Refactoring: Definition

„a change made to the internal structure of software to make it easier to understand and cheaper to modify without changing its observable behavior“

TTCN-3 Refactoring Catalog

- 28 refactorings from Fowler applicable to TTCN-3,
- 20 refactorings specific for TTCN-3.

Catalog structure: Refactorings for improving
  - Test behaviour (20 refactorings):
    • Extract Altstep,
    • ...
  - Overall test suite structure (22 refactorings):
    • Extract Module,
    • ...
  - Data description (6 refactorings):
    • Inline Template Parameter,
    • ...
Refactoring

- Refactoring Format:
  - Name, Summary, Motivation, Mechanic, Example

- Example (Inline Template Parameter):

```java
module ExampleModule {
  template ExampleType exampleTemplate(charstring addressParameter) := {
    ipv6 := false,
    ipAddress := addressParameter, "127.0.0.1"
  }

  testcase exampleTestcase() runs on ExampleComponent {
    pt.send(exampleTemplate("127.0.0.1"));
    pt.receive(exampleTemplate("127.0.0.1"));
  }
}
```
Outline

- Motivation
- Metrics
- Refactoring
- TRex Tool
- Summary / Outlook
TRex Features (1/2)

- TTCN-3 core notation editing
  - Syntax Highlighting, Code Completion, Outline View, Code Formatter, Text Hover, Open Declaration, Reference Finder

- Compiler Integration
  - Telelogic Tau G2/Tester

- Metrics

- Rule-Based Refactoring Suggestions

- Refactorings:
  - Rename
  - Inline Template
  - Inline Template Parameter
  - Merge Template
TRex Features (2/2)
Outline

- Motivation
- Metrics
- Refactoring
- TRex Tool
- Summary / Outlook
Summary

• Means for improving TTCN-3 test suites
  – Metrics
  – Rule-based issue detection
  – Refactoring

• TRex:
  – IDE for TTCN-3
  – Automatic assessment and restructuring
Outlook

• Current work
  – Enhanced editing functionality
  – Metrics based on control-flow and call-graphs
  – Identification of useful metrics boundary values

• Outlook
  – Implementation of more refactorings
  – Pattern-based issue detection
  – Bi-simulation to prove equivalence of test suites
TRex Availability

www.trex.informatik.uni-goettingen.de

- Open Source
- Eclipse Public License
- Available Now!
- Join the project!
• Thank you for your attention!

Please visit us at the tool exhibition!

www.trex.informatik.uni-goettingen.de