



Update TestStep and Run ExecutionList using TC Shell

Hands-on Guide

Contents

What is this document for?_	3
Disclaimer _	3
Update TestStep and Run ExecutionList using TC Shell_	4

What is this document for?

This hands-on guide will allow you to explore the process of **Updating the TestStep and run the ExecutionList using TC Shell** without opening Tosca Commander and enhance your functional knowledge of the product.

In this document, you are provided with detailed instructions to walk you through the steps to achieve the desired result(s) for the listed assignment(s). We encourage you to use this document to support your knowledge of Tosca, compare it with your alternate practical implementation method(s), and improve your overall know-how of the tool.

Disclaimer

Information in this document is subject to change without notice. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose without the express written permission of Tricentis GmbH.

© 2023 by Tricentis GmbH

Update TestStep and Run ExecutionList using TC Shell

+ Objective

By the end of these exercise(s), you will be able to update the TestStep and execute the ExecutionList using TC Shell without opening Tosca Commander

+ Why is this important?

The TC-Shell is the command-line control of Tosca Commander and can be started in two different modes, interactive and script.

- ◆ The interactive mode provides you with assistance and options. You have access to the full range of functions of the GUI version in the Tosca Commander
- ◆ In the script mode the Shell works off a series of commands from a so-called script data file. In this process, you have no possibility to intervene with any interaction. This mode serves for the automation of processes, for instance, to be able to work off an extensive test portfolio without supervision

Using TC Shell, you will update the TestStep and execute the ExecutionList with interactive mode and scripting mode.

+ Prerequisite

- 1 | Open any Tosca Workspace
- 2 | Create a component folder with the name **TC Shell Hands On**
- 3 | Import the base subset **Create new text file.tsu**
- 4 | Execute the test case once
- 5 | You will observe a File named **Tosca File.txt** would be created at
C:\Tosca_Projects\Tosca File.txt

+ 1. Working with Interactive Mode

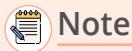
Using the Interactive mode, we can directly write the commands in the Command Prompt and execute them.

+ Steps to Perform

Here are the steps you need to perform to update the TestStep and execute the ExecutionList using the interactive mode. Please remember to follow the steps in sequence to achieve the desired results.

Update TestStep and Run ExecutionList using TC Shell

Section I: Open Tosca Workspace and note the TestStep and ExecutionList node path



Note

As mentioned in the pre-requisites, import the base Subset either at the project root level or in a new Component Folder to begin with the steps below.

- 1 | Open the Tosca Workspace where you imported the Subset. Navigate to the TestCase section and open the TestStep Value **Text** and expand the **Properties** tab. Copy the NodePath and note it down in a notepad.

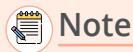
Properties	
Name	Value
Text	
NodePath	/TC Shell Hands On/TestCases/Create File with data/Set the Buffer/Text
CreatedBy	Admin
CreatedAt	21-12-2022 16:04:28
ModifiedBy	Admin
ModifiedAt	17-01-2023 20:36:59
HasMissingRefere...	False
UniqueId	3a08466c-8c1f-48b1-d7a0-f91fdbb4bcd3
DataType	String
ValueToUse	Tricentis Tosca is a continuous testing platform to automate your test ca...
DisabledDescription	
Condition	

- 2 | Now navigate to ExecutionList name **Create File with data**, and expand the Properties tab. Copy the NodePath and note it down in a notepad.

Properties	
Name	Value
Create File with data	
NodePath	/TC Shell Hands On/Execution/ExecutionLists/Create File with data
CreatedBy	Admin
CreatedAt	21-12-2022 16:13:31
ModifiedBy	Admin
ModifiedAt	21-12-2022 16:14:52
HasMissingRefere...	False
UniqueId	3a084674-d61d-dbab-790e-26a73299bd0a
SynchronizationPo...	CustomizableDefaultIsOn
Description	
Revision	0
OwningGroupName	All Users (inherited)
ViewingGroupName	<NO VIEWING GROUP>
IsCheckedOutByMe	False
Include in Analytics	True
IsMandate	False
NumberOfEntries	1
NumberOfTestCas...	1
NumberOfTestCas...	0
NumberOfTestCas...	1
NumberOfTestCas...	0
NumberOfTestCas...	0
ManualItems	0
AutomatedItems	2

Update TestStep and Run ExecutionList using TC Shell

3| Close the Tosca Workspace.



Note

Please make sure while using TC Shell, Tosca Commander is closed.

Section II: Open Command Prompt and enter the command

1| Open the Command Prompt.

2| Go to the Tosca Commander directory using the command -

cd "C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander"
then press Enter.

```
C:\ Command Prompt  
Microsoft Windows [Version 10.0.19045.2364]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\kavgu>cd "C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander"  
  
C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander>
```

3| Go to the Tosca Workspace where you imported the subset using the command –

TCShell.exe -workspace "C:\Tosca_Projects\Tosca_Workspaces\Test\Test.tws" -login "admin" null then press Enter.

```
C:\ TCProject: 'Test'  
Microsoft Windows [Version 10.0.19045.2364]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\kavgu>cd "C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander"  
  
C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander>TCShell.exe -workspace "C:\Tosca_Projects\Tosca_Workspaces\Test\Test.tws" -login "admin" null  
Checking License ... License found!  
TCPProject:Test >
```

4| Go to the TestStep using the command – (Please use the node path you had copied from Section 1 step 1) **jumptonode "/TC Shell Hands On/TestCases/Create File with data/Set the Buffer/Text"** then press Enter.

```
C:\ XTestStepValue: '/TC Shell Hands On/TestCases/Create File with data/Set the Buffer/Text'  
Microsoft Windows [Version 10.0.19045.2486]  
(c) Microsoft Corporation. All rights reserved.  
  
C:\Users\kavgu>cd "C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander"  
  
C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander>TCShell.exe -workspace "C:\Tosca_Projects\Tosca_Workspaces\Test\Test.tws" -login "admin" null  
Checking License ... License found!  
TCPProject:Test > jumptonode "/TC Shell Hands On/TestCases/Create File with data/Set the Buffer/Text"  
XTestStepValue:Text >
```

Update TestStep and Run ExecutionList using TC Shell

5| Type the command Task using the command – **task**

```
[x] XTestStepValue: '/TC Shell Hands On/TestCases/Create File with data/Set the Buffer/Text'
microsoft Windows [Version 10.0.19045.2486]
c) Microsoft Corporation. All rights reserved.

:\Users\kavgucd "C:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander"

:\Program Files (x86)\TRICENTIS\Tosca Testsuite\ToscaCommander>TCShell.exe -workspace "C:\Tosca_Projects\Tosca_Workspaces\Test\Test.tws" -login "admin" null
hecking license ... License found!
[Project:Test > jumptonode "/TC Shell Hands On/TestCases/Create File with data/Set the Buffer/Text"
TestStepValue:Text > task
1]= 'Jump to ModuleAttribute'
2]= 'Search all relevant XParams'
3]= 'Disable'
4]= 'Create XTestStepValue tree'
5]= 'Translate value'
6]= 'Create Reuseable TestStepBlock Parameters'
7]= 'Search all references'
8]= 'Search'
9]= 'Cut'
10]= 'Copy'
11]= 'Delete'
12]= 'Rename'
13]= 'Modify DataType'
14]= 'Modify ActionMode'
15]= 'Modify Value'
16]= 'Modify ActionProperty'
17]= 'Modify Condition'
18]= 'Modify Path'
19]= 'Purge missing references'
20]= 'Create XTestStepValue (after this)'
21]= 'Create Duplicate (after this)'
22]= 'Jump to Object in TestCase'
23]= 'Translate name'
24]= 'API Checkout'
25]= 'API Create Values from Default for XTestStep'
26]= 'Escape value'
27]= 'API Maintenance'
28]= 'API WebSocket Listener'
29]= 'API ValidateEndpointTask'
30]= 'API Message Tracing'
31]= 'API Manage'
32]= 'API BulkCreateBusinessParameters'
Select Task to execute by number # [1-32,0=none] or by name >
```

6| Now we can see all of the possible tasks that we can perform on a TestStep. In this case, you must modify the value of the TestStep so you need to select the task number 15 Modify Value. Type **15** and press Enter. (Alternatively, you can type the task name as well).

7| Then it will ask to use Regular Expressions. Type **no**.

8| Now, you need to write the modified text:

Tricentis Tosca is a Model Based Automation Testing Tool

```
[x] XTestStepValue: '/TC Shell Hands On/TestCases/Create File with data/Set the Buffer/Text'
[18]= 'Modify Path'
[19]= 'Purge missing references'
[20]= 'Create XTestStepValue (after this)'
[21]= 'Create Duplicate (after this)'
[22]= 'Jump to Object in TestCase'
[23]= 'Translate name'
[24]= 'API Checkout'
[25]= 'API Create Values from Default for XTestStep'
[26]= 'Escape value'
[27]= 'API Maintenance'
[28]= 'API WebSocket Listener'
[29]= 'API ValidateEndpointTask'
[30]= 'API Message Tracing'
[31]= 'API Manage'
[32]= 'API BulkCreateBusinessParameters'
Select Task to execute by number # [1-32,0=none] or by name > 15
Use Regular Expression (yes/no) > no
Replace with: > "Tricentis Tosca is a Model Based Automation Testing Tool"
1 objects modified!
XTestStepValue:Text >
```

Update TestStep and Run ExecutionList using TC Shell

9| Now go to the ExecutionList using the command – (Please use the node path you had copied from Section 1 step 2) **jumptonode "/TC Shell Hands On/Execution/ExecutionLists/Create File with data."**

```
ExecutionList: /TC Shell Hands On/Execution/ExecutionLists/Create File with data'
[18]='Modify Path'
[19]='Purge missing references'
[20]='Create XTestStepValue (after this)'
[21]='Create Duplicate (after this)'
[22]='Jump to Object in TestCase'
[23]='Translate name'
[24]='API Checkout'
[25]='API Create Values from Default for XTestStep'
[26]='Escape value'
[27]='API Maintenance'
[28]='API WebSocket Listener'
[29]='API ValidateEndpointTask'
[30]='API Message Tracing'
[31]='API Manage'
[32]='API BulkCreateBusinessParameters'
Select Task to execute by number # [1-32,0=none] or by name > 15
Use Regular Expression (yes/no) > no
Replace with: > "Tricentis Tosca is a Model Based Automation Testing Tool"
1 objects modified!
XTestStepValue:Text > jumptonode "/TC Shell Hands On/Execution/ExecutionLists/Create File with data"
ExecutionList:Create File with data >
```

10| Type command “Task” Using the command – **task**.

```
ExecutionList: /TC Shell Hands On/Execution/ExecutionLists/Create File with data'
[20]='Run'
[21]='Run as Manual TestCase'
[22]='Archive actual ExecutionLog'
[23]='Search all ExecutionEntries'
[24]='Search all ExecutionEntries: No Result'
[25]='Search all ExecutionEntries: Failed'
[26]='Search all ExecutionEntries: Passed'
[27]='Transfer Output values'
[28]='Clear Log'
[29]='Search all synchronized folders'
[30]='Synchronize'
[31]='Search all objects with connection properties'
[32]='Print Report ... ExecutionEntries with ActualLog'
[33]='Print Report ... ExecutionEntries with ActualLog and Issues'
[34]='Print Report ... ExecutionEntries with detailed Logs'
[35]='Print Report ... ExecutionLists with ArchiveLogs and ActualLogs'
[36]='Write automation objects'
[37]='API Checkout'
[38]='API Maintenance'
[39]='API WebSocket Listener'
[40]='API Message Tracing'
[41]='API Manage'
[42]='API BulkCreateBusinessParameters'
Select Task to execute by number # [1-42,0=none] or by name >
```

11 | Write Task number **20** or name **Run** to run the Execution Entry.

Update TestStep and Run ExecutionList using TC Shell

```
ExecutionList: '/TC Shell Hands On/Execution/ExecutionLists/Create File with data'
[27]='Transfer Output values'
[28]='Clear Log'
[29]='Search all synchronized folders'
[30]='Synchronize'
[31]='Search all objects with connection properties'
[32]='Print Report ... ExecutionEntries with ActualLog'
[33]='Print Report ... ExecutionEntries with ActualLog and Issues'
[34]='Print Report ... ExecutionEntries with detailed Logs'
[35]='Print Report ... ExecutionLists with ArchiveLogs and ActualLogs'
[36]='Write automation objects'
[37]='API Checkout'
[38]='API Maintenance'
[39]='API WebSocket Listener'
[40]='API Message Tracing'
[41]='API Manage'
[42]='API BulkCreateBusinessParameters'
Select Task to execute by number # [1-42,0=none] or by name > 20
Run started with TBoxRunTaskExecutor
Verifying required licenses ...
Execution Running ...
Processing execution results ...
Run finalizing ...
Run finished
ExecutionList:Create File with data >
```

Expected Result

- ◆ The Execution Entry will be executed successfully.
- ◆ The File **Tosca File.txt** would be updated with the new value.

+ 2. Working with Scripting Mode

To start the TC-Shell in the script mode, a script file has to be transferred in form of a parameter at the request.

+ Steps to Perform

Here are the steps you need to perform to update the TestStep and execute the ExecutionList using scripting mode. Please remember to follow the steps in sequence to achieve the desired results.

- Section I: Open Tosca Workspace and note the TestStep and ExecutionList node path

Please refer the Section 1 from the Interactive Mode section.

- Section II: Create TCS File

1 | Open Notepad++ and write the command (Please remember to update the node paths with the ones you had copied in section 1)

Update TestStep and Run ExecutionList using TC Shell

```
//Navigate to TestStep
jumpToNode "/TC Shell Hands On/TestCases/Create File with data/Set the Buffer/Text"
//Modify value
task "Modify Value"
no
"Tricentis Tosca is a continuous testing platform to automate your test cases"

//Navigate to ExecutionList
jumpToNode "/TC Shell Hands On/Execution/ExecutionLists/Create File with data"
//Execute the ExecutionList
task Run
Save
```

2| Save the file as **Update_Teststep_And_Run_EL.tcs**

Section III: Create Bat File and Execute it

1| Open Notepad++/Notepad and write the command

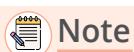
```
TCShell.exe -workspace <workspacePath> -login <username> <password> <tcsFilePath>
```

For example:

```
TCShell.exe -workspace "C:\Tosca_Projects\Tosca_Workspaces\Test\Test.tws" -login "
admin" null "C:\Tosca_Projects\Update_Teststep_Auth_Run_EL.tcs"
```

2| Save the file **Update_Teststep_Auth_Run_EL.bat**.

3| To execute the **bat** file, double-click on it.



If you run into issues make sure:

- 1| The double quotes are in the correct format, it is recommended to retype the double quotes both in the tcs and bat files.
- 2| The node path that you used is correct.
- 3| While using TC Shell, make sure Tosca Commander is closed.

Update TestStep and Run ExecutionList using TC Shell

Expected Result

- ◆ The **TestStep** is successfully updated via the Interactive and the Scripting Mode.
- ◆ The **ExecutionList** is successfully executed via the Interactive and the Scripting Mode.
- ◆ The File **Tosca File.txt** would be updated with the new value.