Running Regression Tests From Command Line

You can run regression test from command line. This is needed, if you want to e.g. integrate the regression tests into a continuous integration server such as Jenkins.

Call the Regression Test Command Line Tool like:

```
java -jar RegTestRunner.jar -project <path to the Builder project folder>
[-suite <name of the test suite to be executed>] [-logfile <filename
including path>] [<target Bridge>]
```

e.g. java -jar RegTestRunner.jar -project "C:\E2E Documentation\Advanced Modeling\PState" - suite "Build" -logfile C:\Temp\testlog.xml.

Example File (Builder project Advanced Modeling/PState):

<your example path>\Advanced Modeling\PState\uml\pstatePurchaseOrder.xml

Parameters

Regression Test Parameters

To select the tests to perform, you can specify the following parameters:

Parameter	Mandatory	Description	Example
-project	0	Specify the path to the Builder project that contains the regression tests.	-project "C:\E2E Documentation\Advan ced Modeling\PState"
-suite		Specify the test suite you want to run. The test suite must exist within the project given with -project. If you specify no test suite, all tests of the project will be executed.	-suite "QA Tests /PurchaseOrderExamp le"
-file		Deprecated since Analyzer 6.0.27Specify the path to a testsuite.xml that contains the tests you want to perform. For more information on where to find such a file, refer to Regression Test File and Folder Structure.	-file "C:\E2E Documentation\Advan ced Modeling\PState\reg ressiontest\QA Tests\testsuite. xml"
-list		Lists all available test suites in the project specified by -project. No tests executed.	-project "C:\E2E Documentation\Advan ced Modeling\PState" - list
-logfile		Optionally specify a filename (including extension of your choice) and a path. The RegTestRunner will generate a file with the given name to the given path. If you specify a filename only, the logfile will be generated to the location of the test suite. Image: This logfile is not to be confused with the logfile that is created if you have added a logfile option in the Regression Test Tool (see Logging Test Runs).	-logfile C: \Temp\myLogFile.txt
		For more information on the contents of the logfile, see Format of the Test Logfile further below.	

 Parameter 	eters
c	Regression Test
	Parameters
c	Target Bridge
	Parameters
 Setting 	a Request Timeout
 Examp 	les
 Format 	t of the Test Logfile

 Troubleshooting: Bridge Certificate Exception

Problem

Solution

Related Pages:

On this Page:

- Regression Test File and Folder Structure
- Setting a Request Timeout for the Analyzer

Target Bridge Parameters

Optionally specify a target Bridge to run the tests against. Specify all or none of the following parameters:

Parameter	Description	Example
-host	Host name of the Bridge. Used for SOAP, deploy, start and stop tests.	-host testbridge. e2e.ch
-port	Bridge port. Used for deploy, start and stop tests.	-port 8080
-username	Bridge user to authenticate for deploy, start and stop tests.	-username admin
-password	Password of the Bridge user. Used for deploy, start and stop tests.	-password secure1234
-node	Bridge node name. Used for deploy, start and stop tests. If it is the same as -host , you can skip this parameter.	-node localhost

Note, that the target Bridge parameter values will only be applied to the tests, if the tests do not have these values specified directly in the test themselves. Tests that have e.g. specified a dedicated host in the test case properties will not get overwritten by the target Bridge parameter values.

To make your tests compliant to using the target Bridge parameters, you have to specify the related data (host, port, user credentials and node) via test case options. How to do that is explained in detail on Addin g Options to a Test Suite pp.

Setting a Request Timeout

Analyzer 7.2.0 Regression test actions that do a network request have a timeout. You can change the timeout defaults via Java system properties:

Action	Default Timeout	Java System Property	Value
Establish connection	1 minute	<pre>sun.net.client. defaultConnectTimeout</pre>	Specify a value in milliseconds. A value of -1 means: no timeout.
Wait for response	10 minutes	sun.net.client. defaultReadTimeout	Specify a value in milliseconds. A value of -1 means: no timeout.

The following test actions do support timeouts:

- Deploy Bridge Repository (Deploy Test)
- Start Bridge Service
- Stop Bridge Service
- Send a Request to the Bridge

Set the changed timeouts via the Regression Test Command Line Tool call like

java -Dsun.net.client.defaultReadTimeout=5000 -jar RegTestRunner.jar ...

The command above starts tests with a timeout of 5000 ms for waiting for a response of network requests.

To set the same timeouts for the Analyzer refer to Setting a Request Timeout for the Analyzer.

Examples

To try out the Regression Test Command Line Tool, you can use the PState Builder project that is delivered with the E2E Examples.

Example File (Builder project Advanced Modeling/PState):

<your example path>\Advanced Modeling\PState\uml\pstatePurchaseOrder.xml



Analyzer	Regression Test Command Line Tool	
Overview Trace Analyzer Log Analyzer Regression Tests Properties Options Options Options Options Properties Options Properties Options Options Start PurchaseOrderExample Options Properties Options Properties Properties Properties Properties Options Properties Options	C:\E2E Documenta tion\Adva nced Modeling> java - jar RegTestRu nner.jar -project PState - list /Build Tests /QA Tests /QA Tests /Purchase OrderExam ple	
Options Image: Create Purchase Order 1 Image: Create Purchase Order 2 Image: Create PurchaseOrders Image: Create PurchaseOrders Image: Create PurchaseOrder Image: Create PurchaseOrder		

cumenta .on\Adva ed deling> ava ar egTestRu ner.jar project State st Build ests A Tests A Tests urchase derExam .e A Tests urchase derExam .e reatePu haseOrd A Tests urchase derExam .e addNewIt A Tests urchase derExam .e addGrati cations A Tests /Purchase OrderExam ple /getAllPu rchase0rd ers /QA Tests /Purchase OrderExam ple /getPurch aseOrder /QA Tests /Purchase OrderExam ple

/checkOut /QA Tests /Purchase OrderExam ple /getPurch aseOrders ByDate /QA Tests /Purchase OrderExam ple /closePur chase0rde r /Dev Tests /Dev Tests /Purchase OrderExam ple /Dev Tests /Purchase OrderExam ple /createPu rchase0rd er /Dev Tests /Purchase OrderExam ple /addNewIt em /Dev Tests /Purchase OrderExam ple /addGrati fications /Dev Tests /Purchase OrderExam ple /getAllPu rchase0rd ers /Dev Tests /Purchase OrderExam ple /getPurch aseOrder /Dev Tests /Purchase OrderExam ple /checkOut /Dev Tests /Purchase OrderExam ple /getPurch

aseOrders
ByDate
/Dev
Tests
/Purchase
OrderExam
ple
/closePur
chaseOrder



C:\E2E Documenta tion\Adva nced Modeling> java jar RegTestRu nner.jar -project PState suite "Build Tests" Running Test '<buildRe positoryT est>Build repositor y of Purchase0 rderExamp le'. Finished: 19.094 seconds <testsuit es> <testsuit e name=" TestSuite s.Build Tests" tests="1" > <testcase errors=" 0" name=" Build repositor y of Purchase0 rderExamp le" time=" 19.091"/> < /testsuit e> < /testsuit es> C:\E2E Documenta tion\Adva nced Modeling> java jar RegTestRu nner.jar

-project

🚷 E2E Analyzer - Regression Tests: PState				- 0	×
Project View Options Help					
🐠 🕂 🖕 😋 📾 🛐 🕕					
Overview d ^a P				Cane	cel
Trace Analyzer Log Analyzer Regression Tests	Testment				
🕀 🏕 TestSuites	rest progress				
Properties Options	0/4		Providence -		
B- Build Tests	Runs: 0/1	Errors: U	Dapseo one:	: 08	
🖻 🏙 QA Tests	Hierarchy Errors				
Properties Onliner	1ef Add Item 2				
AurchaseOrderExample					
Properties					
Grad Cardena Card					
🖻 🏕 addNewItem	e e e e				
Properties Options	Reference data Hide igr	ored differences	Accept changes	Test output d	lata
- W Add Item 1	o response, log:	27 2	response.log:		•
- af Add Item 2	<7xml version="1.0" enco	ding="UTF-28 2	xml version="1.</td <td>0" encoding="UTF-</td> <td></td>	0" encoding="UTF-	
Assertion Failed Error - [Reference	<soap-env:envelope td="" xmlns<=""><td>: SOAP-ENV- 29 2</td><td><pre>SOAP-ENV:Envelop</pre></td><td>e xmlns:SOAP-ENV-</td><td>÷ 🖕</td></soap-env:envelope>	: SOAP-ENV- 29 2	<pre>SOAP-ENV:Envelop</pre>	e xmlns:SOAP-ENV-	÷ 🖕
A actAlPurchaseOrders	xnlns	xed="http30 3	0	xmlns:xsd="http	E
getPurchaseOrder	xnins	- SOAD_FNC- 32 3	2	xmins:pridge="r	
🗈 🥵 checkOut	- xnlns	xsi="htty 33 3	3	xmlns:xsi="http	
getPurchaseOrdersbyDate dosePurchaseOrder	<soap-env:body soa<="" td=""><td>P-ENV: encc 34 3</td><td><soap-env: b<="" td=""><td>ody SOAP-ENV:enc</td><td></td></soap-env:></td></soap-env:body>	P-ENV: encc 34 3	<soap-env: b<="" td=""><td>ody SOAP-ENV:enc</td><td></td></soap-env:>	ody SOAP-ENV:enc	
Stop PurchaseOrderExample	<ns0:addnewi< td=""><td>temRespons 35 3</td><td><soaf< td=""><td>-ENV:Fault></td><td></td></soaf<></td></ns0:addnewi<>	temRespons 35 3	<soaf< td=""><td>-ENV:Fault></td><td></td></soaf<>	-ENV:Fault>	
🗄 🥵 Dev Tests		Response> 36 3		<faultcode>bridg</faultcode>	5 =
Upranes AvrhageOrderExample	*			-deard 2	· ~
	4	>	< I channel 1	<	_
	* changes		Changed	nserved bele	ueu
				25.01.2018	08:53

suite "QA Tests /Purchase OrderExam ple" Running Test '<startse rvice>Sta rt Purchase0 rderExamp le'. Running Test '<Testcas eTest>Cre ate Purchase Order 1'. Running Test '<Testcas eTest>Cre ate Purchase Order 2'. Running Test '<Testcas eTest>Add Item 1'. Running Test '<Testcas eTest>Add Item 2'. Running Test '<Testcas eTest>Add Gratifica tions for Wishes Unlimited ۰. Running Test '<Testcas eTest>get AllPurcha seOrders' Running Test '<Testcas eTest>Get Purchase Order 1'. Running Test '<Testcas eTest>Get Purchase Order 4'. Running Test '<Testcas eTest>Che

PState -

ck Out Order 1'. Running Test '<Testcas eTest>Che ck Out Order 2'. Running Test '<Testcas eTest>Get Purchase Orders from 1970-11-01'. Running Test '<Testcas eTest>Clo se Purchase Order 1'. Running Test '<Testcas eTest>Clo se Purchase Order 2'. Running Test '<stopser vice>Stop Purchase0 rderExamp le'. Finished: 8.261 seconds <testsuit es> <testsuit e name=" TestSuite s.QA Tests. Purchase0 rderExamp le" tests=" 15"> <testcase errors=" 0" name=" Start Purchase0 rderExamp le" time=" 1.198"/> <testcase errors=" 0" name=" Stop Purchase0

rderExamp

le" time=" 5.694"/> < /testsuit e> <testsuit e name=" TestSuite s.QA Tests. Purchase0 rderExamp le. createPur chase0rde r" tests="2" > <testcase errors=" 0" name=" Create Purchase Order 1" time=" 0.316"/> <testcase errors=" 0" name=" Create Purchase Order 2" time=" 0.116"/> < /testsuit e> <testsuit e name=" TestSuite s.QA Tests. Purchase0 rderExamp le. addNewIte m" tests="2" > <testcase errors=" 0" name=" Add Item 1" time=" 0.119"/> <testcase errors=" 1" name=" Add Item 2" time=" 0.111"> <error message="

Assertion Failed Error -[Referenc e = D: E2EDocumenta tion 18\Advanc ed Modeling\ PState\re gressiont est\refer ence\libr aries\Pur chase0rde rExample\ addNewIte m\Purchas eOrderSer vice. Purchase0 rderPort. addNewIte m∖Add Item 2\request .log] [differen t] Expected text value 'l' but was '99' comparing < purchase0 rderID ...> 1< /purchase OrderID&g t; at /env: Envelope [1]/env: Body[1] /ns0: addNewIte m[1] /purchase OrderID [1] /text() [1] to < purchase0 rderID ...> 99< /purchase OrderID&g t; at /env: Envelope [1]/env:

Body[1] /ns0: addNewIte m[1] /purchase OrderID [1] /text() [1] [Referenc e = D: E2EDocumenta tion 18\Advanc ed Modeling\ PState\re gressiont est\refer ence\libr aries\Pur chase0rde rExample\ addNewIte m\Purchas eOrderSer vice. Purchase0 rderPort. addNewIte m∖Add Item 2\respons e.log] [differen t] Expected attribute name 'xmlns: ns0' but was 'null' comparing [...] < /testsuit es>

Format of the Test Logfile

The logfile generated by the -logfile parameter is formatted in a junit compatible xml format that is used by Jenkins and others:

```
<testsuites>

<testsuite name="TestSuites" tests="10"/>

<testsuite name="TestSuites.Dev Tests" tests="10"/>

<testsuite name="TestSuites.Dev Tests.PurchaseOrderExample" tests="

10">

<testcase errors="0" name="Create Purchase Order 1" time="0.854"/>

<testcase errors="1" name="Create Purchase Order 2" time="0.01">

<testcase errors="1" name="Create Purchase Order 2" time="0.01">

<testcase errors="1" name="Create Purchase Order 1" time="0.854"/>

<testcase errors="1" name="Create Purchase Order 2" time="0.01">

<testcase errors="1" name="Create Purchase Order 1" time="0.01">

<testcase errors="1" name="Create Purchase Order 2" time="0.01">

<testcase errors="1" name="Create Purchase Order 2" time="0.01">

<testcase errors="1" name="Create Purchase Order 2" time="0.01">

<testcase</testcase>

[...]
</testsuite>
</testsuite>
```

The XML structure of the logfile reflects the test case structure in the Regression Test Tool and gives some additional information on the test execution:

XML Element	XML Attribute	Description
testsuites	-	Root element.
testsuite	name	Name of the test suite including test suite path in a test suite hierarchy.
	tests	Number of tests in the test suite.
testcase	name	Name of the test case.
	errors	Number of errors during test execution.
	time	Elapsed test time in seconds.
error	message	Text of the test error message.

Troubleshooting: Bridge Certificate Exception

Problem

Running tests with the Regression Test Command Line Tool, Bridge calls fail and you are getting this error: java.net.ssl.SSLHandshakeException: java.security.cert. CertificateException: Certificates does not conform to algorithm constraints.

Solution

Check the certificate on your Bridge (see <u>Managing the Bridge Certificate</u>). MD5 certificates are deprecated as insecure. If your Bridge still uses MD algorithm (**MD5withRSA**), you should create a new certificate and restart the Bridge. The new certificate should have algorithm **SHA256withRSA**.

Workaround:

If you can not fix the Bridge as described above (for whatever reason), you can change file <jre>/lib /security/java.security of the JRE you use to run the Regression Test Command Line Tool. Remove the "MD5" parts in jdk.certpath.disabledAlgorithms and jdk.tls.disabledAlgorithms to enable the usage of MD5 certificates.