

Integration of Fokus TTCN-3 Tools into Hyades

Diana Vega, Bernard Stepien, George Din

© 2005 by «Diana Vega, Bernard Stepien»; made available under the EPL v1.0 | 28.02.2005 | Fraunhofer Fokus



Project aim

 Fokus – main contributor to standardization of TTCN-3 language and its execution environment (ETSI)

TTCN-3 Testing Tools – TTworkbench product

- TTCN-3 Core Language Editor Eclipse Plug-in
- TTCN-3 Compiler Eclipse Plug-in
- Execution environment
- Coupling with Hyades
 - Integration of the tools in Hyades project
 - Combine the high-level language capabilities of the TTCN-3 test language with the test management capabilities of Hyades



What is TTCN-3 ?

- high-level language to describe test suites formally
- powerful matching mechanism to compare an oracle to response data
- powerful high-level language to specify test logic as trees
- separation of concerns between the abstract test specification and the concrete test implementation using an adapter and coder/decoder (codec)
- powerful test data specification feature -- data templates -- that allows unlimited structuring and reusability of test data and specification of matching rules.
- message and procedure oriented.
- **Parallel test** components and configurations.
- Modularity. Code can be distributed into modules and imported.
- TTCN-3 comes in three versions: core text, graphical and tabular.



TTCN-3 Test module example

```
module PingPong {
                  type port PingPongPortType message {
                                    out charstring;
                                    in charstring:
                  }
                  type component MTCType {
                                    var charstring theWrongResponse;
                                    port PingPongPortType ping pong port;
                  }
                  type component SystemType {
                                    port PingPongPortType system_ping_pong_port;
                  }
                 testcase PlayPingPong() runs on MTCType system SystemType {
                                    map (mtc: ping pong port, system: system ping pong port);
                                    ping_pong_port.send ("ping");
                                    alt {
                                                      [] ping pong port.receive ("pong") {
                                                                        log ("received a pong");
                                                                        setverdict (pass)
                                                      [] ping pong port.receive {
                                                                       log ("did not receive a pong");
                                                                        setverdict (fail)
                                                      }
                                    };
                                    stop
                 }
                  control {
                                    execute (PlayPingPong());
                  }
```



What does the integration mean

Hyades - open source project

But it allows vendor testing tools to be integrated in its structure

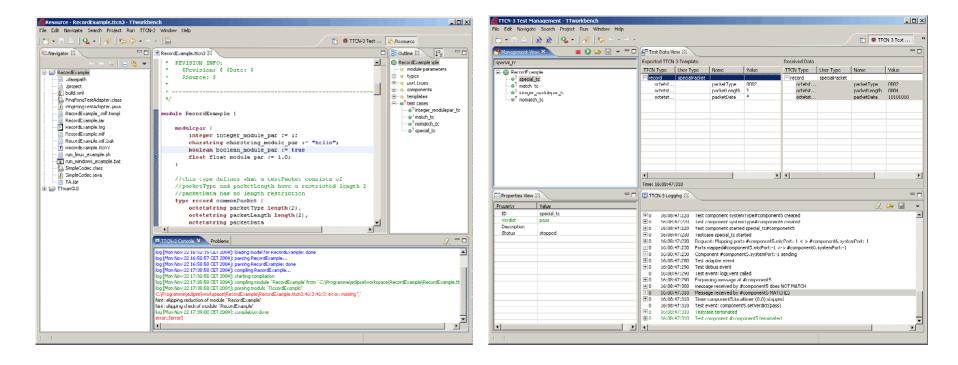
Benefit / Use Hyades tiers

- User Interface
 - provides testing, monitoring, profiling, logging tools
- Data Models
 - models for storing the persisting data
- Execution Framework
 - an extensible mechanism for executing and controlling tests

5



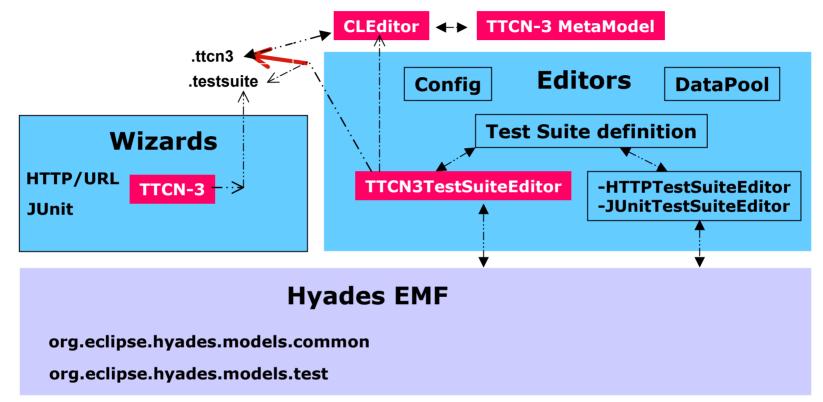
Separate TTCN-3 Eclipse based products





7

Hyades integration approach



- → use Hyades generic wizards and generic test suite editor
- \rightarrow create two kinds of resources
 - one based on TTCN-3 metamodel
 - one based on Hyades metamodel

Integration of TTCN-3 Fokus Tools in Hyades | © 2005 by «Diana Vega, Bernard Stepien»; made available under the EPL v1.0



Steps for the integration/Implementation

- Wizard support
 - New ->Test Suite->TTCN-3 Test Suite
 - Preference pages
 - Pop-up action/options
- Integration of the Editor
 - **CLEditor** –abstract TTCN-3 test specification editor for Eclipse
 - TTCN3TestSuite Editor
 - based on Hyades TestSuiteEditor
 - includes CLEditor

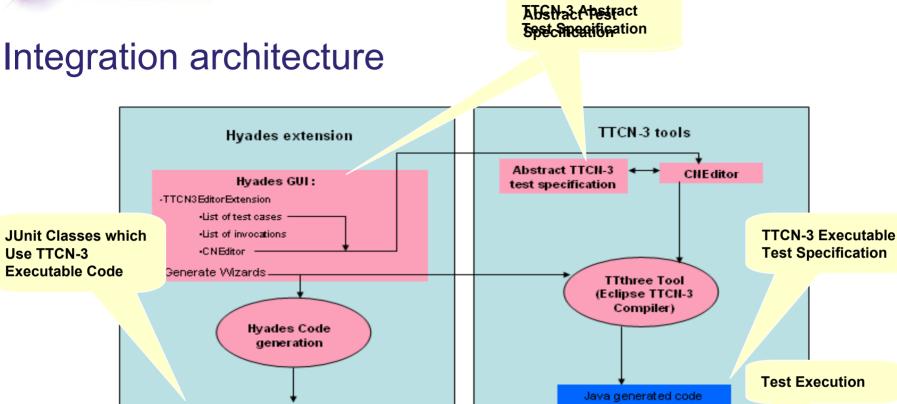
Integration of the Compiler

- GUI support via Generation Wizards
- a new wizard page is available with compilation options
- invoke TTCN-3 Compiler plug-in activity when wizard finishes

Execution

- start the Hyades execution engine
- call methods from TTCN-3 tools for test execution (the library TTthreeRuntime)
- visualize the execution: test verdicts via Hyades generic execution editor





Start Execution (find the testcase body in TTCN-3 Executable)

JUnit Classes which

Hyades JUnit test case Class

....etc

executeTTCN3TestCase("testCase1") }

executeTTCN3TestCase("testCase2") }

Hyades GUI – Visualize the execution

method testCase1() {

Executable Code

Use TTCN-3

Logging, Verdict Visualization

executeTTCN3TestCase(...)

TTCN-3 Tools for test

Execution

Integration of TTCN-3 Fokus Tools in Hyades | © 2005 by «Diana Vega, Bernard Stepien»; made available under the EPL v1.0



Hyades Extension points usage

Java code Generation – extension point definition

```
<extension

point="org.eclipse.hyades.test.ui.generateWizards">

<generateWizard

name="%GEN_WTITLE"

type="org.eclipse.hyades.test.ttcn3.junit.testSuite"

icon="icons/full/ctool16/generate_wiz.gif"

description="%GEN_WDESC"

class="org.eclipse.hyades.test.ttcn3.internal.junit.wizard.GenerateWizard"

extension="testsuite">

</generateWizard>

</extension>
```

TTCN-3 TestSuite New Wizard – extension point definition

```
<extension point="org.eclipse.ui.newWizards">
<wizard name="%WIZ_TST_TTCN3_SUITE_TTL"
icon="icons/full/obj16/ttcn3.gif"
category="org.eclipse.hyades.test.ui.wizards.new/testSuite"
class="org.eclipse.hyades.test.ttcn3.internal.junit.wizard.TTCN3TestSuiteNewWizard"
id="org.eclipse.hyades.test.ttcn3.internal.junit.wizard.TTCN3TestSuiteNewWizard">
<descriptionset.hyades.test.ttcn3.internal.junit.wizard.TTCN3TestSuiteNewWizard"
<description>
%WIZ_NEW_TTCN3_TST_SUITE_DSC
</description>
</wizard>
```



Usage – New Wizards – TTCN-3 Test Suite

🚝 Plug-in Development - ttcn3_1.testsu			
File Edit Navigate Search Project Run W [한 • 문 또 은 사용 · 관] 왕 • 회] 한 [한 문 Navigator • × 판 _{Ra} tton3_1.test	ଽ・ڲ・]ૐቆ້፟፟፟፟፟፟፟፟፟፟፟፟፟፟፟ ●] ይ ୬] % ·	Select Creates a new TTCN3 file	
Image: Second system Image: Second system Image: Second	→ Project P	Distributed TTCN Testing Eclipse Modeling Framework Eclipse Modeling Framework Eclipse Model Creation Wizards Distributed TTCN Test Suite Distributed TTCN Testing Eclipse Modeling Framework Eclipse Model Creation Wizards Distributed TTCN3 Test Suite Java Other Plug-in Development Profiling and Logging Simple StatCon Test Recording Test Elements Test Suite UML Diagrams Examples	
Test Na Executi, Project Java/test_ttcr Compare With Replace With	Properties		
Restore from Local H Properties	istory	< Back Next > Finish	Cancel



Usage – Edit the resource in the TTCN-3 MultiPage Editor

🗲 Te	est - TestSuite_TTCN3 - Eclipse	Platform		E	Test - TestSuite_TTCN3 - Eclipse Pla	itform	
File I	Edit Navigate Search Project De	ployment Run Window Help		File	Edit Navigate Search Project Deploy	ment Run Window Help	
È	· 🗆 🖳 🛆 🗛 🕅 🗍 🖓	· 💩] 🎭 •] 🖋] 🍫 🔶 • 🔿 -			Š - E E. A ¶ Bk AŠ - ø	與 灸・ タ *5 ⇔・⇔→	
Ē	💐 Test Navigator 🛛 🔻 🗙	E TestSuite_TTCN3 ×		Ē	💐 Test Navigator 🔍 👻 🗙	E: TestSuite_TTCN3 X	
	6 0 E ¥\$	TTCN3 Test Suite Definition Edito	r 🄶	={]⊳	00E X\$)	<u>^</u>
	SimpleProject TestProject TCN3TestProject TotSuite_Http TestSuite_JUnit TestSuite_TTCN3 TestSuite_TTCN3	General Information This section describes the general information about this object. Name TestSuite_TTCN3 Description TestSuite_TTCN3_description Type: TTCN3 Test Suite File: /TTCN3TestProject/TestSuite_TTCN3.testsuite Source Information Package Name: test Currentian TUCN2 Test Cores: Perhavier, TUCN2 code addit	▼TTCN3 Test Cas The following TTCN3		SimpleProject TestProject TCN3TestProject TStSuite_Http TestSuite_Junit TestSuite_TTCN3 TestSuite_TTCN3	<pre>type component MTCType { port postamt bundespost; } testcase send_a_car() runs on MTCType system map(mtc:bundespost, system:bundespost); bundespost.send(renault); bundespost.receive("received in good cond setverdict(pass) } control { var verdicttype theVerdict; execute (send_a_car()); } } </pre>	
Ľ	Navigator Test Navi Execution	Overview TTCN3 Test Cases Behavior TTCN3 code edito	or		Navigator Test Navigator Execution N	Overview TTCN3 Test Cases Behavior TTCN3 code editor	
Τe	estSuite_TTCN3 [/TTCN3TestProject/Te	stSuite_TTCN3.testsuite]			TestSuite_JUnit [/TTCN3TestProject/TestSuit	e_JUnit.testsuite]	



Test suite editor integration

🚰 Test - PingPong - Eclipse Platform		🚝 Java - PingPong - Eclipse Platform
Elle Edit Navigate Search Project Run Window Help		Elle Edit Navigate Search Project ITMex Profile Run Window Help
] 🖆 + 🔜 🗁] 券 + Q + Q +] A 🤤 Φ + ↔ +	😭 🖹 Test 🐉 Java 👋	否・□□□ A 號 珍・★・矢・ 粂・ 公 谷 谷 谷 (* D = 4 號 告 □ = * č (
EQ Test Navigator 🛛 💫 🖳 PingPong 📲 PingPong X		🗄 🛱 Package Explorer 💌 🗙 🔚 PingPong X
<pre>// test cases definitions // test cases</pre>		Image: Description Image: Description Image: Description Image: Description Image: Description Image: Description Image: Description Image: Description Image: Description Image: Description Image: Desc
Overview TTCN-3 Test Cases Behavior TTCN-3 code editor		
Tasks 🗖 TTCN+3 Console 🛛	Clean 🗖 🗆	
log (Sun Feb 27 12:02:59 EST 2005): loading model for PingPong: done	×	Package Explorer Hierarchy



Usage–Generation of the Java code among wizards

Test - TestSuite_TTCN3 - Eclipse Platfo	orm.		Java - demo_TTCN3 File Edit Navigate Seard		
) 🛎 - 🗔 🔍 📥 🗞	TTCN-3 Executable Code and JUnit class generation TTCN-3 Executable Test Definition Code Generation	9
ile Edit Navigate Search Project Deploymer			*>	The TTCN-3 test's java code can not modifyed. The java generated classes are related to the TTCN-3 code mentioned in the test definition.	
🛎 • 🗟 🖳 🗛 🗛 🛍 🛛 🖓 • 🖉	弐 ・ ∦ ∜⊨ (= + => →		Test Navigator	[Logging] Code Generation Java Compiler Module Loader	
Image: SimpleProject Image: SimpleProject Image: SimpleProject Image: SimpleProject	estSuite_TTCN3 × TCN3 Test Suite Definition Editor General Information a section describes the general information about object.	TTCN3 Test Cas	ImportProject ImportProject <t< th=""><th>Verbosity level C debug C log C hint C warning C error C failure</th><th>M</th></t<>	Verbosity level C debug C log C hint C warning C error C failure	M
TestSuite_TTCN3	stSuite_TTCN3 scription stSuite_TTCN3_description	Ξ	Package Explorer Test binoPono I (1) Init/dinoPono C Java - demo_TCDP File Edit Navigate Sear J → → → → → C → → → →	3.1 - Eclipse Platform	
File	e: /TTCN3TestProject/TestSuite_TTCN3.testsuite			src	
test	ckage Name: st	<u>v</u>	- 또했 JUnit - 또했 pindPong - 당동 pindPong - 당동 TTCN3_0 - 당동, TTCN3_2 관 (글 TTCN3_Split_Fil		'se
				Cance Next > Einish Cance	:el
TestSuite_TTCN3 [/TTCN3TestProject/TestSuite_	_TTCN3.testsuite]		Package Explorer Tes	t Navigator Execution Navigator Tasks TTCN-3 Console	



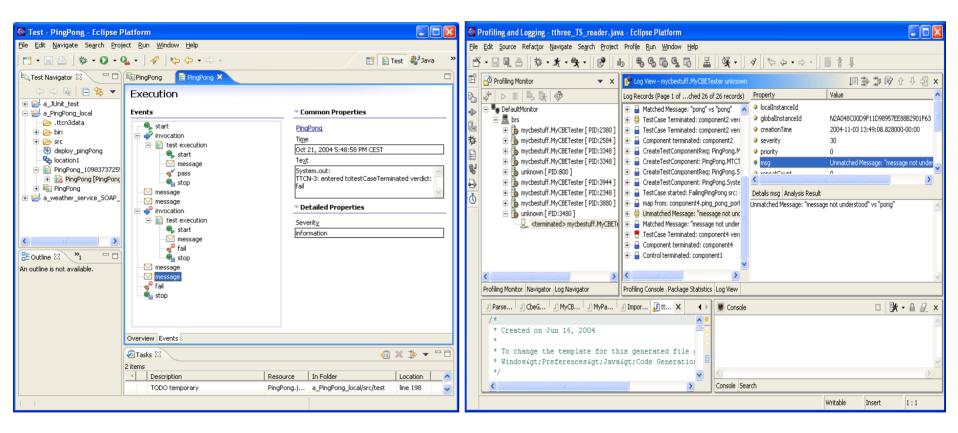
Execution integration

- Generated Hyades execution code consists in invoking a TTCN-3 test case from within a Hyades test case method's code.
- TTCN-3 test verdict is passed to Hyades via the JUnit fail() function for failed test cases and implicitly ignored for passed test cases.

```
public void playpingpong() throws Exception {
    // this code has been generated for Hyades/TTCN-3 - DO NOT MODIFY
    String theTCVerdict = executeTTCN3_TestCase("PlayPingPong");
    if (theTCVerdict.equalsIgnoreCase("fail")) {
        fail();
    } else if (theTCVerdict.equalsIgnoreCase("error")) {
        fail(); // this is a work around, errors are caught by TCN-3 tool
    } else if (theTCVerdict.equalsIgnoreCase("none")) {
        fail();
        fail();
```



TTCN-3/Hyades execution & logging





Hyades/TTCN-3 Integrations issues

- Mapping of the TTCN-3 control part into Hyades behavior concept is very limited. Hyades does not have conditional test case invocations capabilities like TTCN-3.
- TTCN-3 parametric test cases concept can not be mapped to any Hyades concepts.
- TTCN-3 test steps can not be mapped into Hyades concepts.
- Mapping TTCN-3 data templates can not be mapped to Hyades datapool concept mostly because TTCN-3 semantic requirements does not allow data to be external to TTCN-3.
- Logging concept differences: Hyades is reporting only pass or fail while TTCN-3 reports on evaluation of alternatives and could use more diversified color coding.
- No mapping between the TTCN-3 modularity and Hyades (no imports).



Conclusions – Further work – TPTP Integration

- TTCN-3 integration uses the least common denominator principle which implies that many powerful TTCN-3 concepts must be left out.
- TPTP Platform sub-project integration
- Choreography Component
 - Testability Interface