

MITE-UserManual

Table of contents

1	Introduction	6
1.1	Purpose	6
2	About MITE	6
2.1	Overview	6
3	MITE Installation	7
3.1	Pre-requisites	7
3.1.1	ALM1 Procedure	7
3.1.2	ALM2 Procedure	8
3.2	Executable file	11
4	MITE Home Window	11
5	Mapping a Project	12
5.1	Mapped Project Overview and Summary	14
5.2	Project Summary and Sub-tabs	15
5.3	Options available on Project -Right click	15
5.4	View More Details Option	16
5.5	Reload Project	16
6	Create Test Suite	16
6.1	Fill Test-Suite particulars	17
6.2	Test Suite Summary and Sub-tabs	17
6.3	Options available on Test Suite -Right click	19
6.4	Delete - Recover operations on Test-Suite	19
6.5	Load Test Suite	20
6.6	Shared/ReUse Test suite	20
6.7	Branching of a TestSuite in PTC	23
6.8	Delete scenario for Flushed Test Suite	29
6.9	Open Test suite from Project summary Table	31
6.10	Add member pop up	32
6.11	Baseline Test suite	33
7	Import Label database	38
7.1	Import Label database through Label Database window	39
7.2	Fill Import Label Database particulars	40
7.3	Options available on imported Label Database files -Right click	47
7.4	On View More Details	48
7.5	On View Table	48
7.6	On Copy Label Database File	49
7.7	On Paste Label Database File	49
7.8	On Delete Label Database File	50
7.9	On Reload Label Database File	51
7.10	On Update Label Database File	52
7.11	Update and Synchronize options for Diagnostics and Labels	52
7.12	Label data base file with respect to Dut's	58
7.13	PDX Extraction Changes in the version MITE - v2.1.6.0	60
7.14	PDX Extraction Changes in the version MITE - v 2.4.0.1	60
7.15	ARXML/FIBEX Arrayed Labels	61
8	MITE Parameters	62
8.1	Test Parameterization	62
8.2	MITE interface	63

8.3 Using Parameters in Test Cases	64
8.3.1 Adding a New Parameter	64
8.3.2 Import Parameters from another Test Suite	64
8.3.3 Add / Remove parameter(s) at Test Suite Level	65
8.3.4 Add / Remove parameter(s) at Test Case Level	66
8.3.5 Add parameter(s) in Test Case editor	67
8.3.6 Edit parameter save list option	68
8.3.7 Parameters Value SetIn	70
9 Parameter Naming Convention	71
9.1 Providing Parameter values in Test Cases or Test Sets	71
9.2 Rename parameter(s)	72
9.3 Reviewing and managing Parameter configurations	72
9.3.1 Label/User-define Parameters in MITE:	73
9.3.1.1 Label Parameters	73
9.3.1.2 User Define Parameters	74
9.4 Parameter Common Name	75
9.5 PTC Content:	75
10 Create Test Case	76
10.1 Fill Add Test Case particulars	76
10.2 Options available on Test Case-Right click	78
10.3 Delete - Recover operations on Test-Case	79
10.4 Add the capability in MITE to insert images to test cases	79
11 Test Case Editor	82
11.1 Elements of Test Case Editor	83
11.2 Key-Words for Test-Case(s) Authoring	84
11.3 Additional Remarks	85
11.4 Comments	86
11.5 Traceability	87
11.6 Recorders	90
11.7 Extensions	91
11.8 After Extensions in MITE	95
11.9 User Defined Labels.	96
11.10 Extensions Pre Processing	97
12 Copy/Paste Operations	99
12.1 Cut and Paste operation	99
13 Delete/Recovery Operations	103
14 Import/Export Operations	104
15 Shortcut Keys & Mouse operations	104
16 Flushing into PTC server	105
16.1 About Flushing	105
16.2 Environment for Flushing into PTC server	105
16.3 Contents which can be flushed into PTC	105
16.4 How to flush the contents into PTC server?	106
16.5 PTC Test Suite state validation	106
16.6 MITE Non-Dependency on Sandbox	107
16.7 MITE User Scenarios	108
17 Test Suite Resynchronization	110
17.1 Parameters Merge in Resync Window	113
17.2 Resynchronization Merge window.	113
18 Source View Set	118

19 Script generation	120
19.1 Test Bench Configuration and Label Mapping	120
19.1.1 Motion Desk TBC	129
19.1.2 Sensor sim	144
19.1.3 Innoviz XMB Reprocessing Test Bench Configuration	148
19.1.4 Magna Image Grabber	150
19.1.5 CANoe XIL Interface	153
19.1.6 CAR MAKER	157
19.1.7 Car maker with matlab(remote Access files)	180
19.1.8 Atmospheric condition	184
19.1.9 User Defined labels in TBC	186
19.1.10 Single\Multiple Bites and Bytes Information	187
19.1.11 Remote Batch Automation	188
19.1.12 Power Relay-Neo RAD	194
19.1.13 SCPI Power Supply	198
19.1.14 Export/Import TBC	202
19.1.15 Image Comparison	203
19.2 Configuration Enable – Disable	207
19.3 Label Mapping	207
19.3.1 MAP Labels	210
19.3.2 TBC Frames MAP Labels	211
19.3.3 LabelMapping based on the selected TestCases and their dependent TestCases Selection & Search ,Alphabetically order of Map labels	216
19.3.4 Autofill in the Label Mapping	219
19.3.5 Image Resolution setting in label Mapping frame	221
19.4 Test Script Generation	222
19.4.1 MITE Test Script Generation Classification	223
19.4.1.1 Test Script Generation	223
19.4.2 Project Creation	224
19.4.3 Project Creation: Diagnostic	225
19.4.4 Reload Configuration in Automation Desk	226
19.5 Test Script Execution	227
19.6 Test Script Generation using Test Session	227
19.7 Check & Uncheck all child cases from tree right click option	229
19.8 Requirements for Specific Mapper Implementation in TBC	231
19.9 Label Details Tab	233
19.10 MITE and Automation Desk Integration	234
19.11 Multiple Suite test script generation and project creation	235
19.11.1 Multiple Allowances of Remote Diagnostic Access Files	237
19.12 SDF Extraction of Arrayed Labels	239
19.13 LOGGERS FROM CONTROL DESK	240
19.14 CDD at TestScript Level	242
19.15 Hyperlink In Warning Tab	244
19.16 Multiselection of Test Cases in Test Script Viewset	245
20 MITE LabVIEW	248
20.1 Pre-requisites	248
20.2 Test Case Editor	249
20.3 Test Script	249
20.4 Test Bench Configuration	250
20.5 Configuration Enable – Disable	252

20.6 Label Mapping	252
20.7 Test Script (LabVIEW Vis) Generation	255
21 Other Additional Features	256
21.1 MITE Demo Project :- Demo project is added to the list for the first time MITE users	257
21.2 Spell Check:- A check for misspellings by using a 'spell check' option on 'right-click' for Description fields only at Test suite , at Test case levels and for Additional Remarks.	257
21.3 Optimize Test Case:- Appears on Test Case right click options.	258
21.4 Un-Do operation (Ctrl+Z) :- Un-do operation takes place on CTRL+Z	258
21.5 Re-Do operation (Ctrl+Y) :- Re-do operation takes place on CTRL+Y	259
21.6 MITE Statistics	259
21.7 Q & A Links	261
21.8 Downloading User Manual & FAQ	262
21.9 Find & Replace, Rename Functionallities	263
21.10 Email Attachment & Videos	264
22 Exercise – Test Case Authoring in MITE	264
22.1 CAN/LIN tests case(s)	265
22.2 Diagnostic Test Case(s)	266
22.3 Authoring rules for value entry	267
23 MULTIPLE NODE SELECTION IN TREE FOR COPY/PASTE	267
23.1 Multiple test case nodes selection with right click pop-up	268
23.2 Label database hiding	268
23.3 Right side page update while clicking node in multiple node selection	269
23.4 Validation for paste event after copy one or more test cases	271
23.5 Select and deselect multiple nodes from tree	271
23.5.1 Test Case level "Project summary" table	272
23.5.2 Test Case level "Test suite summary" table	273
23.5.3 Test Script level "Test Suite Summary" table	273
24 Test Session	274
24.1 Add / Map a Project	274
24.2 Add Test session in project	276
24.2.1 Add Test Session Frame	277
24.3 Reload Project	278
24.4 Load Test Session	279
24.5 Reload TestSession	281
24.6 Project creation and Load/Reload TBC	281
24.7 Test Session START/STOP Execution	283
24.8 Test Objectives	285
24.9 Delete Test Session	286
24.10 Consolidated HTML report	287
24.11 Global Header Test Run details	288
24.12 Test Session Flushing into PTC Report Path	294
24.13 Multiple Time Execution without user intervention	298
25 Test Case Flush	299
25.1 Description	299
25.2 Implementation	299
26 TEST Script Flush	300
26.1 Reload test Script From PTC	304
27 MITE SUPPORT	304

1 Introduction

Introduction

Created with the Personal Edition of HelpNDoc: [Full-featured EPub generator](#)

1.1 Purpose

This document describes the features of MITE and its usage for Test-Case(s) Authoring.

Created with the Personal Edition of HelpNDoc: [Free EPub and documentation generator](#)

2 About MITE

About MITE

Created with the Personal Edition of HelpNDoc: [Create iPhone web-based documentation](#)

2.1 Overview

This tool is an Integrated Test Environment for Test-Case authoring, Test-Script generation based on test bench configuration and Test-Execution software for the validation of embedded systems in automotive environments.

- ✓ Test-Case authoring in a **standardized format**
- ✓ **Generates Test-Scripts** based on the test bench configuration
- ✓ Supports a wide range of test tools and test environments (HIL, Vehicle)

- ✓ **Intuitive Graphic User Interface**
- ✓ Interface to **Application life cycle management tools (PTC Integrity)**
- ✓ **Standardization** of all the test artefacts (Test-Cases, Test-Scripts and Results).
- ✓ Because of its **modular architecture** we quickly customize and integrate to any tool involved in the testing process.

Created with the Personal Edition of HelpNDoc: [Full-featured EPub generator](#)

3 MITE Installation



Created with the Personal Edition of HelpNDoc: [Qt Help documentation made easy](#)

3.1 Pre-requisites

❖ Java Software

Before executing MITE, Java needs to be installed in the respective PC/System in which MITE tool executes.

- a. Java software can be installed from the Google web link
- b. Java can be installed using the “MITE.exe” file, by clicking on the MITE.exe, user will be directed to the Java Software installation web link

TOOLS/Versions- All MITE versions are supported as mentioned with different tools and their versions.

1. Automation Desk(5.4/5.6)
2. Vector Canoe(Versions 11 to 15)
3. Control Desk(Below 6.3)
4. Motion Desk(4.2)
5. Model Desk(4.7)
6. Vector CANAPE--15
- 7.VLC Media-64 bit
8. Programmable power supply tools--a)BK Precision 1697/1696 ,9201/9205 series
b)TDK-Lambda-Z series
c)TDK-Lambda-GENSYS
9. Innoviz record reprocessing:-Lidar_Prosim4.1.3
10. MIG server-1.5

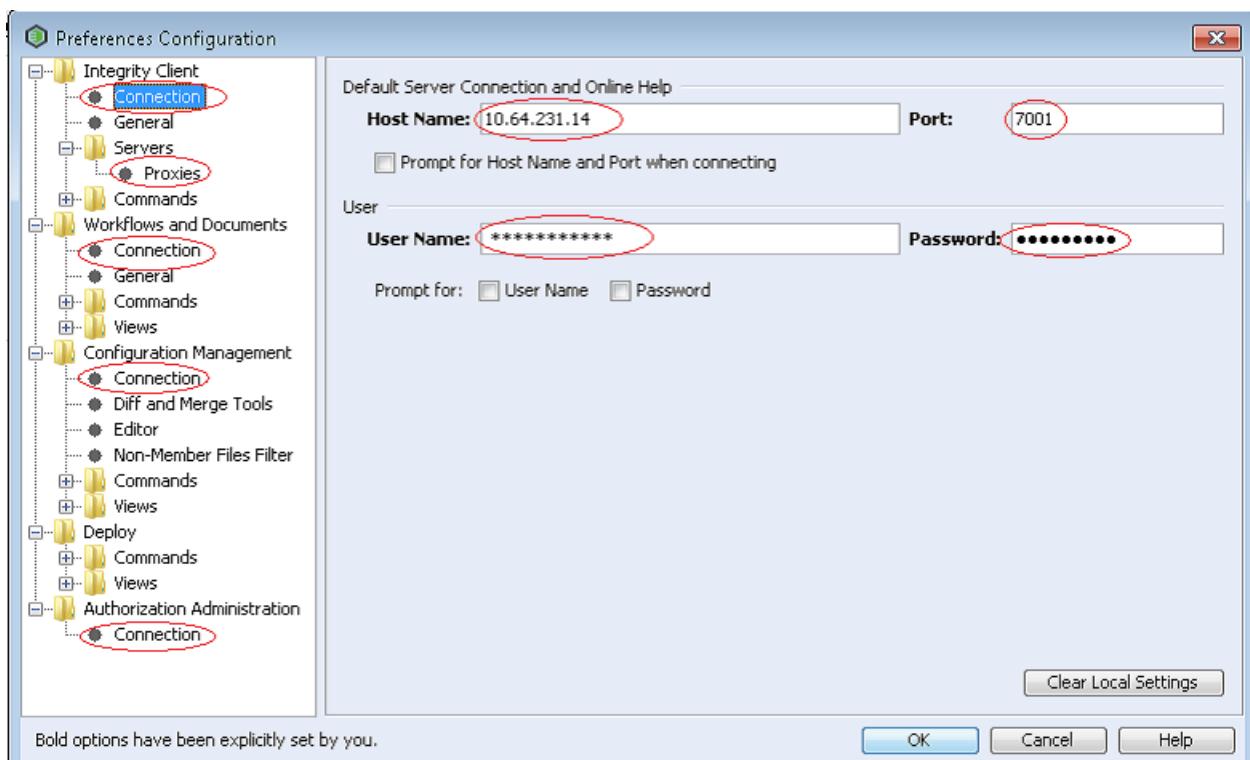
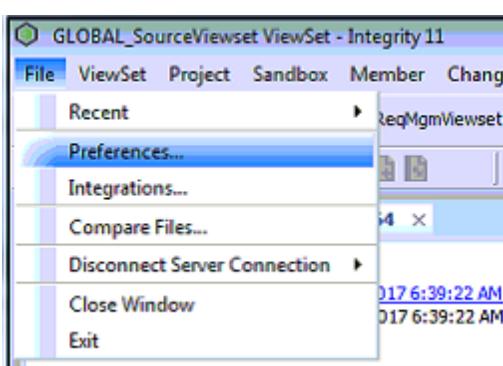
Created with the Personal Edition of HelpNDoc: [Create cross-platform Qt Help files](#)

3.1.1 ALM1 Procedure

For Using ALM1, User need to follow below procedure to be in sync with PTC:-

PTC settings □

1. PTC client should be installed in the system already
2. PTC client should be open and running
3. Should have Host Name and Port Name filled in all the required areas in File □ Preferences; as shown below



4. Environmental path settings:

"C:\Program Files (x86)\Integrity\ILMClient11\bin"

5. Uncheck "Prompt for User Name and Password" in all the places

Created with the Personal Edition of HelpNDoc: [Free Web Help generator](#)

3.1.2 ALM2 Procedure

For using ALM2 ,User need to follow below procedure to be in sync with PTC:-

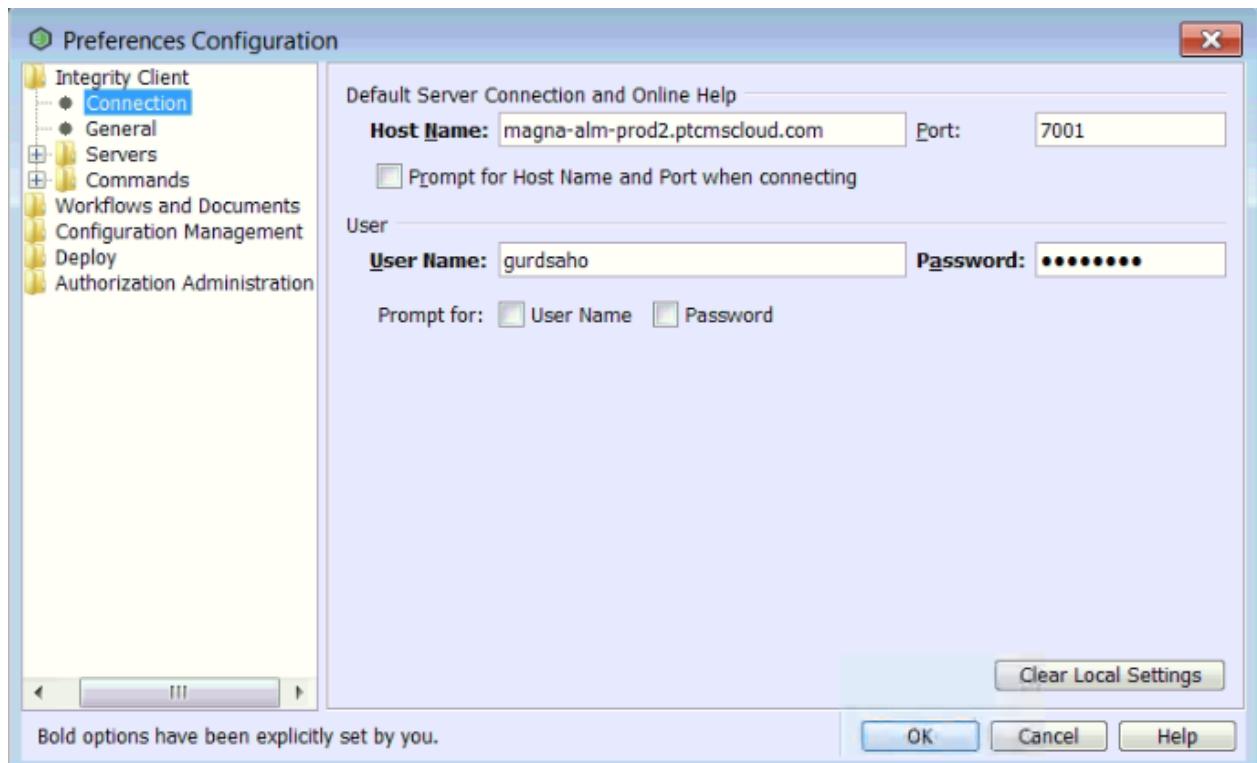
1. Create **settings** folder in this path "C:\Program Files (x86)\Integrity\ILMClient11_2\settings"

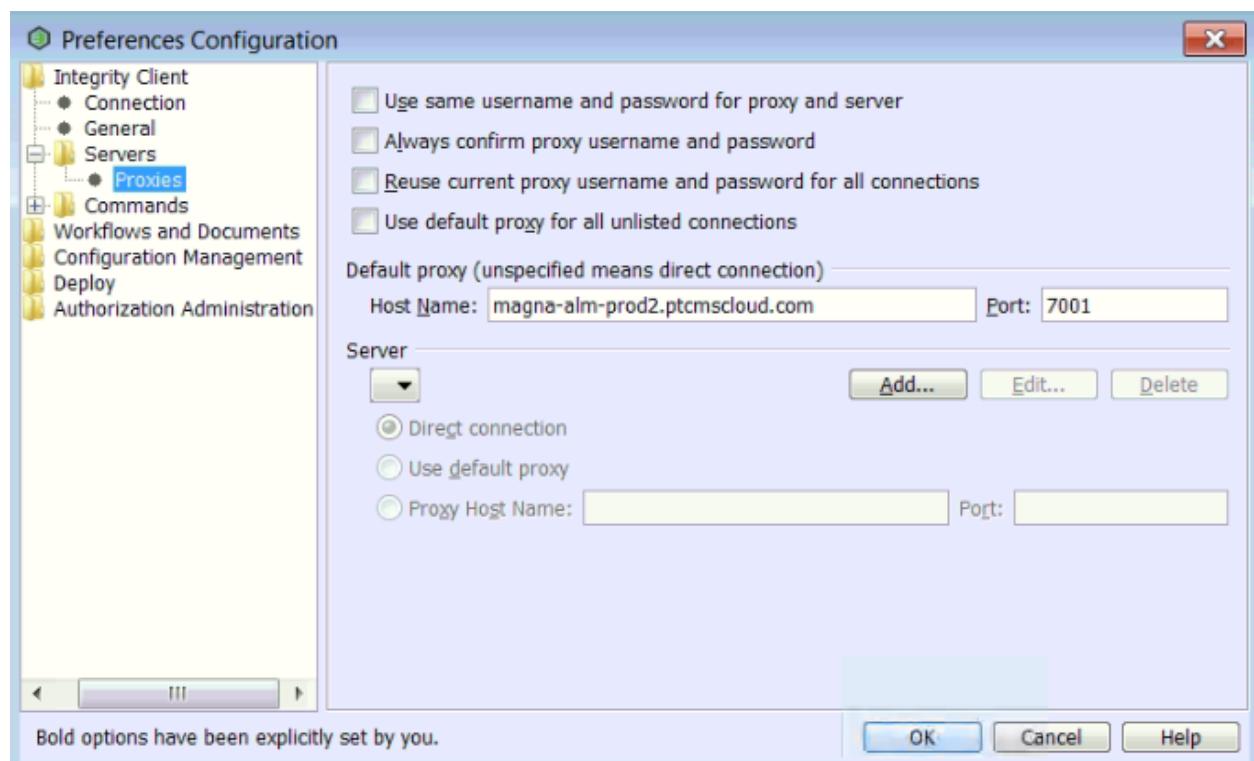
2. Place the below batch file on Desktop to run User Client for 11.2



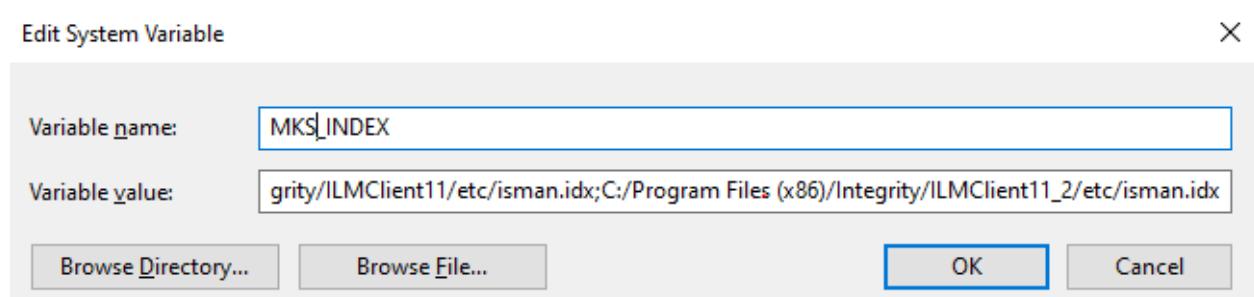
Integrity 11_2
UserClient.bat

3. Copy files "IntegrityClientSite.rc" and "jssecacerts" from \\eahmsrepo01\PTC_Software\Integrity_V11_2 and paste on user's desktop
4. Copy "jssecacerts" to C:\Program Files (x86)\Integrity\ILMClient11_2\jre\lib\security
5. Copy "IntegrityClientSite.rc" to C:\Program Files (x86)\Integrity\ILMClient11_2
 - a. Replace file in location
6. Restart system
7. Open 11.2 using the BAT file > select Import view set> verify host is correct server [magna-alm-prod2.ptcmscloud.com]> select first 4
8. In preferences Set 4 connection tabs with correct host/server name [magna-alm-prod2.ptcmscloud.com] and user's password

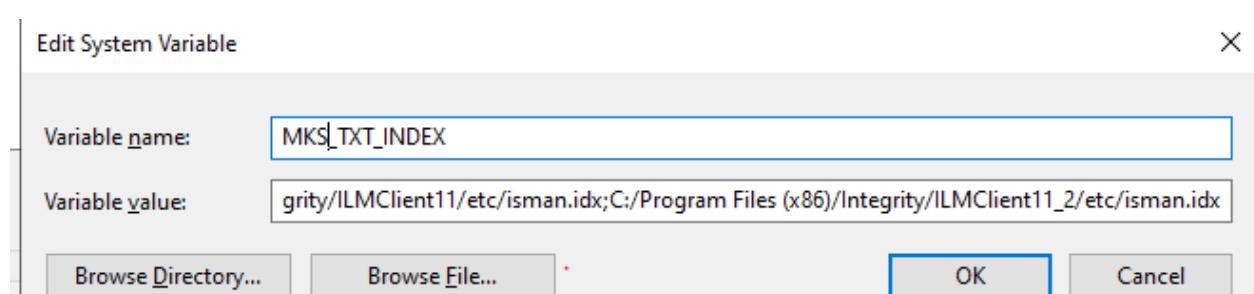




Setting the Environment Path.



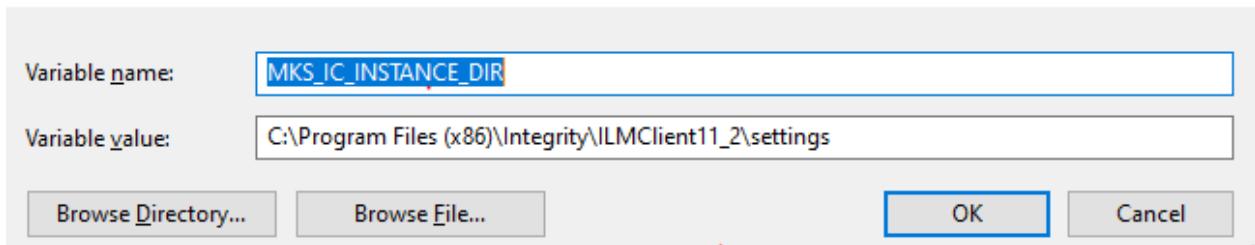
Variable name : MKS_INDEX
 Path : C:/Program Files (x86)/Integrity/ILMClient11/etc/isman.idx;C:/Program Files (x86)/Integrity/ILMClient11_2/etc/isman.idx



Variable name : MKS_TXT_INDEX
 Variable value : C:/Program Files (x86)/Integrity/ILMClient11/etc/isman.idx;C:/Program Files (x86)/Integrity/ILMClient11_2/etc/isman.idx

Edit System Variable

X



Variable name : MKS_IC_INSTANCE_DIR

Variable value : C:\Program Files (x86)\Integrity\ILMClient11_2\settings

Variable name: path

Variable path:: C:\Program Files (x86)\Integrity\ILMClient11_2\bin

Created with the Personal Edition of HelpNDoc: [Produce online help for Qt applications](#)

3.2 Executable file

Through an executable file, by just double-clicking on the “MITE.exe” icon, MITE application can be installed on System successfully



Created with the Personal Edition of HelpNDoc: [Easily create Qt Help files](#)

4 MITE Home Window

As soon as MITE launches through respective executable file “MITE.exe”, the first window will be “MITE Home Window” as shown in below figure,

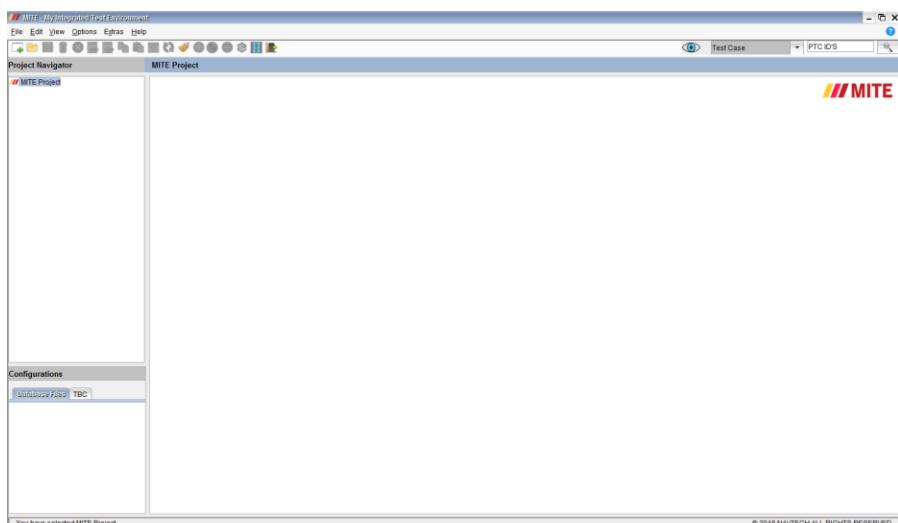


Figure 2: MITE Home Window

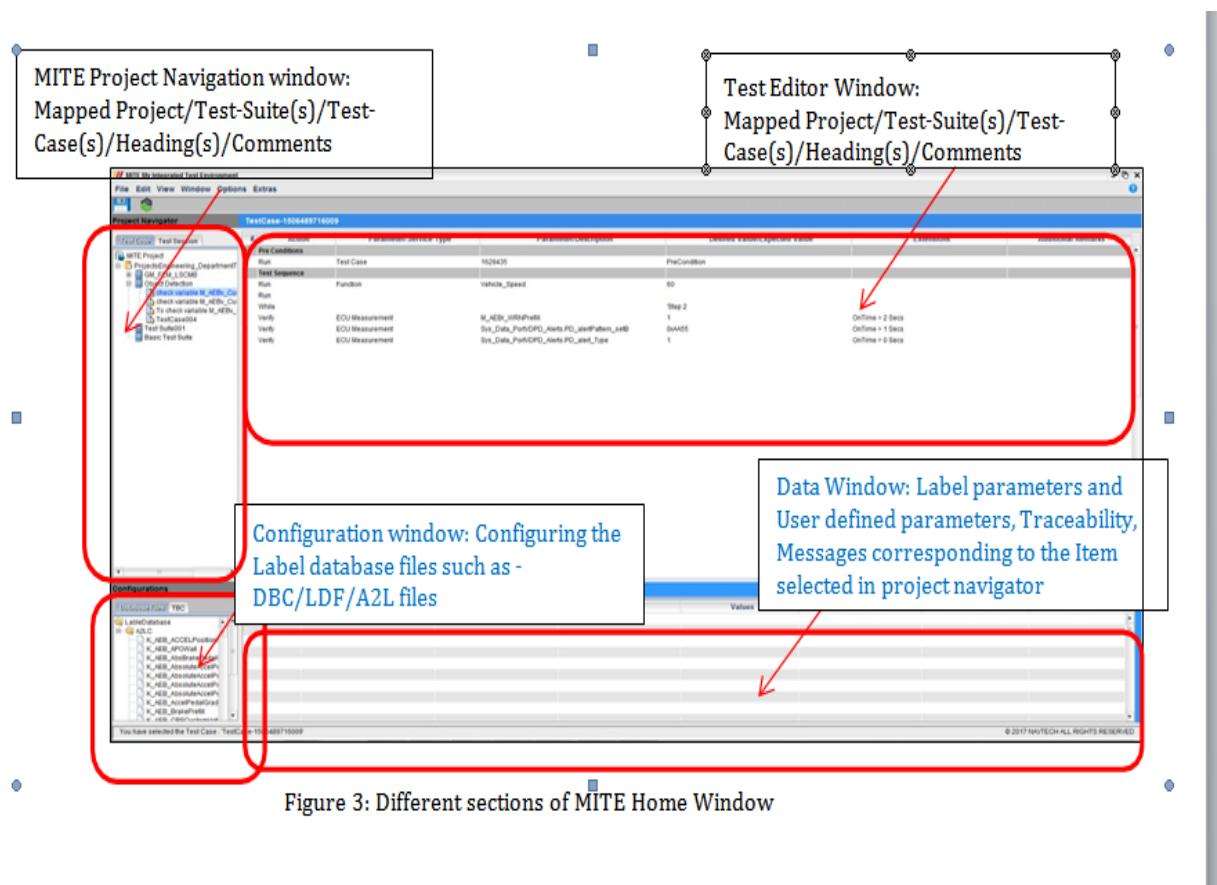


Figure 3: Different sections of MITE Home Window

Created with the Personal Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

5 Mapping a Project

To author Test-Cases in MITE user needs to map a PTC project into MITE. This is to

synchronize the data between PTC and MITE.
This can be done by using “Add/Map a Project” option.

Steps to Map a project:

1. Select MITE Project (on left most corner of the MITE home window)
2. Right Click and select the option – “ Map a Project”

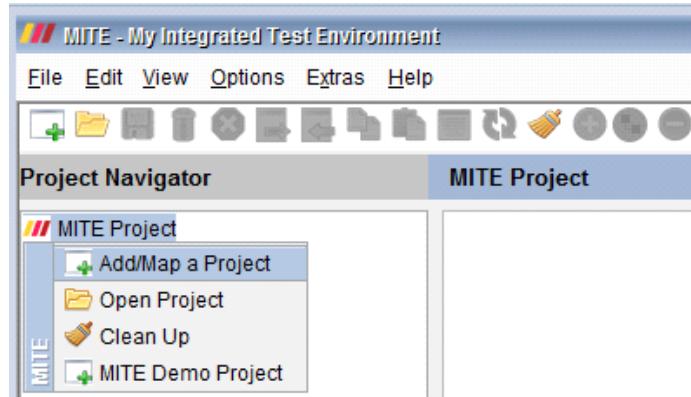


Figure 4 : Map a Project

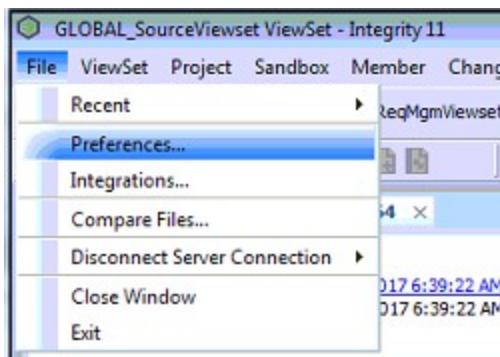
The options available on Right click on “MITE Project” are as below:

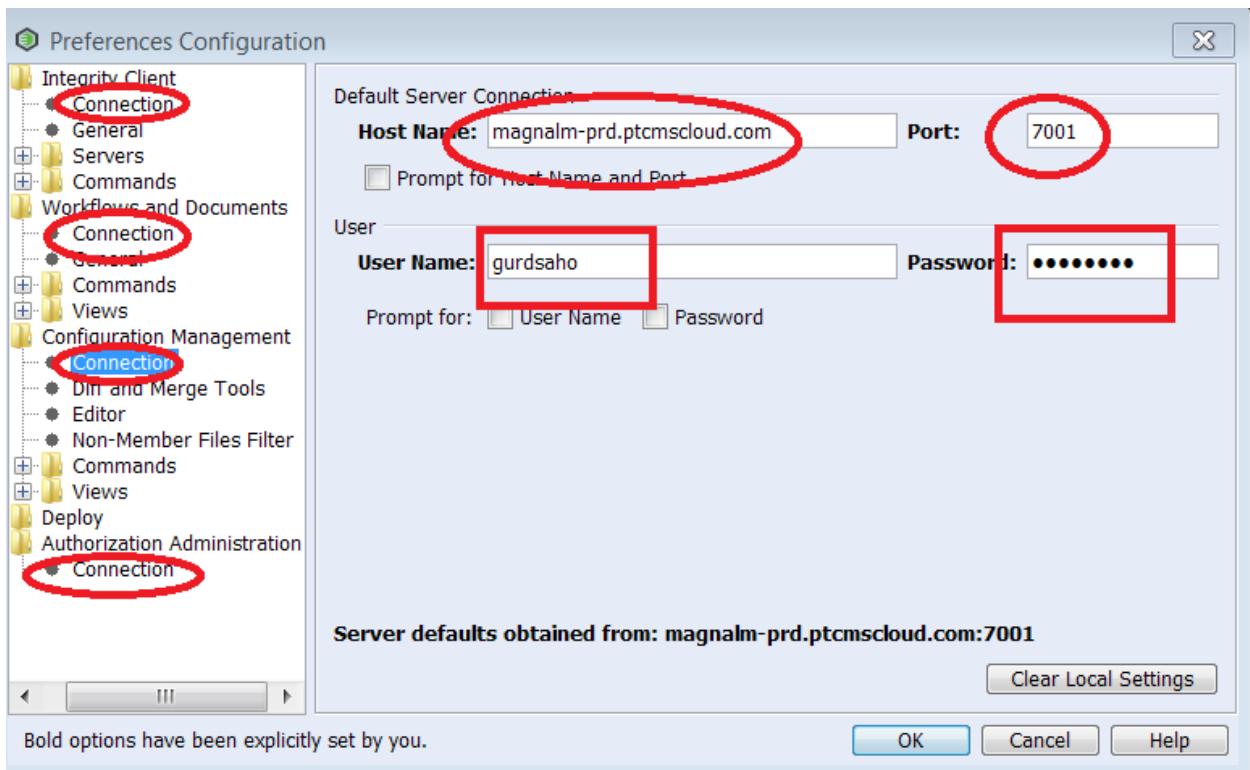
Select on Item	Options available on Right click	Operation performed
Mite Project	Add/Map a Project	For mapping a project into MITE from PTC server
	Open Project	To open a project which was closed by the user using – Close Project option only
	Clean up	For removing the deleted items of the project(s) from back end folders
	MITE Demo Project	Demo project for the First time Users

Table 1: Right click options on “MITE Project

NOTES:

1. *PTC client should be installed in the system already*
2. *PTC client should be open and running*
3. *Should have Host Name and Port Name filled in all the required areas in File→ Preferences; as shown below*
4. *Environmental path settings : “C:\Program Files (x86)\Integrity\LMClient11\bin”*





Created with the Personal Edition of HelpNDoc: [Free CHM Help documentation generator](#)

5.1 Mapped Project Overview and Summary

The information about the project will be available by selecting/clicking on the required project as shown in Figure below:

#	Test Suite ID	Type	Summary	State	MITE Compatibility
1	1975393	Suite	RVC_Level2	Open	True
2	1975386	Suite	Copy-paste test suite	Open	True
3	1975388	Suite	Param	Open	False
4	1973112	Suite	Testsuite01_modified	Open	False
5	1973230	Suite	Ts01	Open	False
6	1973318	Suite	sample	Open	False
7	1973327	Suite	sdf	Open	False
8	1974156	Suite	Test Suite 7_12	Open	True
9	1974159	Suite	Test Suite 7_12	Open	True
10	1974203	Suite	Test Suite 7_12	Open	True
11	1974220	Suite	Test Suite 8_12	Open	True
12	1975375	Suite	date10012018 latest	Open	True

Figure 5: Mapped Project details

- It provides the details about all the Test-Suite(s) available in the selected project
- Test-Suite(s) with ID :** Test-Suite(s) readily available in PTC integrity
- Test-Suite(s) without ID :** The newly added Test Suite(s) and the ones which are under development and not yet flushed into PTC

- iv. **State:** It provides the Test Suite(s) current work-flow state
- v. **MITE Compatibility:** It provides the information about the Test-Suite(s) compatibility i.e., whether the respective Test-Suite(s) are in MITE format or not. Its values are “True” or “False” respectively

Created with the Personal Edition of HelpNDoc: [Full-featured Help generator](#)

5.2 Project Summary and Sub-tabs

The complete information about the imported Project will be available by selecting the Project as shown in Figure:

- i. It provides the details about all the Test-Suite(s) available in the selected Project
- ii. **Test Suite with ID :** This indicates that it has already been flushed into PTC
- iii. **Test Suite without ID :** This indicates that it is a newly added Test -Suite or the one which is not yet flushed into PTC
- iv. **Created by :** It provides Name , Time and Date details of the Test Suite created
- v. **Modified by:** It provides Name , Time and Date details of when the Test Suite was modified at the latest

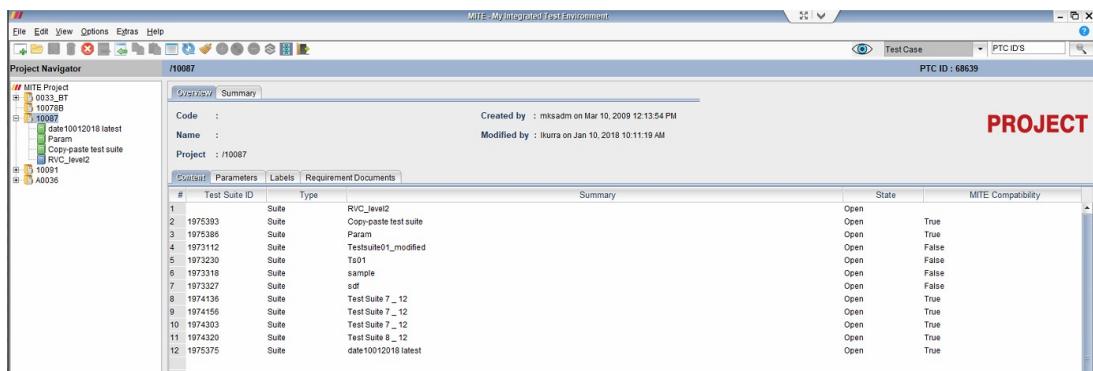


Figure 6: Project details window and Sub-Tabs

- vi. **Sub-Tabs :** The below is the list of sub-tabs available for Project(s).

 - α. **Content :-** Provides all the available Test-Suite(s)
 - β. **Parameters :-** Provides the consolidated list of User-Define parameters which were used by the user while authoring Test-Case(s) through MITE editor
 - γ. **Labels :-** Provides the consolidated list of Label data base parameters which were used by the user while authoring Test-Case(s) through MITE editor
 - δ. **Requirement Documents :-** Provides the list of requirement documents used in the project

Created with the Personal Edition of HelpNDoc: [Easily create iPhone documentation](#)

5.3 Options available on Project -Right click

The options available on Right click on “Mapped Project” are as below:

Select on Item	Options available on Right click	Operation performed
----------------	----------------------------------	---------------------

Project	View More Details	For project details
	Create Test Suite	For adding a new Test-Suite
	Open Test Suite	Open a closed Test Suite
	Import Test Suite	Test suite(s) which are exported into Zip file can be imported
	Recover Test Suite	To recover the deleted Test-Suite
	Reload Project	It reloads the Test Suite(s) of the selected project from PTC on to the Project Summary table
	Paste Test Suite	To paste the copied Test-Suite(this option will enable only after test suite copy)
	Close Project	To Close the current project

Table 2: Right click options on “Mapped Project”

Created with the Personal Edition of HelpNDoc: [Free PDF documentation generator](#)

5.4 View More Details Option

The View more details option is available at each and every level which provides the detailed information. This can be viewed using Right-Click option, as explained below:

Item	Operation performed	Brief Explanation
Project	Right click on Project – View More Details	For Project “Description” is editable
Test Suite	Right click on Test Suite – View More Details	For Test Suite “Short Name and Description” are editable
Test Case	Right click on Test Case – View More Details	For Test Case “Description” is editable
Heading	Right click on Heading – View More Details	For Heading “Description” is editable
Comments	Right click on Comments – View More Details	For Comments “Description” is editable

Table 3: View More Details

Created with the Personal Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

5.5 Reload Project

Reload Project from PTC: - It reloads all the available Test Suite(s) of the selected project from PTC and updates the Project Summary table. Using this option user will be able to receive the updated information or data with respect to the selected Project done by other user(s).

To reload project: Right Click on the project (from Project Navigator) – Select the option “Reload Project”, this will update the list of Test-Suite(s) available within the selected project in Project Content table.

Created with the Personal Edition of HelpNDoc: [Easy EBook and documentation generator](#)

6 Create Test Suite

Only after mapping a project, a Test suite can be added/created

into MITE Project. Steps to Create Test Suite:

1. Right Click on the mapped Project
2. Select option – “Create Test Suite”



Figure 7: Create Test Suite Option

Created with the Personal Edition of HelpNDoc: [Full-featured EBook editor](#)

6.1 Fill Test-Suite particulars

For creating a Test Suite successfully the below fields are require to be filled and then click on “OK” button.

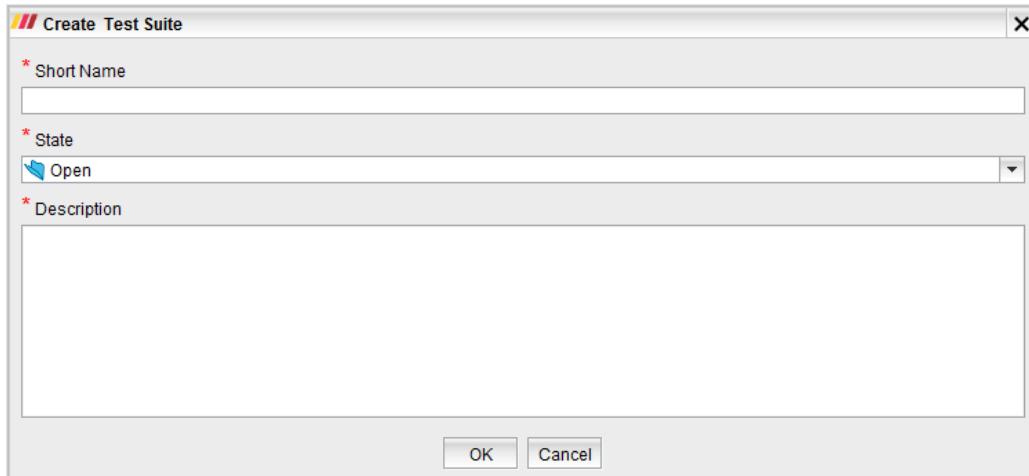


Figure 8: Test-Suite particulars

On adding Item	Fields	Operation performed
Test Suite	Short Name	Test-Suite Short Name
	State	Open by default for new Test-Suite
	Description	Description of the Test-Suite

Table 4: Test-Suite(s) particulars

Created with the Personal Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

6.2 Test Suite Summary and Sub-tabs

The complete information about the selected Test-Suite will be available by clicking on the required Test- Suite as shown in Figure 8:

- I. It provides the details about all the Test-Case(s) available in the selected Test-Suite
- II. **Test Suite with ID :** This indicates that it has already been flushed into PTC
- III. **Test Suite without ID :** This indicates that it is a newly added Test -Suite or the one which is not yet flushed into PTC

- IV. **Created by :** It provides Name , Time and Date details of the Test Suite created
 V. **Modified by:** It provides Name , Time and Date details of when the Test Suite was modified at the last



Figure 9: Test Suite details window and Sub-Tabs

- i. **Sub-Tabs :** The below is the list of sub-tabs available for Test-Suite(s)

- α. **Summary :- Provides all the available Test-Case(s) inside the selected Test-Suite**
- β. **Parameters :-** Provides the list of User-Define parameters which were used by the user while authoring Test-Case(s) through MITE editor
- γ. **Labels :-** Provides the list of Label data base parameters which were used by the user while authoring Test-Case(s) through MITE editor
- δ. **Messages :-Provides information about the errors/Warnings/information occurred while Test-Suite flushing into PTC**

- Errors – List the errors occurred while flushing the Suite with respective Short Name of the Test-Case(s). Until errors are resolved the Test-Suite(s) will not be flushed into PTC.
- Warnings – List the warnings occurred while flushing takes place. With un-resolved warnings a Test-Suite cannot be successfully flushed into PTC.
- Information – On successful flushing of Test-Suite(s) only, the message “Test Suite flushed successfully “will be displayed in this sub-tab, which is otherwise empty.

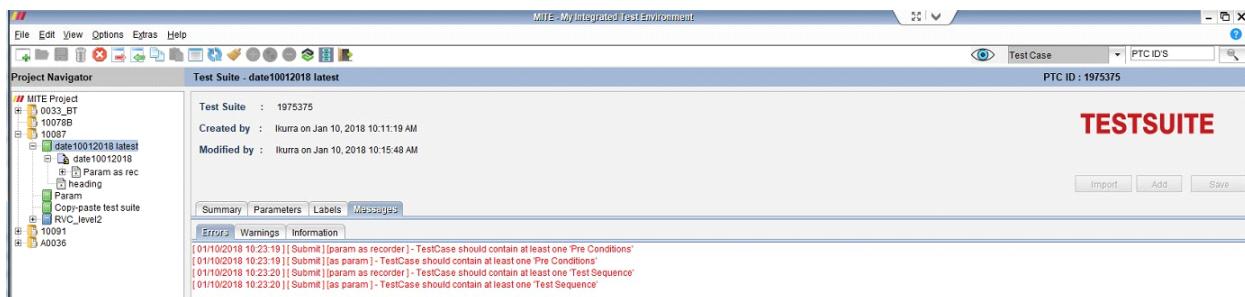


Figure 10: Test Suite Messages – Errors Sub-Tab

6.3 Options available on Test Suite -Right click

The options available on Right click on “Test Suite” are as below:

Select on Item	Options available on Right click	Operation performed
Test Suite	View More Details	For Test Suite detailed Information
	Create Test Case	For adding/creating a new Test-Case
	Import Test Case	The selected Test Suite will imported from Zip file
	Recover Test Case	The selected Test Case will be Recovered in same/another Test Suite
	Copy Test Suite	To Copy the Test-Suite
	Paste Test Suite	To Paste the Copied Test Suite
	Export Test Suite	The selected Test Suite will exported as a .Zip file
	Delete Test Suite	The Selected Test Suite can be Deleted. This option will appear only for Test-Suite(s) without PTC IDs
	Close Test Suite	The Selected Test Suite can be Closed
	Resync Test Suite	It takes Test Suite updates from the PTC for the selected test suite and updates the Test suite Summary table (this option is available only for the Test suite with ID)

Table 5: Right click options on “Test Suite”

6.4 Delete - Recover operations on Test-Suite

Delete - Recover options are available in right-click options only. A deleted Test-Suite can only be added back into the respective tree by using “Recover” option only (NOT by Load Test-Suite).[For more information Please refer to section 14 Delete/Recovery Operations]

Delete - Recover operations can be performed at two levels : Test-Suite and Test-Case levels

Test-Suite Delete:

To delete a Test-Suite, select the respective Test-Suite from its respective project folder/tree. Right-Click and click on “Delete Test-Suite”, Test-Suite will be deleted from the project.

On successful deletion, the project folder/tree and its summary will perform a refresh operation which will update the Project tree and its summary table accordingly.

Test-Suite Recover:

To recover a Test-Suite, select the respective Project folder. Right-Click and click on

"Recover Test-Suite", a dialog box will appear with the list of recently deleted Test-Suite(s) select the require Test-Suite and click OK. Test-Suite will be added to the project tree. On successful recovery, the project folder/tree and its summary will perform a refresh operation which will update the Project tree and its summary table according to the latest tree structure.

Created with the Personal Edition of HelpNDoc: [Free Qt Help documentation generator](#)

6.5 Load Test Suite

Load Test suite from PTC: - Any Test-Suite(s) from PTC MITE compatible True or False can be loaded from PTC using "Load Test Suite" option as shown below:

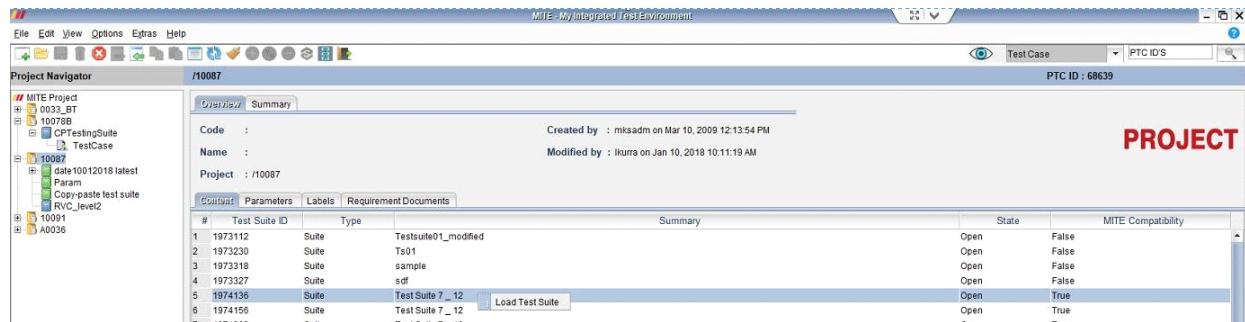


Figure 11: List of MITE Compatible True and False Test-Suite

Steps to follow to Load a Test-Suite : with MITE Compatibility as True

1. Click on the project (from Project Navigator) – This will give the list of Test-Suite(s) available within the selected project in Project Content table, Test-Suite(s) with MITE Compatibility True can only be loaded and re-submit into PTC
2. Select the Test-Suite (from the Project Summary table) – Right-click on the Test-Suite , "Load Test-Suite" option appears. Click on – Load Test-Suite, load operation will execute and Test-Suite will be added to the Project tree.
3. Can edit the loaded Test-Suite(s) as per user requirements and flush/submit again into PTC and PTC Id(s) will remain same as before for former Test-Case(s) under it.
 - For the Test-Suite(s) which are already present in the Project tree (under Project Navigator window) will not have the "Load Test-Suite" option.
 - Test-Suite with MITE Compatibility as False can also be loaded just as a view to the user. User cannot submit this test suite into PTC.

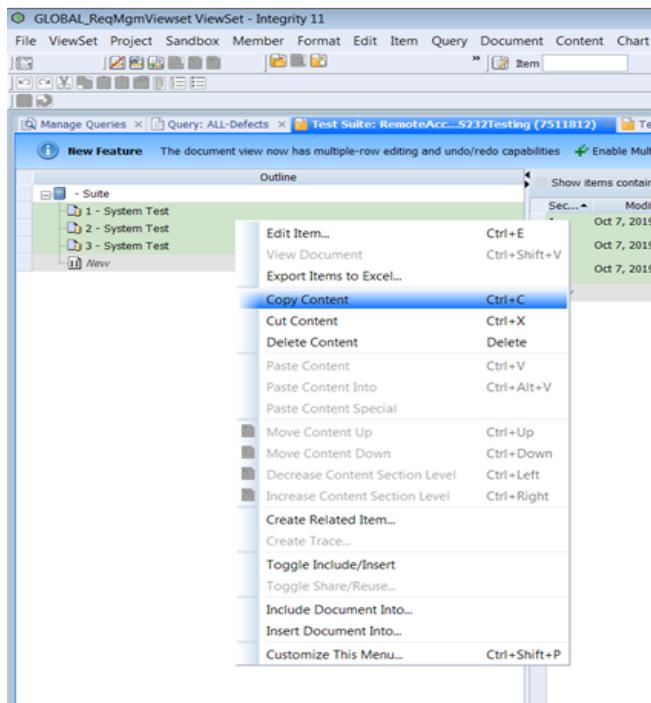
Created with the Personal Edition of HelpNDoc: [Easily create iPhone documentation](#)

6.6 Shared/Reuse Test suite

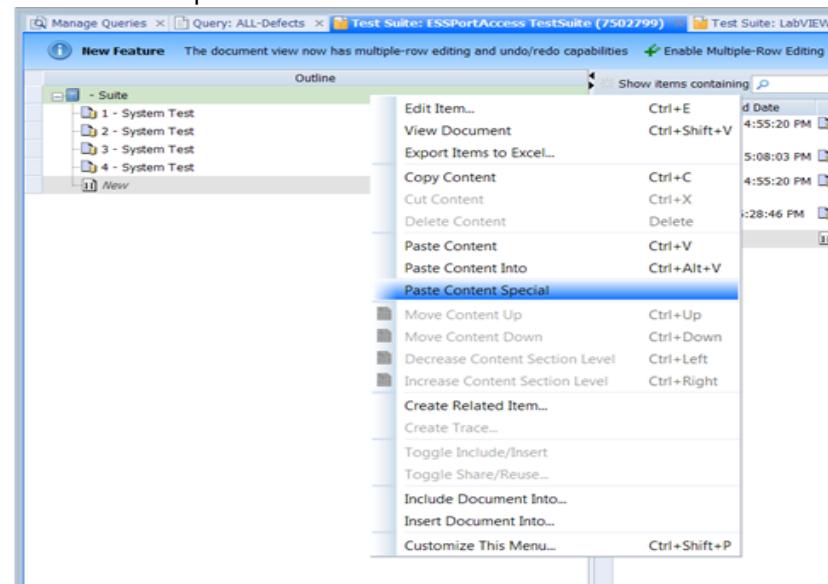
Branching is a method of reusing existing TestCases Content as starting points for future usage.

Step 1:

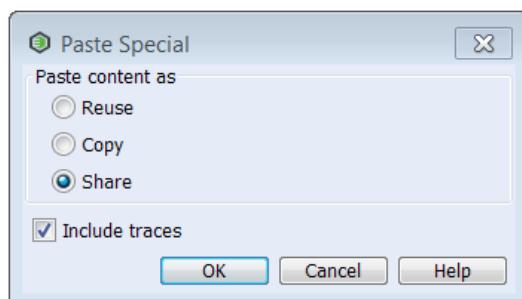
Select Parent TestSuite's TestCases data which is required to be shared and then Right click→ select Copy Content option in PTC.



Step 2: Select Child TestSuite where data is required to be pasted → Select TestSuite → Right click → Paste Content Special.

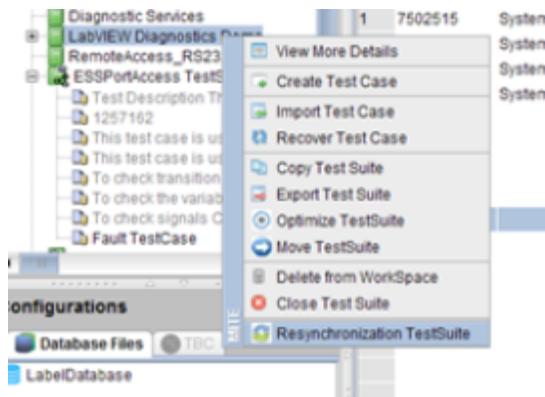


Step 3: After getting pop - up select share option as shown below. Till this step all the actions should be done in PTC.



When we select content as Share option then the shared TestCase will be in synchronization with source TestCase and when we select content as Reuse option then there will not be any synchronization with source TestCase.

Step 4: Open MITE → Load Child TestSuite from the Project → Select Child Suite → Right click → Resync TestSuite



Step 5: After Resync we can observe Share symbol for that TestSuite in MITE as shown below:

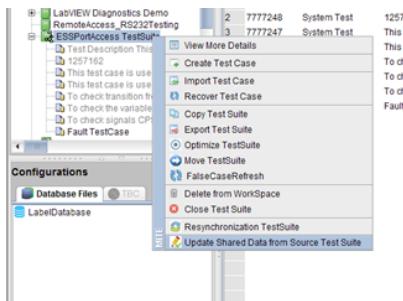
#	Test Case	Description	Status	Share	Valid	Priority
1	777...	System... Test Description This test case is used to check the preconditions...	reviewed	750...	Share	
2	777...	System... 1257162	propos...	750...	Share	High
3	777...	System... This test case is used to verify combination of programming sessi...	reviewed	750...	Share	
4	777...	System... This test case is used to check the Negative Response Code - N...	propos...	750...	Share	
5	777...	System... To check transition from ENABLED State to OFF State when FCM i...	propos...	750...	Share	
6	777...	System... To check the variables in ENABLED State	propos...	750...	Share	
7	777...	System... To check signals CPS_Collision_Preparation_IO.CollPrSysCntrSt...	propos...	750...	Share	
8	750...	Fault TestCase	draft		author	High

Step 6: In Shared TestSuite, shared TestCases editor is not editable.

#	Test Sequence
1	
2	
3	
4	
5	
6	
7	

#	Post Conditions
1	
2	
3	
4	
5	
6	
7	

Step 7: User need to select (Select Shared Child TestSuite → Right click→Update shared data from source test suite) Option to get updated data from PTC every time when there is updated data in Parent TestSuite from the PTC.



Step 8: After taking update please flush the TestSuite to PTC by clicking on the PTC Submit button.



In this shared concept, updating the shared Test Case from both PTC and Source TestCase is not possible. Basically, the order of steps to be followed for shared TesSuites is:

- 1) Load Test Suite as shared.
- 2) Re Synchronize the shared TestSuite.
- 3) Update the shared TestSuite
- 4) Flush the shared Test suite into the PTC.

So, with these steps, the resync changes in the shared Test Case will be overwritten with the changes from the Parent Test Suite.

So, basically, update option is given priority over re synchronize for shared items.

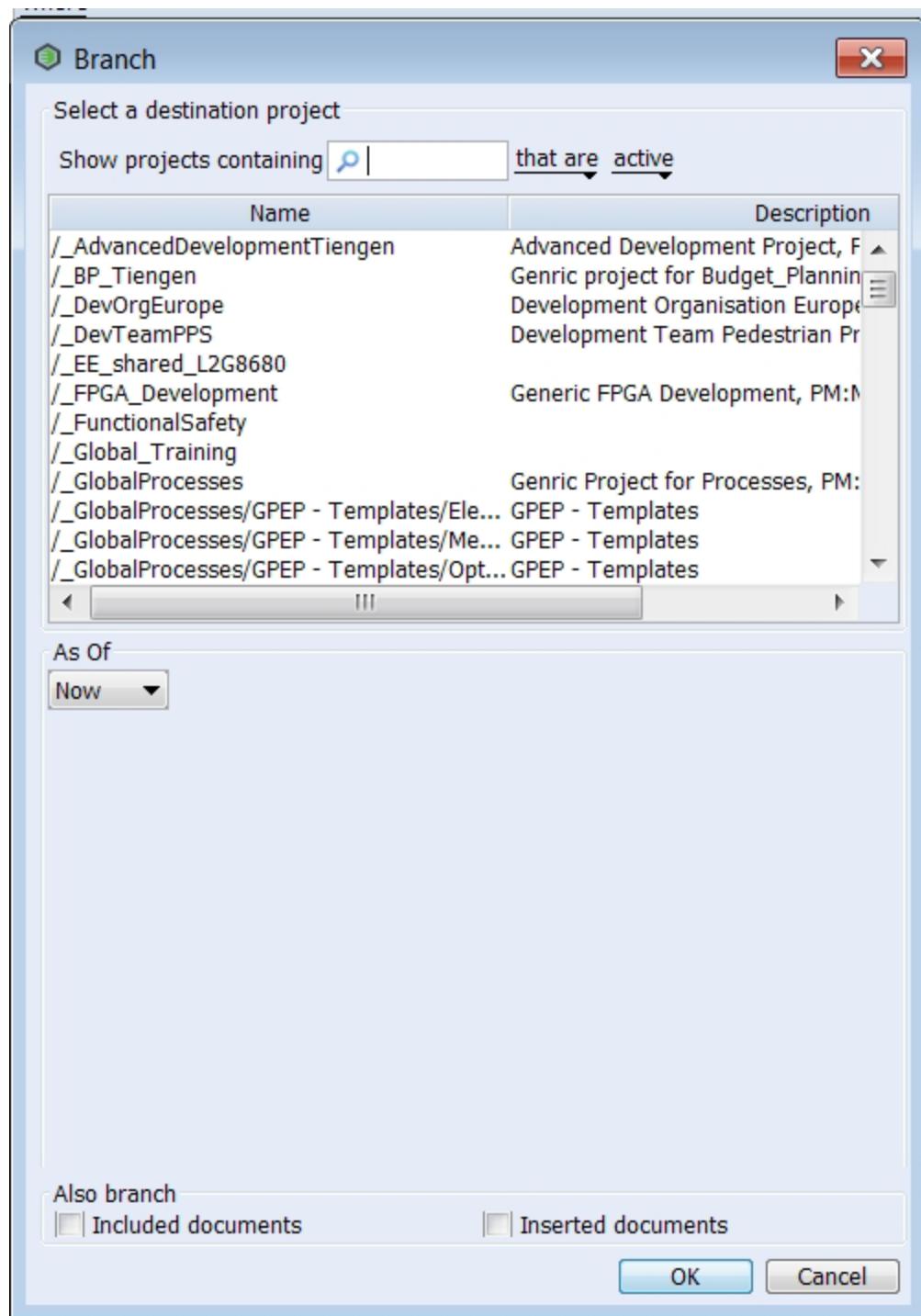
NOTE: If you modify Category or Description or Parameters of a Reuse TestCase then it will become Author TestCase.

Created with the Personal Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

6.7 Branching of a TestSuite in PTC

Branching of a TestSuite:

In PTC, to create an exact copy of existing test suite we have an option like branching the test suite. To Branch a test suite, open the test suite that needs to be branched in the document view. Now, click on document menu in the file menu and select the option "Branch". This will show the list of projects. Select the project from the list to which the test suite has to be branched which is as shown below:



After selecting the project and clicking on OK, exact replica of the test suite will be created in the selected project which is shown as below:

Test Suite: 8927442
 Branched from [8916312](#) by [Bharathi Kolla \(bharkoll\)](#) on Mar 27, 2020 1:33:33 PM against date [Mar 27, 2020 1:33:33 PM](#)
 Modified by [Bharathi Kolla \(bharkoll\)](#) on Mar 27, 2020 1:33:33 PM

The screenshot shows the MITE Test Suite configuration interface. The top navigation bar includes links for Properties, Parameters, Metrics, Advanced, Historical Data, Branches, Labels, Workflow, and History.

Summary: RVC Diagnostics
Short Name: RVC Diagnostics
Description: Sample
 [MITE Locations :/Playground/project.pj]
 [MITE TestSuite :TestSuite-1585128428009]
 {CurrentTime:03/25/2020 15:23:09}

Process:
Category: Suite
Project: /Projects/Engineering_Department/Tools/MITE (Out of Phase)
State: Open (Open)
Revision Date: Mar 27, 2020 1:33:33 PM
Valid Change Order: true

Options

Now, the reference mode for these items will be "Reuse" as shown below

The screenshot shows the MITE Test Item List interface. The left sidebar displays an outline of the test suite structure, including a heading named "1 - Heading". The main area is a table listing test items:

Section	Category	Reference Mode	Text	ID	Document ID
1	Heading	Reuse	SID \$11: ECU Reset	8927444	8927442
1.1	System Test	Reuse	To verify ECU Reset supported sessions and suppress format	8927445	8927442
1.2	System Test	Reuse	To verify Hard Reset shall be implemented to emulate a power cycle.	8927446	8927442
1.3	System Test	Reuse	To verify SID11 Does Not Support functional addressing	8927447	8927442
1.4	System Test	Reuse	To verify After a reset, Camera shall start in default session	8927448	8927442
1.5	System Test	Reuse	To check defined reset time shall represent an upper limit. Camera shall always reset as soon as possible.	8927443	8927442
New	Heading	Author	New	8927442	

Now, there is another important step that needs to be followed before loading the branched test suite into MITE.

Click on the edit item of the branched test suite. In the description field, there will be marked label created by MITE which defines the path of the sandbox. Refer the image below:

Summary RVC Diagnostics

Short Name RVC Diagnostics

Description Sample

[MITE Locations ./Playground/project.pj]
 [MITE TestSuite :TestSuite-1585128428009]
 {CurrentTime: 03/25/2020 15:23:09}

The text which is highlighted in the above image is the "Marked Label". So, one should remove this marked label from the test suite.

Now, the test suite can be successfully loaded into the MITE. Load the Test Suite from the PTC using the option

"Load Test Suite" as shown below:

Project Navigator

- MITE Project
- L2G8680
- ProjectsEngineering_DepartmentToolsMITE
- L2H0090
- L2G8810

L2G8680 PTC ID : 2452288

PROJECT

#	ID	Type	Summary	State	MITE Compatibility
1	11956566	Test Suite	L2G8680_BMW_ADCAM_SysTestSpec_FFT	Open	True
2	11936764	Test Suite	BBx SW Integration Tests	Open	False
3	11936733	Test Suite	L2G8680::ADCAM::Integration Test	Open	False
4	11816019	Test Suite	L2G8680_BMW_ADCAM_SWQualification_TestSpec_OD	Open	False
5	11793610	Test Suite	L2G8680_BMW_ADCAM_SWQualification_TestSpec_RMD	Open	False
6	11750046	Test Suite	Test Case Creation_uncovered ports_architecture	Open	False
7	11742063	Test Suite	EBIO Test Suit	Open	False
8	11742062	Test Suite	EBII Test Suit	Open	False
9	11624211	Test Suite	ADCAM::BaseSW::Exception Handler:: SW Qualification Test	Open	False
10	11520998	Test Suite	L2G8680_BMW_ADCAM_SysTestSpec_Degradation (Base SW) PART_2	Open	True
11	11466279	Test Suite	ADCAM:: CyberSecurity SWCs::E04 Secure Boot Manager::SW Component ...	Open	False
12	11465283	Test Suite	ADCAM:: CyberSecurity SWCs::MfgM::SW Component Test	Open	False
13	11464991	Test Suite	ADCAM:: CyberSecurity SWCs::SecureActivationManager::SW Component T...	Open	False
14	11464928	Test Suite	ADCAM:: CyberSecurity SWCs::RxSWINHandler::SW Component Test	Open	False
15	11464705	Test Suite	ADCAM:: CyberSecurity SWCs::CIapt2H::SW Component Test	Open	False
16	11441335	Test Suite	hsmandeeyq	Open	False
17	11424037	Test Suite	ADCAM::Cybersecurity:: SWQT	Open	False
18	11369783	Test Suite	L2G8680_BMW_ADCAM_System_Test_Spec_DM	Open	True

You have selected the Project : 'L2G8680'

© 2017 NAVTECH ALL RIGHTS RESERVED

The below image shows the shared test suite once it gets loaded into the PTC.

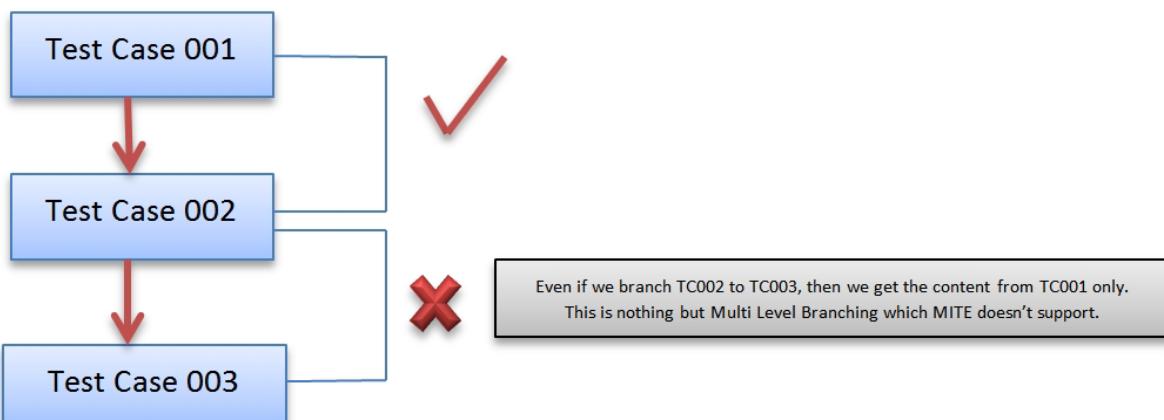
The screenshot shows the MITE software interface. On the left, there is a project tree with several branches under 'ProjectsEngineering_DepartmentToolsMITE'. One branch is 'RVC Diagnostics' which contains 'TestSuiteFlushing', 'Sample TestSuite', 'Copy/PasteForValidForProject', 'trueTestSuiteShared' (which is expanded to show 'Resync Testing', 'VMD', 'AdditionText_Test', 'TestingUI', and 'shortname'), 'GM_L2G5390_FPCM_SystemTestSuite_BenchTesting_MY2020_MITE' (which contains 'SampleMITESuite' and 'SampleSharedTestingBeforeDemo'), and 'PlaygroundEU'. Another branch is 'RVC Diagnosi' (note the extra 'i'). Below the project tree is a 'Configurations' section with tabs for 'Database Files' (selected) and 'TBC'. Under 'Database Files' is 'LabelDatabase'. On the right side of the interface is a large log window titled 'Errors', 'Warnings', and 'Information'. The log window displays a series of messages from January 2020, detailing the initialization of PTC authentication, sending ping requests, and various processing steps related to test suites and configurations.

Process for the versions below 2.8.1.0:

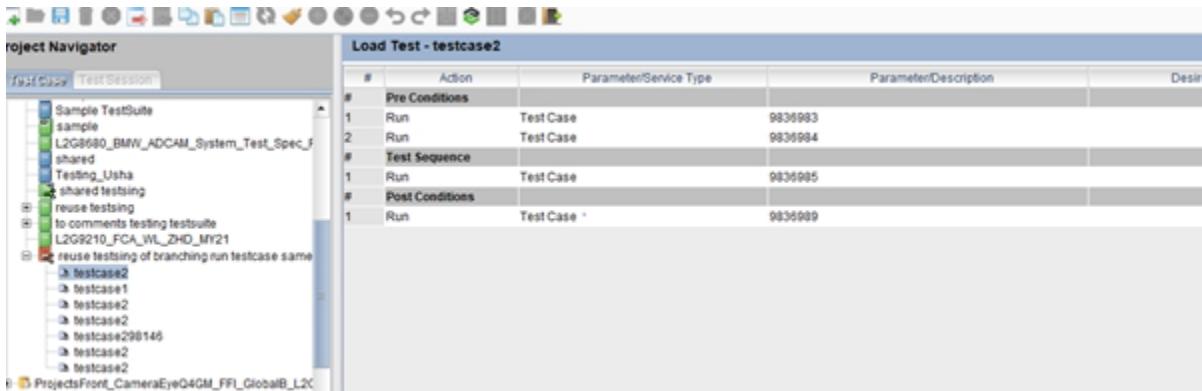
Once the test suite gets loaded as shared, then one can load the source data into the branched test suite also by selecting the option "**Update shared/Reuse data from source test suite**".

NOTE: In branching, when user clicks on "Update Data From Source Test suite", the data will be fetched to the shared test suite with reference to the root id. MITE considers the root id as the source id in the process of branching. This is explained in detail with the help of the following diagram.

MITE doesn't support Multi Level Branching.



As an example from the above example, if the TestCase 003 is updated, then the contents will be updated from the TestCase 001 but not from the TestCase 002. Similarly, from Version, 2.1.8.0, MITE supports updating of Test Case ID's in "Run TestCase...." related steps. But, these updates will also doesn't support Multi Level Branching.

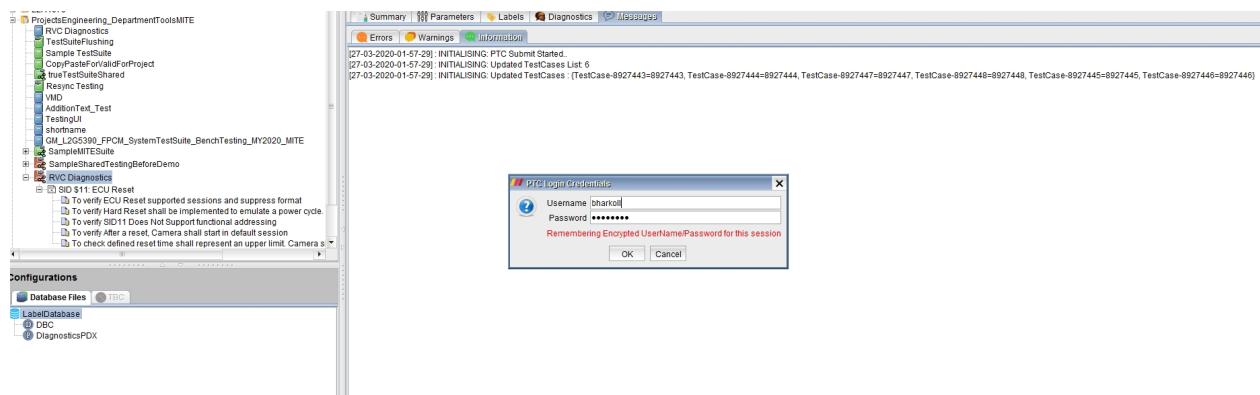


Refer the below images.

After updating the content from source test suite, all the reuse test cases will be auto enabled and they will act like

MITE compatible test cases, where as shared test cases will not be auto enabled. This is how the data from source Test suite gets loaded into the branched test suite in MITE.

The loaded shared Test suite is a non MITE compatible suite as the branching is done directly in the PTC.
To make MITE compatible test suite, right click and select "Enable Test suite" and then flush the test suite into



Process for the versions above 2.8.1.0:

If the version used is above 2.8.1.0, then after branching the suite in the PTC, the following steps has to be followed in the MITE:

- 1) Please make sure of flushing the source Test Suite with the version 2.8.1.0 or above.
- 2) Load the Test Suite that is branched in the PTC into MITE. The Test Cases gets loaded as MITE Compatible Test Cases but not the Test Suite.
- 3) Export the Label Data Base from the source Test Suite and import that into the destination Test Suite.
- 4) Optimize the Test Suite.
- 5) Resynchronize the Test Suite.
- 6) To make the Test Suite MITE Compatible, enable the Test Suite and Flush it.

Created with the Personal Edition of HelpNDoc: [iPhone web sites made easy](#)

6.8 Delete scenario for Flushed Test Suite

Implementation:-

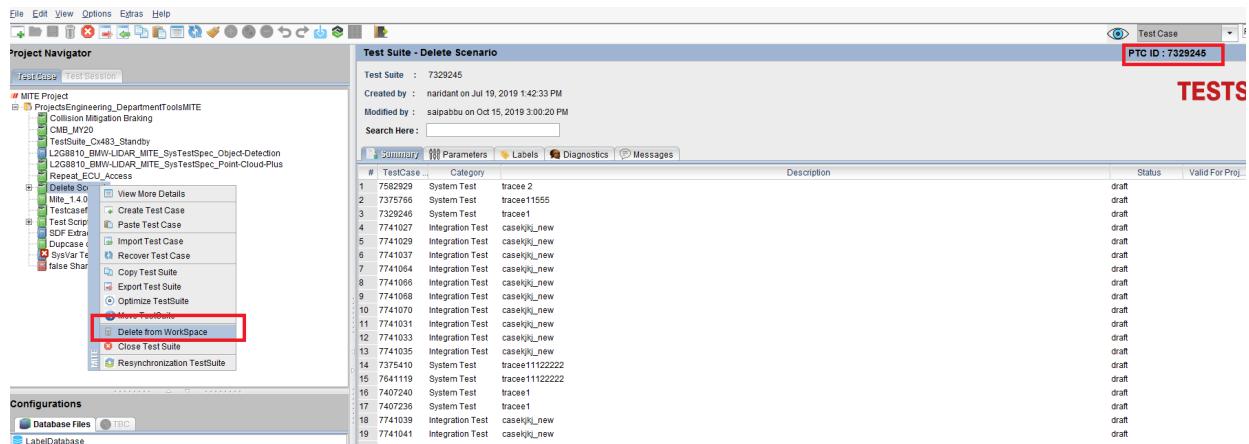
Previously if User wants to delete the Flushed Test Suite then we didn't have delete scenario as per business requirement.

Now we have new option i.e when we right click any particular flushed Test Suite, User can get Delete from Workspace option to delete that Test Suite.

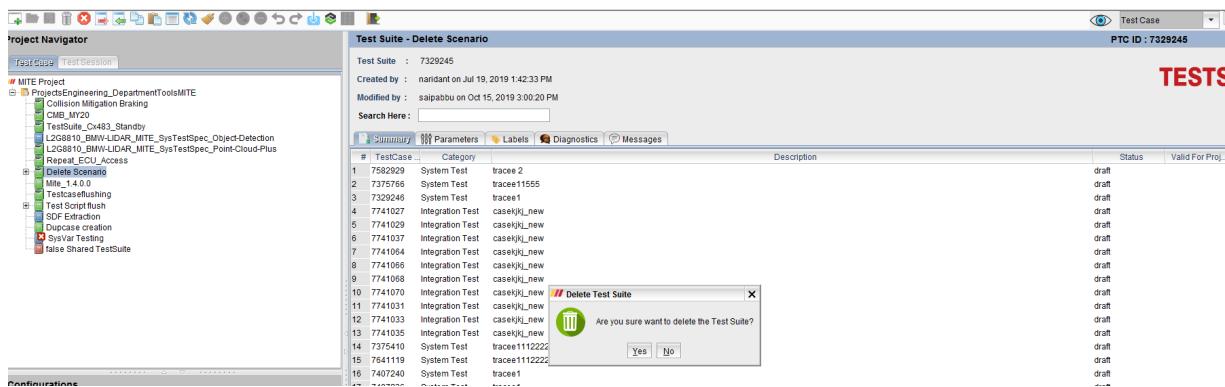
Scenarios:-

Step 1:-

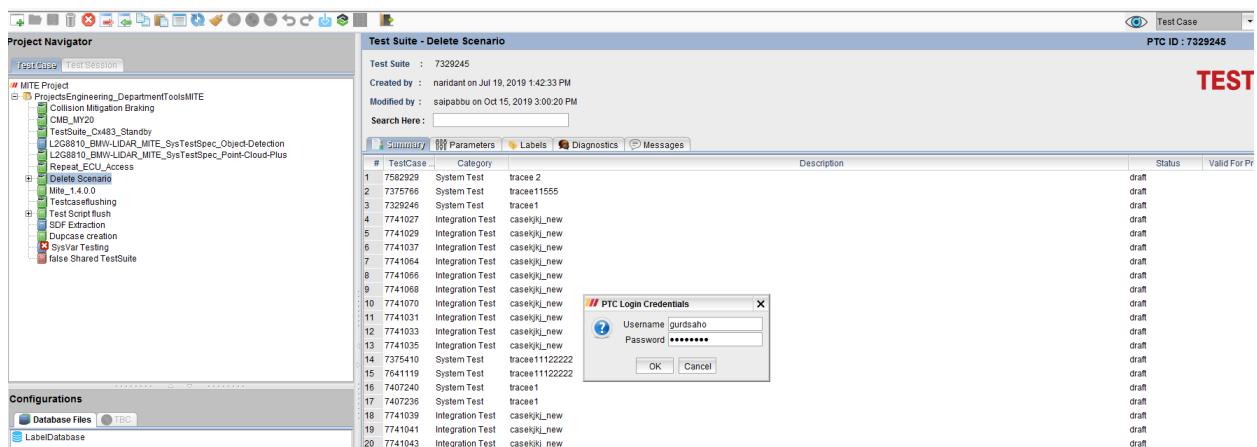
User can delete the flushed Test Suite only if it does not have any changes in it as shown below:



A Pop up will appear below to make sure User wants to delete the Test Suite or not:

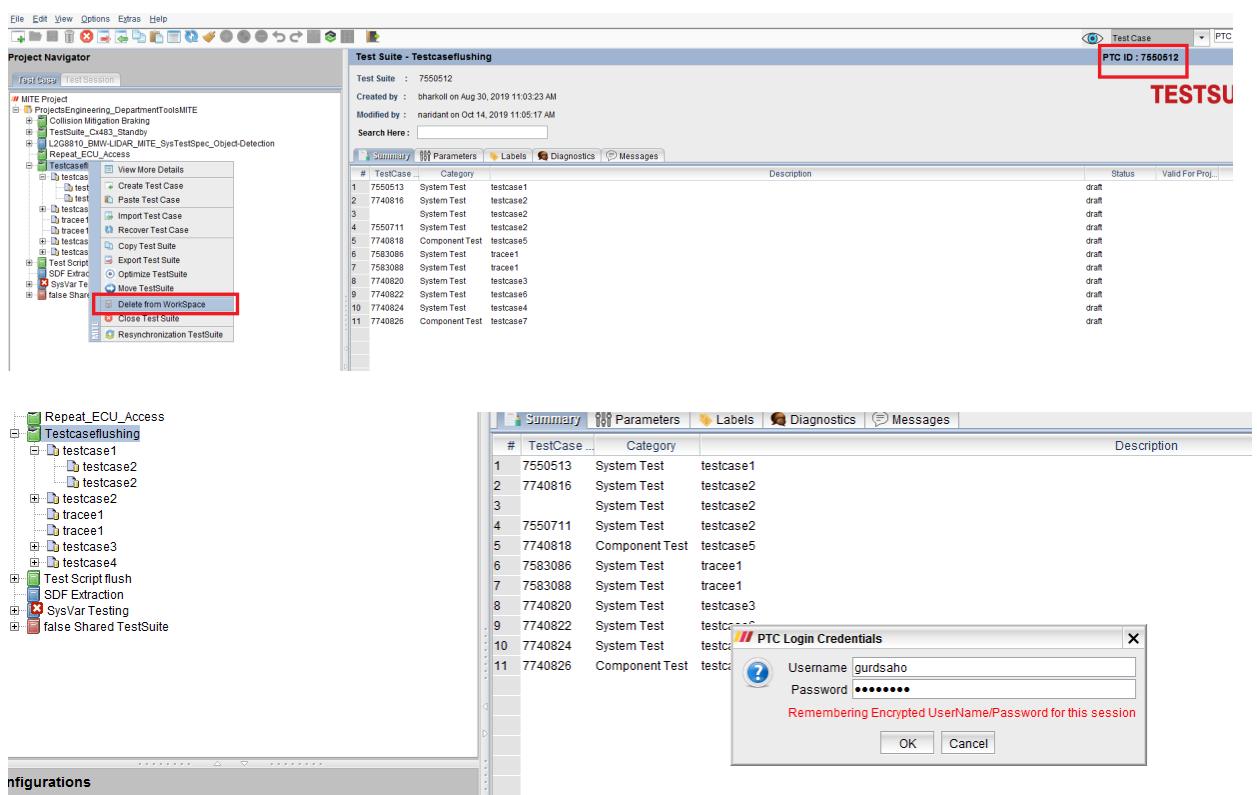


Then User will be asked to submit the PTC credentials as shown below:

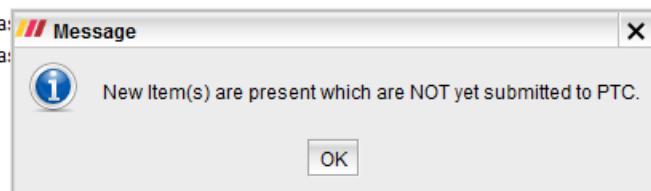


Scenario 2:-

If User wants to delete the Test Suite in which there is some changes then he/she is not allowed to delete the Test Suite as shown below:



3088	System Test	tracee1
0820	System Test	testcase3
0822	System Test	testcase6
0824	System Test	testcase7
0826	Component Test	testcase8



Created with the Personal Edition of HelpNDoc: Easily create Help documents

6.9 Open Test suite from Project summary Table

Open Test suite from project summary table functional

To invoke test suite details at right side we need to choose Test suite node from tree. But in recent enhancement on project summary table we can choose test suite from project summary table by using “open” pop-up menu .

To invoke “open” pop-up menu, we need to do right click option on single row and select Open popup menu.Then the selection will move to belonging test suite and right side panel will update with test suite details.

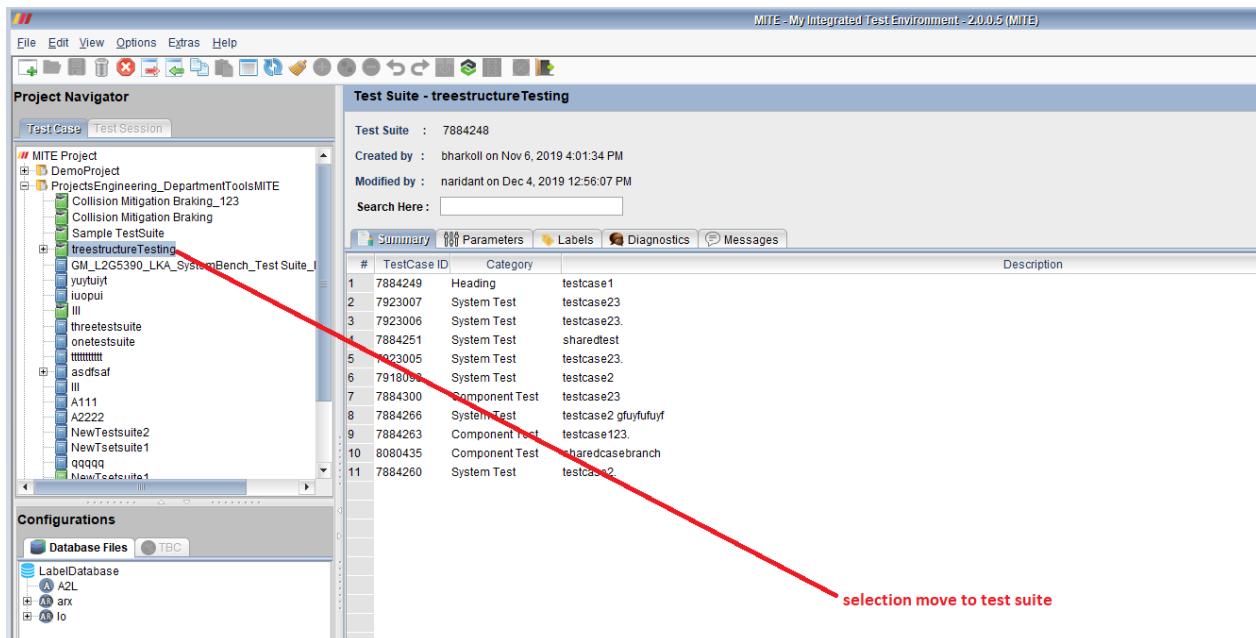
One more thing is, which suite we selected from project summary table, it must be there in belonging project in tree.

Otherwise “OPEN” pop-up menu never invoke for that selected row.

Invoke Open pop-up menu as like below:

#	ID	Type	Summary
1	7955014	Suite	TestSuite_MITE
2	7884248	Suite	treestructureTesting
3	Open	Suite	clicktest
4	7866895	Suite	CMB_MY20
5	7866865	Suite	Testing
6	7866839	Suite	sharedcasetesting
7	7794908	Suite	NewTestSuite JSON
8	7794292	Suite	Resync Testing
9	7790740	Suite	argsreg
10	7790733	Suite	testtwo
11	7790696	Suite	True Shared Testing
12	7789293	Suite	argsreg
13	7789880	Suite	aregtw
14	7789783	Suite	aregtw
15	7789719	Suite	argsreg
16	7789691	Suite	argsreg
17	7789680	Suite	CopyPasteForValidForProject
18	7789658	Suite	CopyPasteForValidForProject
19	7789619	Suite	True Shared Testing

After click on Open pop-up menu selection will move to test suite



Created with the Personal Edition of HelpNDoc: [Create iPhone web-based documentation](#)

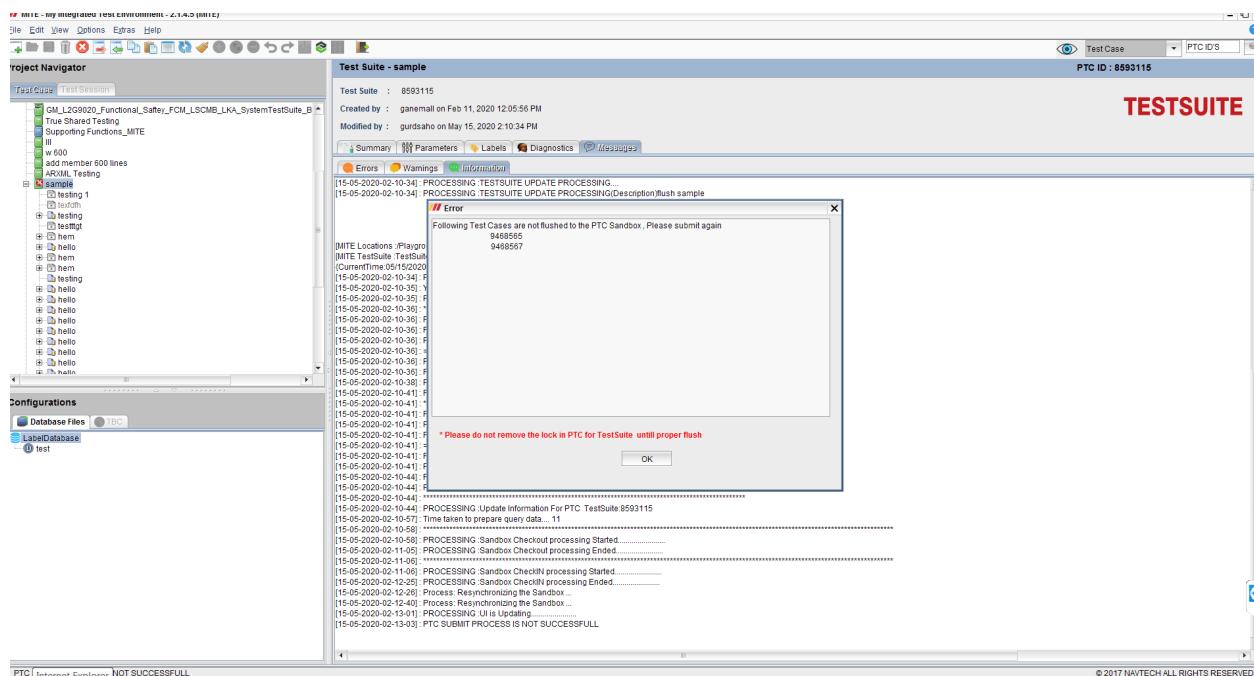
6.10 Add member pop up

Description:

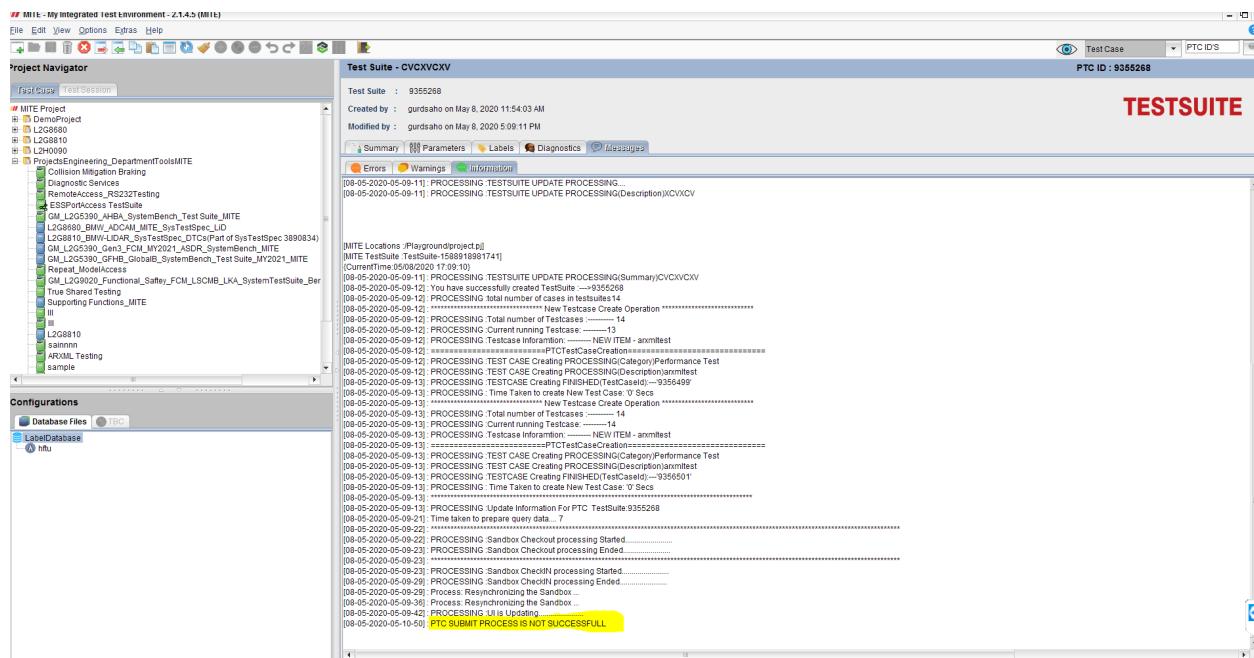
Previously when user is flushing the Test suite few of the Test cases could not go to sandbox but flushing operation was successful.

Now when user is flushing the Test suite pop up is shown if the Test cases doesn't went to sandbox user needs to flush the Test suite again.

Step 1: After flushing if any of Test cases is not flush properly which means it didn't went to sandbox user will get a pop up as below



Step :2 After clicking on Ok button user will get below message.



Step :3 User needs to do Flush operation again.

Created with the Personal Edition of HelpNDoc: Create iPhone web-based documentation

6.11 Baseline Test suite

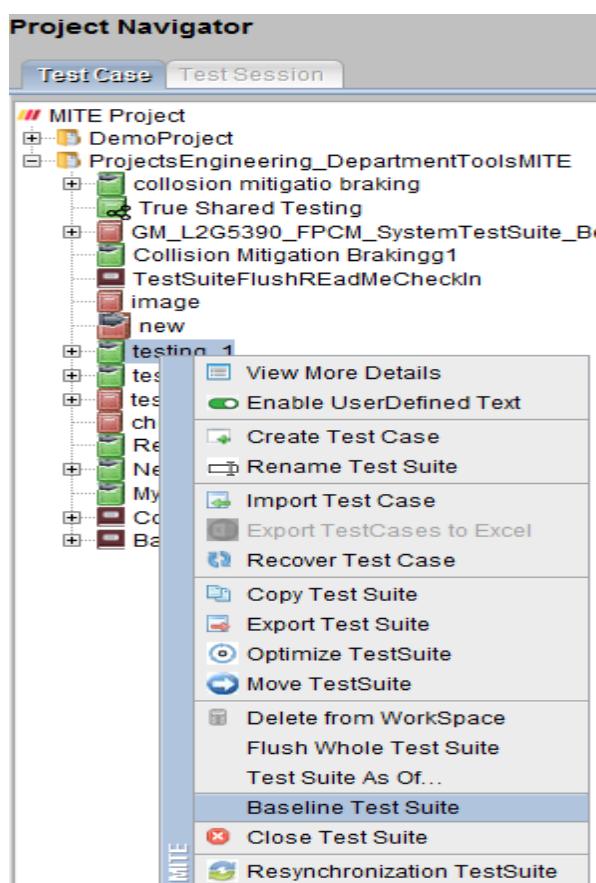
Baseline Test Suite

- A baseline is a special label that can only be added to the test suites. We can create a test suite baseline to mark an important project milestone. For example, we can create test suite baselines for a

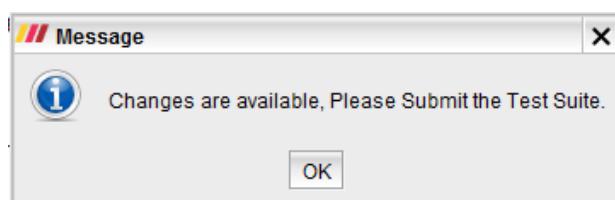
reviewed test suite, a reference test suite. We can use test suite baselines for internal verification and for recording external verification.

- To create the baseline, we need to select the individual test suite that is required to represent the information for our milestone. Each label contains the timestamp. **Baseline Test Suite Option** is appear in the right click of the test suite.
- We can later view the test suites included in the baseline by viewing the **Test Suite As Of** the baseline label. This allows us to view the test suites exactly how they existed when the baseline label was created; however we cannot make any edits to this historical view.
- **Baseline Test Suite Option** is available in test case role
- When a test suite's baseline is completed, the suite's symbol is altered to resemble a book.

1.1 Creating a test suite to Baseline Test Suite:

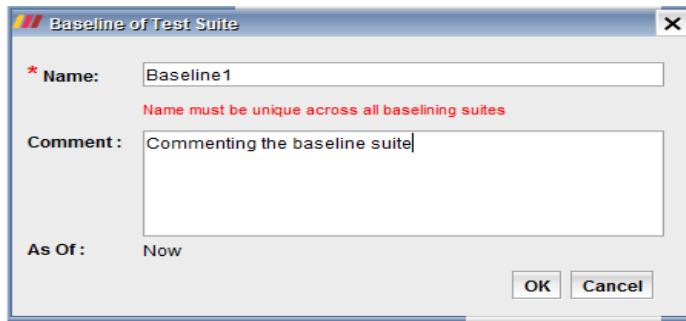


- To create a test suite to a baseline test suite, first choose the test suite to which the user wants to create it, then right-click on the test suite and select the Baseline Test Suite option. If there are any changes with regard to PTC, it will display a message stating that changes are available please submit to PTC.



1.2 Baseline Test Suite:

- Frame consists of the following fields: Name, Comment, As Of, OK, CANCEL

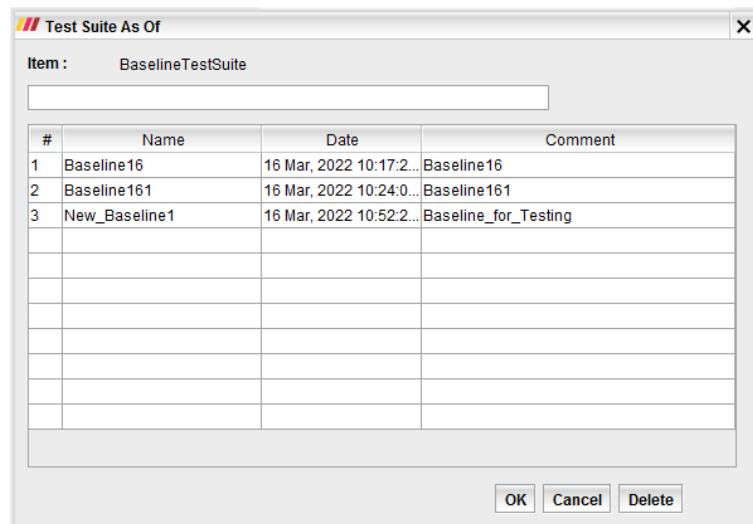


- *Name:** It is the field, whenever a user wants to create a baseline suite, he needs to give the label name in the baseline of test suite frame, and the condition is that the name should be unique for all baselining test suites and mandatory to be given.
- Comment:** It is the field where the user can provide comments to the baseline test suite.
- As Of:** The default option is Now, indicating the current date and time of the test suite
- OK:** When a user wants to save/accept changes after a certain operation, he needs to click the OK button
- Cancel:** When a user does not want to have any changes/accept changes after a particular operation, he needs to click the Cancel button.
- The user must first submit the both test case and test script role of the test suite to PTC before it can be baselined. After successful flushing the baseline test suite can be done, when user clicks the Baseline Test Suite option in MITE editor window the user can see the whole process of baseline. After successful completion of entire process a message will be shown as baselining of test suite is completed.

```
[14-03-2022-12-45-13] : INITIALISING: PTC Authentication Started...
[14-03-2022-12-45-13] : INITIALISING: Trying to establish the connection to the PTC
[14-03-2022-12-45-13] : INITIALISING: Filling the encrypted default username and password. These can be changed by the user at any time
[14-03-2022-12-45-13] : INITIALISING: Validating the user given PTC credentials
[14-03-2022-12-45-13] : INITIALISING: Trying to create a PTC instance based on the given user credentials
[14-03-2022-12-45-13] : INITIALISING: Setting the integrity user preferences
[14-03-2022-12-45-15] : INITIALISING: Integrity client connection is successfull
[14-03-2022-12-45-15] : INITIALISING: Encrypting the user given username and password
[14-03-2022-12-45-15] : INITIALISING: Connection to the PTC has been established successfully
[14-03-2022-12-45-15] : INITIALISING: PTC Authentication Successfull.
[14-03-2022-12-45-16] : INITIALISING: Making Baseline of the TestSuite Process is Started...
[14-03-2022-12-45-16] : PROCESSING: Label Name for the baseling TestSuite : MITE:basenow1
[14-03-2022-12-45-16] : PROCESSING: Comment for the baseling TestSuite : basenow1
[14-03-2022-12-45-21] : PROCESSING: Checking the Sandbox Information
[14-03-2022-12-45-27] : PROCESSING: Checking for Active Sandbox.
[14-03-2022-12-45-32] : PROCESSING: Resynchronizing the Sandbox ...
[14-03-2022-12-45-37] : PROCESSING: Resynchronizing the Sandbox ...
[14-03-2022-12-45-41] : PROCESSING: Time Taken for Sandbox Create or Update: 20 Secs
[14-03-2022-12-45-41] : PROCESSING: ReadMe File is Created.
[14-03-2022-12-45-42] : ****
[14-03-2022-12-45-42] : PROCESSING :Sandbox Checkout processing Started....
[14-03-2022-12-45-42] : PROCESSING :Sandbox Checkout processing Ended....
[14-03-2022-12-45-42] : ****
[14-03-2022-12-45-42] : PROCESSING :Sandbox CheckIN processing Started....
[14-03-2022-12-45-42] : PROCESSING :Sandbox CheckIN processing Ended....
[14-03-2022-12-45-42] : ****
[14-03-2022-12-45-42] : PROCESSING :Sandbox Checkout processing Started....
[14-03-2022-12-45-42] : PROCESSING :Sandbox Checkout processing Ended....
[14-03-2022-12-45-42] : ****
[14-03-2022-12-45-42] : PROCESSING :Sandbox CheckIN processing Started....
[14-03-2022-12-45-42] : PROCESSING :Sandbox CheckIN processing Ended....
[14-03-2022-12-45-44] : PROCESSING: Resynchronizing the Sandbox ...
[14-03-2022-12-45-47] : PROCESSING: Resynchronizing the Sandbox ...
[14-03-2022-12-45-49] : PROCESSING: Resynchronizing the Sandbox ...
[14-03-2022-12-45-53] : FINAL: Baselining of TestSuite is Completed.
```

1.3 Test Suite As Of:

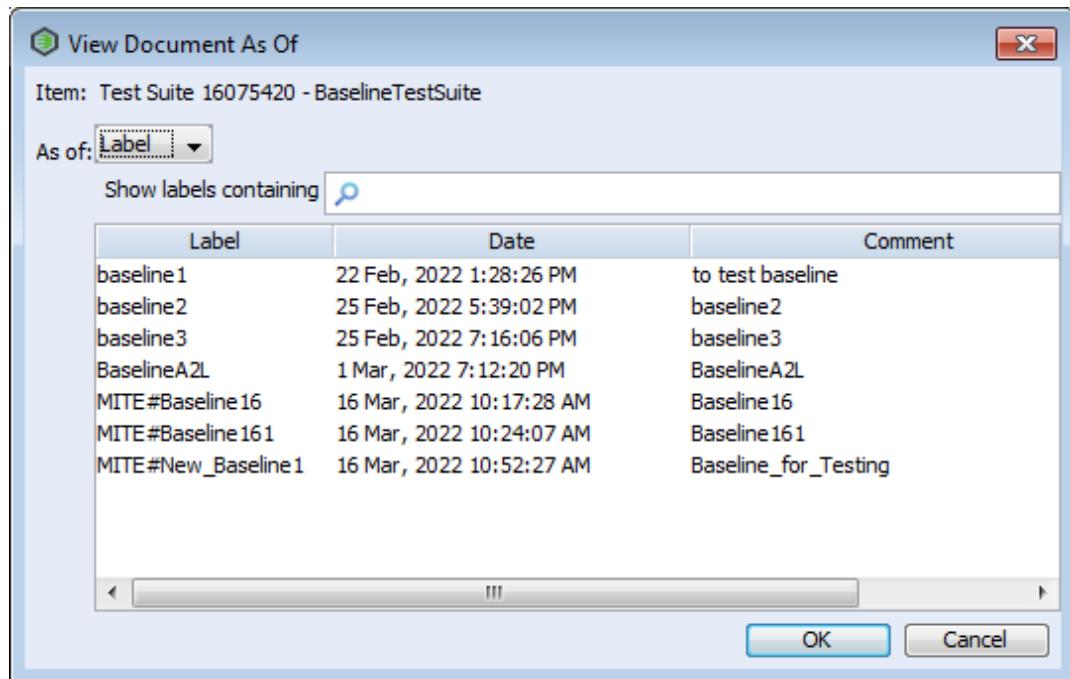
- The test suite which is baselined can be seen in the test suite as of frame. The frame consists of the following fields: Item, Search box, Name, Date, Comment, OK, Cancel, and Delete.



- **Item:** It resembles the Test Suite Name
 - **Search box:** Used to search the required baseline suite from the list
 - **Name:** The label name which is given in baseline frame will be shown
 - **Date:** The date and timestamp of suite when the baseline was made will be shown.
 - **Comment:** The comment which is given in baseline frame will be shown
 - **OK:** When a user wants to make modifications/accept changes after a certain operation, he needs to click the OK button
 - **Cancel:** When a user does not want to have any changes/accept changes after a particular operation, he needs to click the Cancel button.
 - **Delete:** When an user wants to delete a certain baseline test suite from the list of all baseline test suites that are present in the test suite as of frame, he must click the Delete button where the suite get deleted from MITE and as well as from PTC.

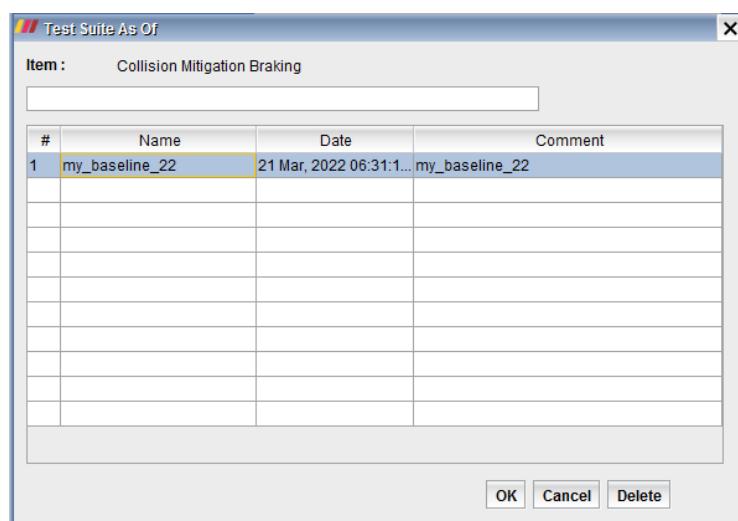
1.4 View Document As Of:

- To see the baselined suites in PTC, Go to respective project and select test suite and do right click and click on view my sandbox, Next search for related suite in document id, Document will be opened. Now click on Document option available in file menu. Navigate to the Historical and Select View As Of option and select label in dropdown to see baselined suites

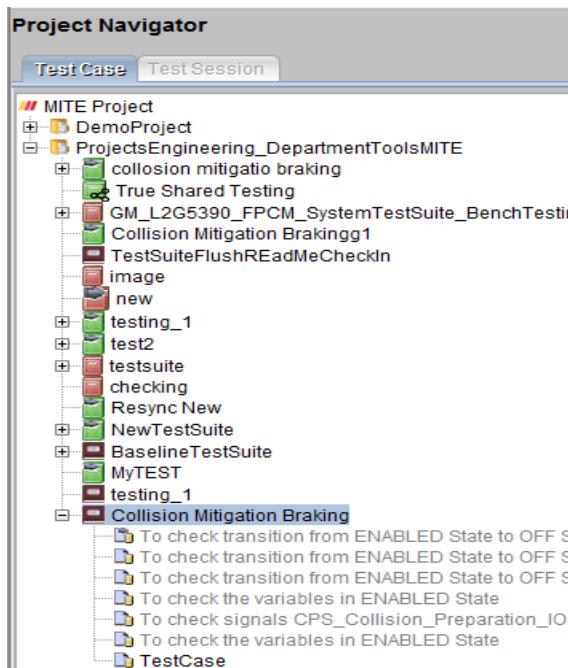


1.5 Loading of Baseline test suite:

- User can load the baseline test suite from the list of baseline suites available in test suite as of frame.

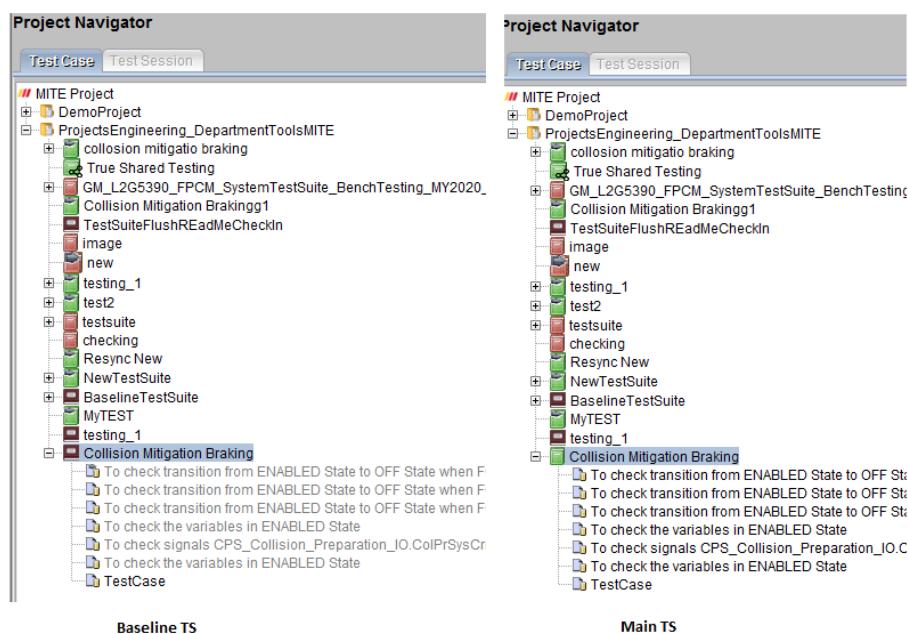


- The baseline test suite cannot be edited, they can only be read. The cases under the baseline suite are in grayed out.
 - In both the roles test case and script the baseline suite is not editable they can only be read. Incase, if user tries to do remap the labels in the label mapping frame then the labels wouldn't be saved.



1.6 Loading of Main Test Suite:

- If the user wants to load the main test suite from the existing baseline test suite, the existing baseline suite must be deleted.
- Go to the existing baseline test suite and right click on it and select delete from workspace option, the suite will get deleted from the project tree and reload the deleted suite by going into summary tab of MITE editor window (I.e, load the test suite)



Created with the Personal Edition of HelpNDoc: [Easy CHM and documentation editor](#)

7 Import Label database

This option will be provided only for Test Suite(s) as the Label Database contains all the parameters/labels information.

To “Import Label Database”: Right-Click on Label Database – present at left most

down corner of the MITE home window

Created with the Personal Edition of HelpNDoc: [Free help authoring environment](#)

7.1 Import Label database through Label Database window

A. Steps to Import Label database:

1. Right-Click on Label Database
2. Select the option –“ Import Label Database Files”

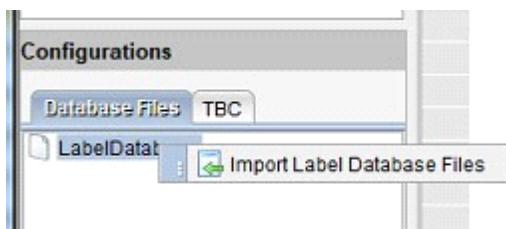


Figure 12: Label Database window

3). The above shown option will be available at any level of Test suite and Test cases, it will reflect for the entire TestSuite

B. Steps to Export Label database:

- 1.Right-Click on Label Database
- 2.Select the option –“ Export”

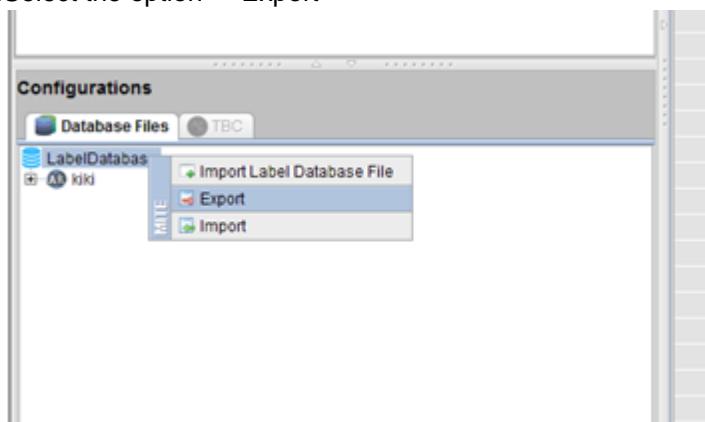


Figure 12.1 : Export Label data base window

C. Steps to Import Label database:

- 1.Right-Click on Label Database
- 2.Select the option –“ Import”

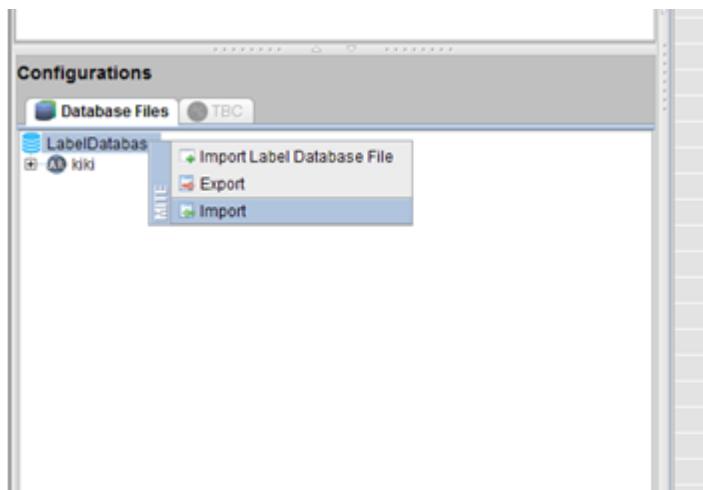


Figure 12.2 : Import Label data base Window

D. Steps to Export Individual Label database:

- 1.Right-Click on Label Database File
- 2.Select the option –“ Export”

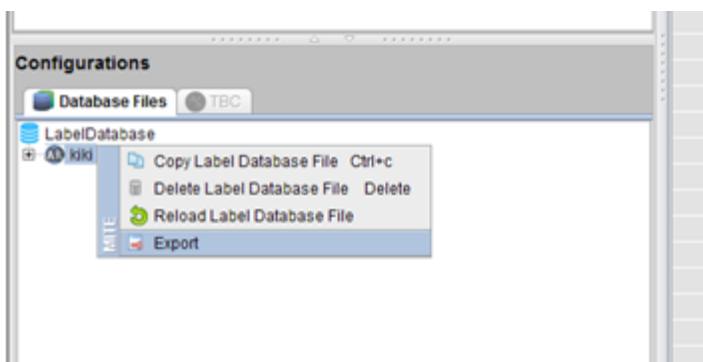


Figure 12.3 : Export of individual Label Data Base File

Created with the Personal Edition of HelpNDoc: [Easily create CHM Help documents](#)

7.2 Fill Import Label Database particulars

- i. For importing Label Database successfully the below fields are require to be filled and then click on “OK” button.

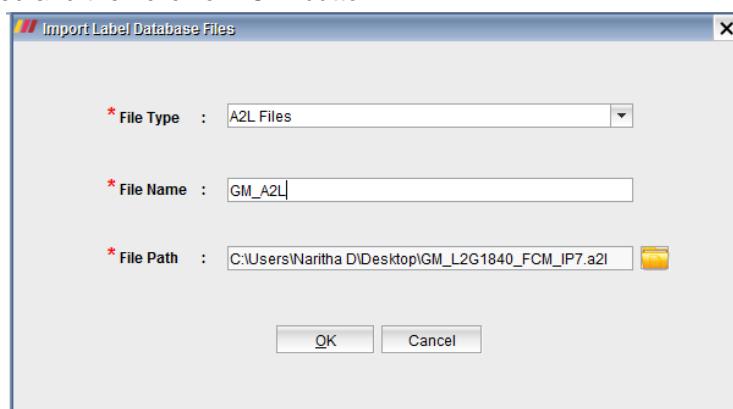
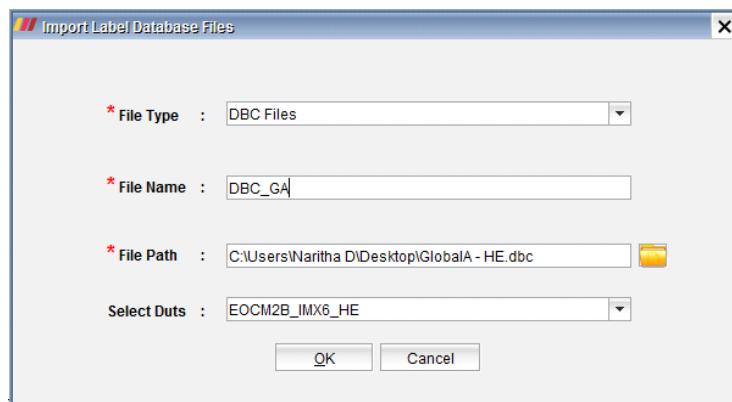
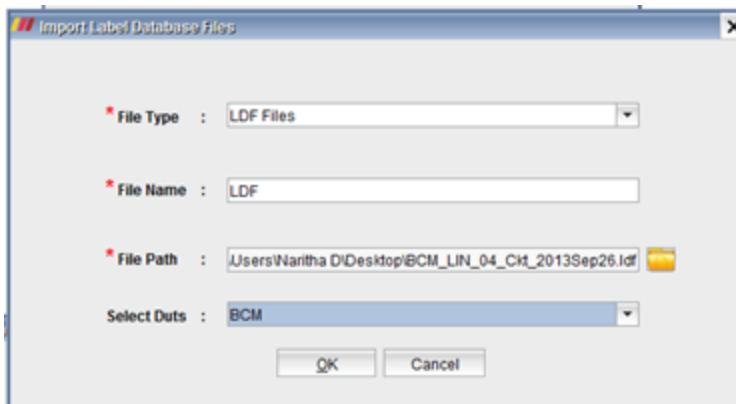
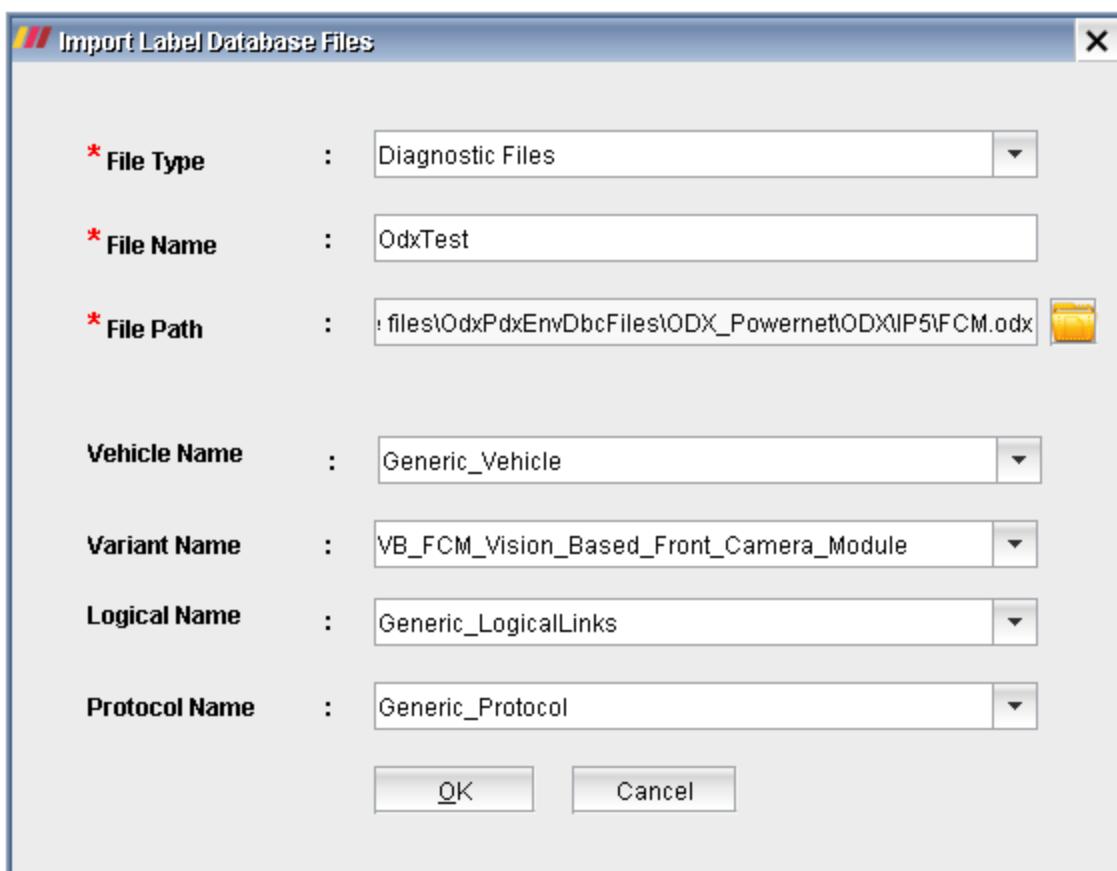
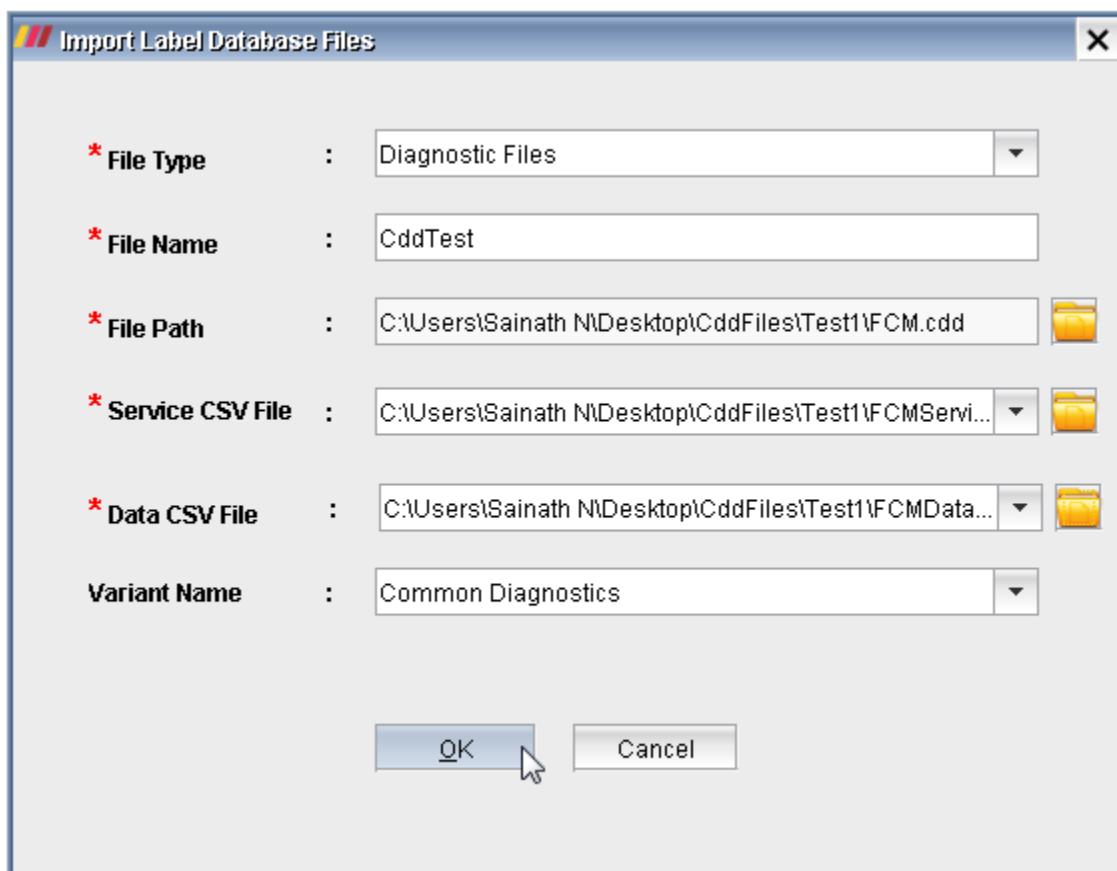
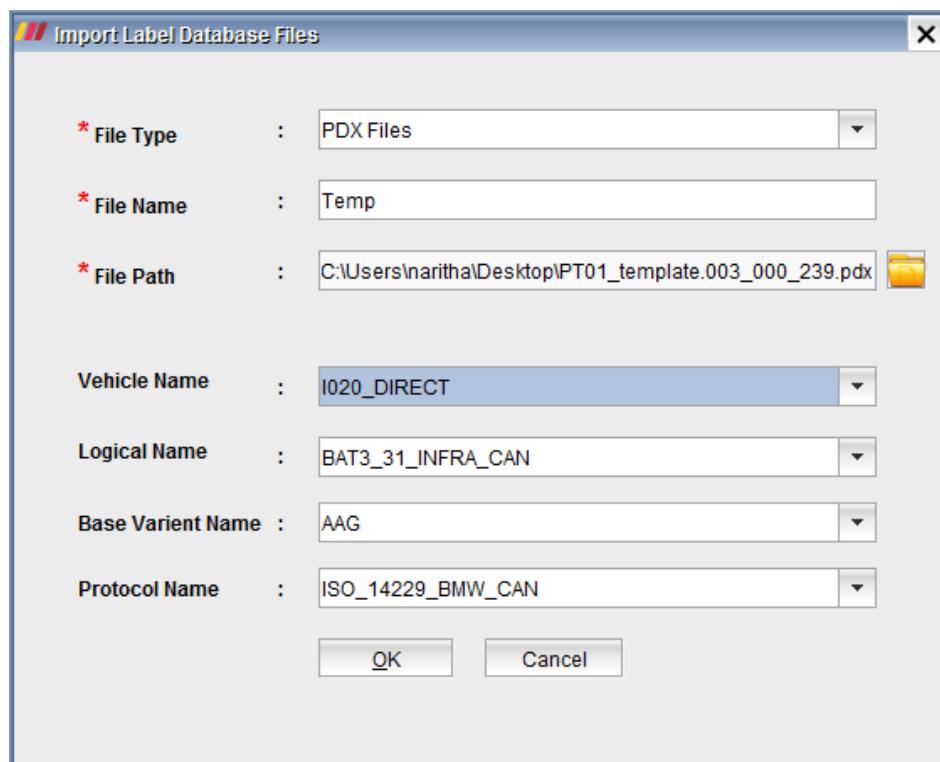


Figure 13: A2L Importing Label Database particulars

Figure 14: **DBC** Importing Label Database particularsFigure 15: **LDF** Importing Label Database particularsFigure 16: **ODX** Importing Label Database particulars

Figure 17: **CDD** Importing Label Database particularsFigure 18: **PDX** Importing Label Database particulars

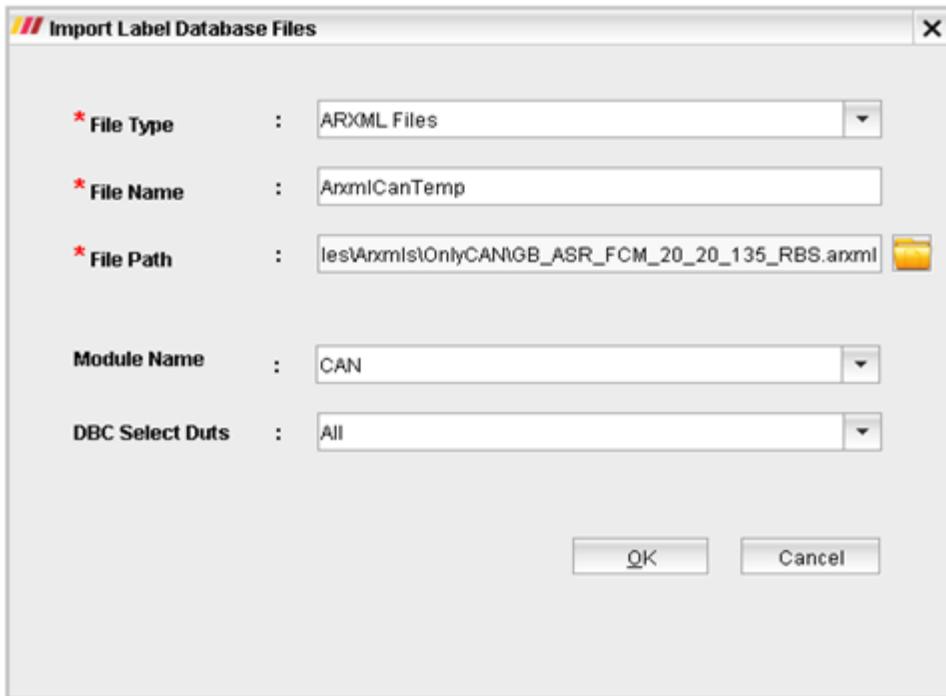


Figure 18.1:ARXML CAN Importing Label Database particulars

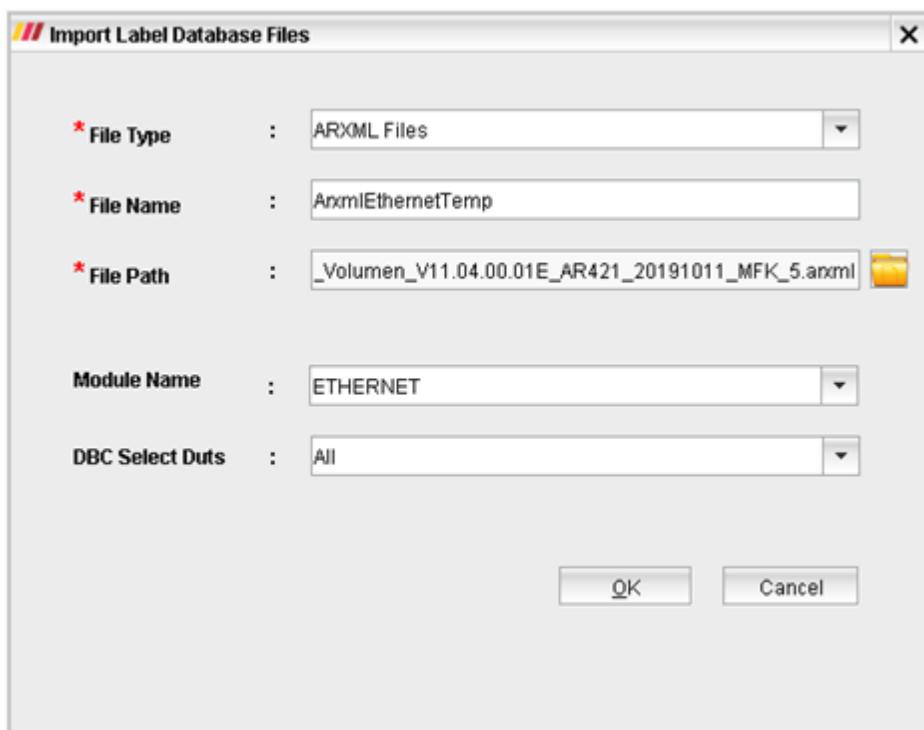


Figure 18.2: ARXML ETHERNET Importing Label Database particulars

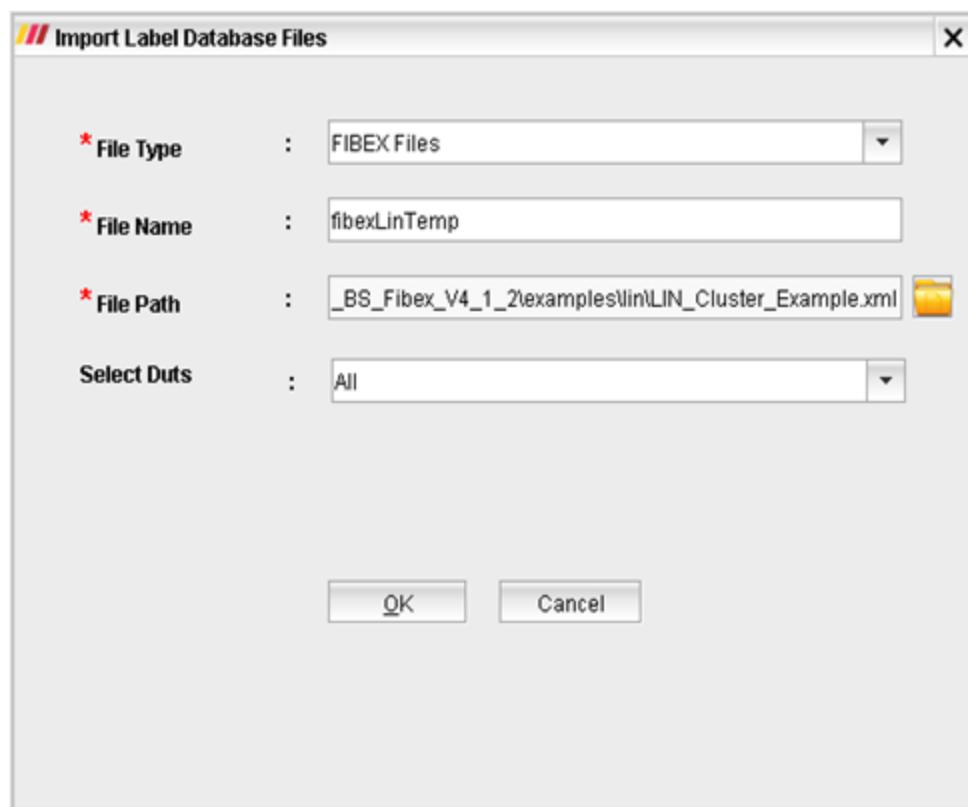


Figure 18.3: **FIBEX LIN** Importing Label Database particulars

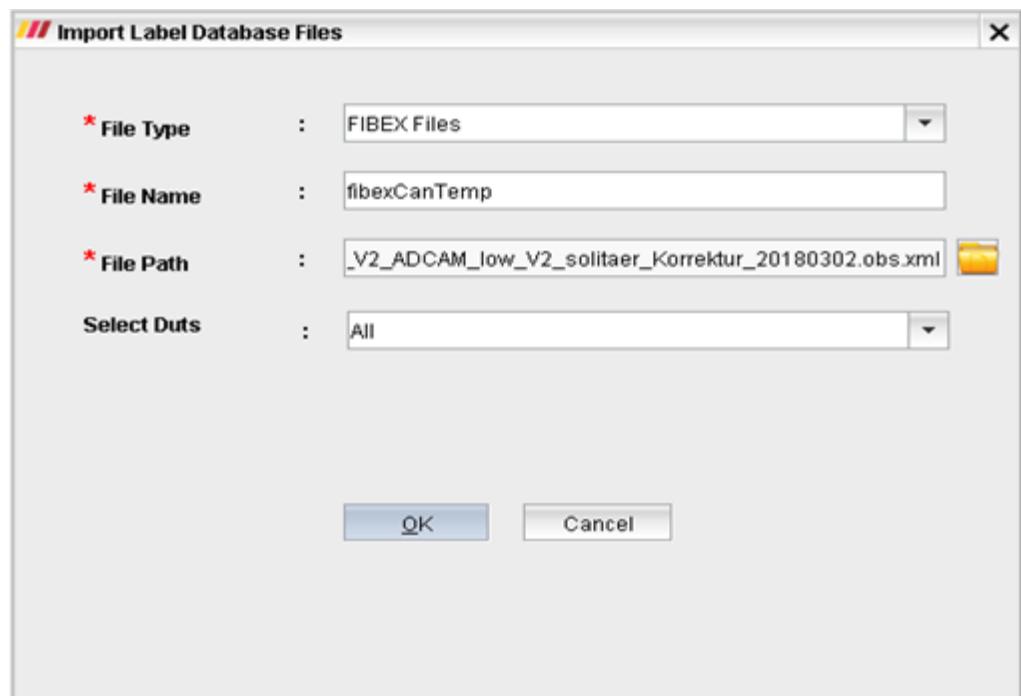


Figure 18.4: **FIBEX CAN** Importing Label Database particulars

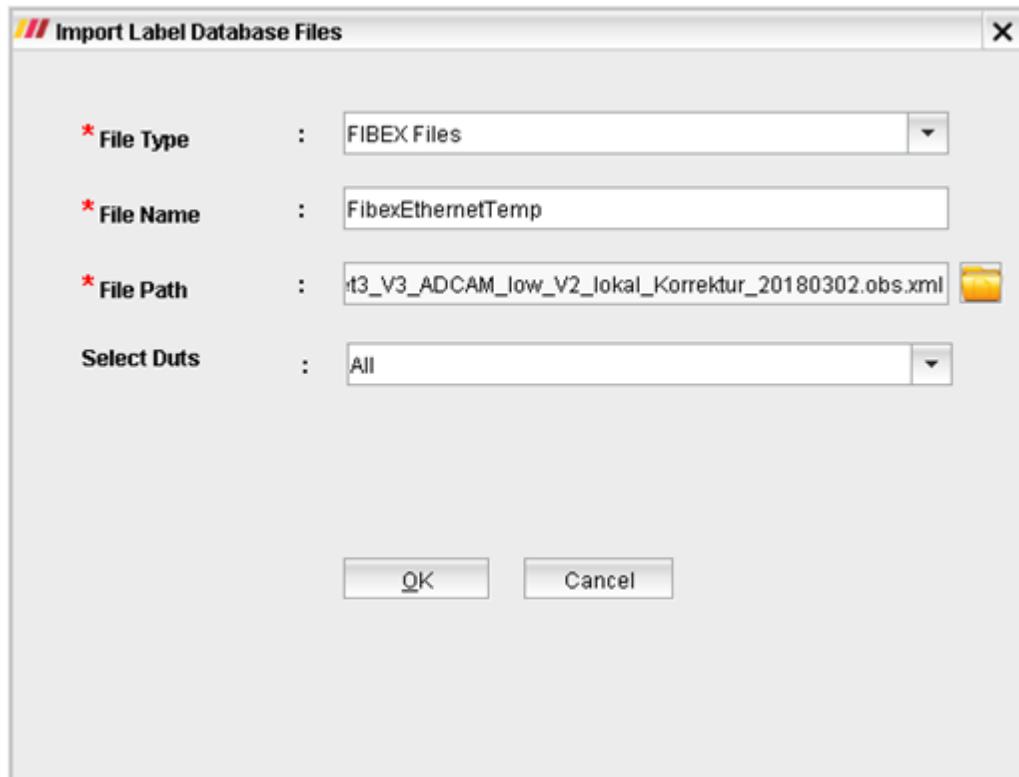


Figure 18.5: **FIBEX ETHERNET** Importing Label Database particulars

On adding Item	Fields	Operation performed
Import Label Database Files	File Type	Type of database dbc, ldf, pdx, odx, cdd, Arxml , Fibex or a2l files
	File Name	User Name for the selected file type
	File Path	Provide the path for the import file

Table 6: Label Database particulars

- ii. After selecting the import file, select the DUT from the “Select DUTs” option as per user requirement.
 - iii. The imported database file(s) and its DUT(s) list will be shown in “Label Database” window
1. From here on follow the same steps described above for “Import Label Database Files” in section-7.2
- B. Imported Label Database Tree:
- All the imported Database files will be saved in a Tree structure on successful import.

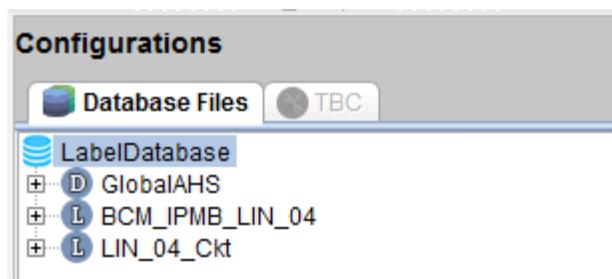


Figure 19: List of added label database

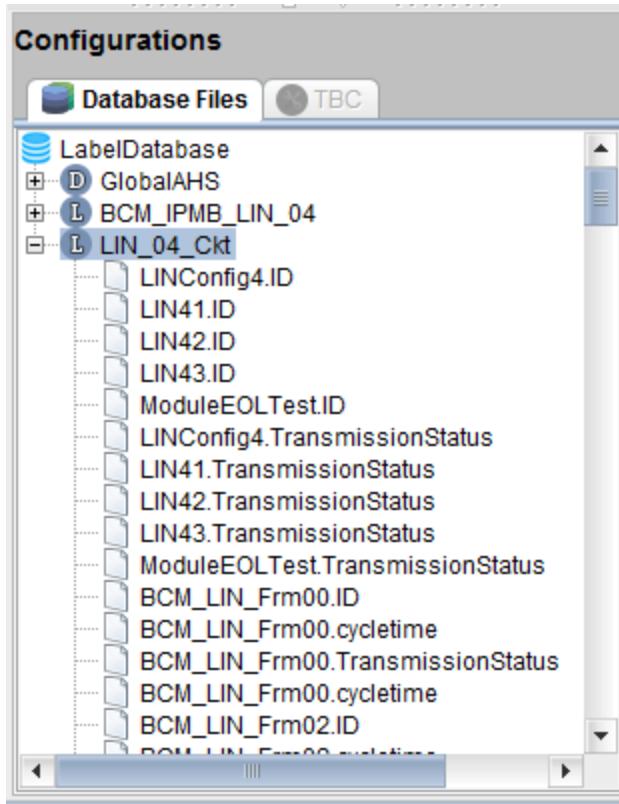


Figure 20: After clicking on ‘+’ symbol

Note:

From MITE v2.1.8.0, services are also available in ARXML files. Please find the below screenshots.

System Test - testcase1				
#	Action	Parameter/Service Type	ParameterDescription	Desired Value/Expected Value
#	Pre Conditions			
1	Set	Ethernet Signal	sit_24725::shortRangeLinkLayerSingleValue::shortRang...	
2		Ethernet Signal	sit_24603::posMapInfo::posMapInfo[1].accuracyLaneAc...	
#	Test Sequence			
1		Ethernet Signal	sit_24725::dynamicContentSingleValue::layerInformation...	
2		Ethernet Signal	sit_25088::pedestrians::pedestrians_azimuthAngle[11]	
3		Ethernet Signal	sit_25088::pedestrians::pedestrians_azimuthAngle[11]	
4		Ethernet Signal	sit_24725::shortRangeLinkLayerSingleValue::shortRang...	
5		Ethernet Signal	sit_24731::roadmodelRoad::poolAttribute[0].attributePoin...	
#	Post Conditions			

A dropdown menu is open over the 'Desired Value/Expected Value' column for the fourth row (Action: Set, ParameterType: Ethernet Signal). The menu contains the following options:

- 0 - false
- 1 - true
- HSFZitem
- BlockageAndWeatherFrontCameraMain
- DisplayDriverAssistanceLateral
- DriverAssistanceFunctionsLongitudinalFollow
- EgomotionCamera

1	Set	Ethernet Signal	sif_24725::shortRangeLinkLayerSingleValue::shortRangeLinkList[0].toLinkList[0].linkReferer...
2		Ethernet Signal	sif_24603::posMapInfo::posMapInfo[1].accuracy/isLaneAccurate
#	Test Sequence		
1		Ethernet Signal	sif_24725::dynamicContentSingleValue::layerInformation.metaDataList[0].generalSpeedLimit...
2		Ethernet Signal	sif_25088::pedestrians::pedestrians.azimuthAngle[11]
3		Ethernet Signal	sif_25088::pedestrians::pedestrians.azimuthAngle[11]
4		Ethernet Signal	sif_
5		Ethernet Signal	sif_24725::shortRangeLinkLayerSingleValue::shortRangeLinkList[0].toLinkList[0].linkReferer...
6			sif_24711::evTrafficSignsFrame::detectedTrafficSignRadialDistanceURC[0]
7			sif_24604::positionsQualityEvent::element.statsInUse valueType
8			sif_24711::evTrafficSignsFrame::mainSignOrTrafficLightReferenceObjectid[0]
9			sif_24604::positionsQualityEvent::noOfResourceElements
10			sif_24725::dynamicContentSingleValue::referencedAttributesList[0].attributeList[0].attributeP...
11			sif_25088::pedestrians::confirmationBits.confBitEgoTurningL3
#	Post Conditions		
1			
2			
3			

Created with the Personal Edition of HelpNDoc: [Free EPub producer](#)

7.3 Options available on imported Label Database files -Right click

The options available on Right click on “Database file” are as below:

Select on Item	Options available on Right click	Operation performed
ANY FILE (DBC,LDF,A2L,CD D, ODX, PDX)	View More Details	For ANY FILE details
	View Table	For selected file, data will be displayed in a table format with all the messages and signals present in the file.
	Copy Label Database File	Copy selected database file
	Delete Label Database File	Delete selected database file

Table 7: Right click options on “Database file”

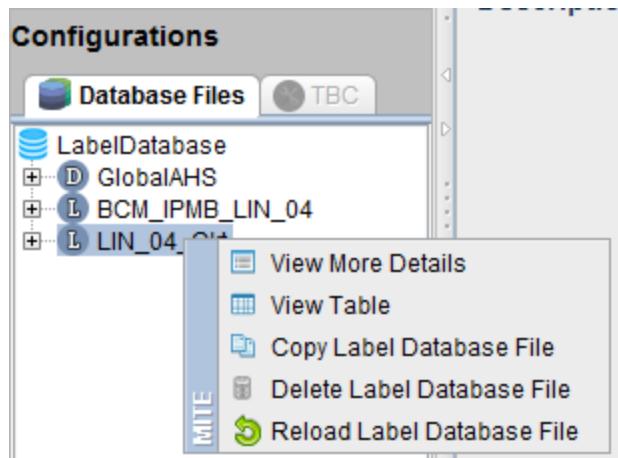


Figure 21: After clicking on ‘+’ symbol

This Position of Right Click Options are slightly different for ARXML

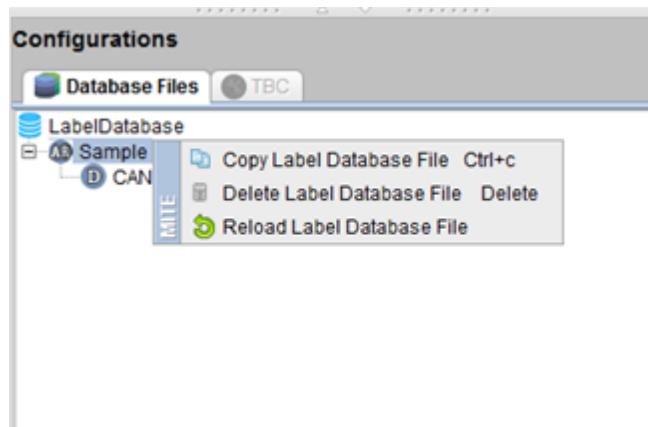


Figure 21.1 : After clicking on '+' symbol for Arxml File

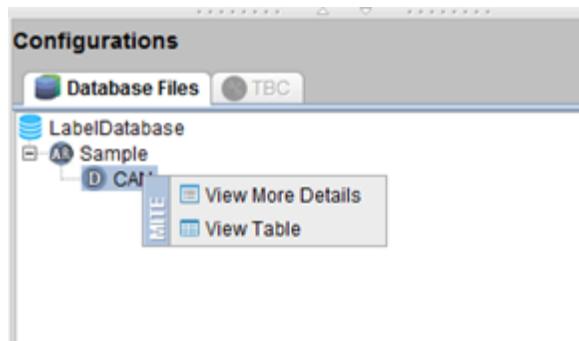


Figure 21.2: After clicking on '+' symbol on child of Arxml File

Created with the Personal Edition of HelpNDoc: [Write EPub books for the iPad](#)

7.4 On View More Details

Displays the following file information:



Figure 22: "View Table" options on "Database file"

Created with the Personal Edition of HelpNDoc: [Free Web Help generator](#)

7.5 On View Table

The following is the table format in which complete data will be shown:

GM_L201840_FCM-A2L

Labels : 36934

#	Label Name	Data Type	Conversion	Minimum Value	Maximum Value
1	ACCAxiTrqCmd	UBYTE	NO_COMP...	0	255
2	ACCAxiTrqCmd.ACCAxi...	UBYTE	NO_COMP...	0	255
3	ACCAxiTrqCmd.ACCAxi...	UBYTE	NO_COMP...	0	1
4	ACCAxiTrqCmd.ACCAxi...	UBYTE	NO_COMP...	0	3
5	ACCAxiTrqCmd.ACCAxi...	UBYTE	NO_COMP...	0	255
6	ACCAxiTrqCmd.ACCAxi...	UBYTE	NO_COMP...	0	255
7	ACCAxiTrqCmd.ACCAxi...	UBYTE	NO_COMP...	0	7
8	ACCAxiTrqCmd.ACCAxi...	UBYTE	NO_COMP...	0	3
9	ACCAxiTrqCmd.ACCAxi...	UBYTE	NO_COMP...	0	1
10	ACCAxiTrqCmd.ACCAxi...	UBYTE	NO_COMP...	0	127
11	ACCAxiTrqCmd._c	UBYTE	NO_COMP...	0	255
12	ACCAxiTrqCmd._c_0_	UBYTE	NO_COMP...	0	255
13	ACCAxiTrqCmd._c_1_	UBYTE	NO_COMP...	0	255
14	ACCAxiTrqCmd._c_2_	UBYTE	NO_COMP...	0	255
15	ACCAxiTrqCmd._c_3_	UBYTE	NO_COMP...	0	255
16	ACCAxiTrqCmdStat	UBYTE	NO_COMP...	0	255
17	ACCAxiTrqCmdStat.AC...	UBYTE	NO_COMP...	0	255
18	ACCAxiTrqCmdStat.AC...	UBYTE	NO_COMP...	0	3
19	ACCAxiTrqCmdStat.AC...	UBYTE	NO_COMP...	0	7
20	ACCAxiTrqCmdStat.AC...	UBYTE	NO_COMP...	0	7
21	ACCAxiTrqCmdStat._c	UBYTE	NO_COMP...	0	255
22	ACCAxiTrqCmdStat._c...	UBYTE	NO_COMP...	0	255
23	ACCBrkSysCmmndExt	UBYTE	NO_COMP...	0	255
24	ACCBrkSysCmmndExt...	UBYTE	NO_COMP...	0	255
25	ACCBrkSysCmmndExt...	UBYTE	NO_COMP...	0	255

Figure 23: "View Table" options on "Database file"

Created with the Personal Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

7.6 On Copy Label Database File

Copies the currently selected database file and allows pasting it into another TestSuite irrespective of the Project.

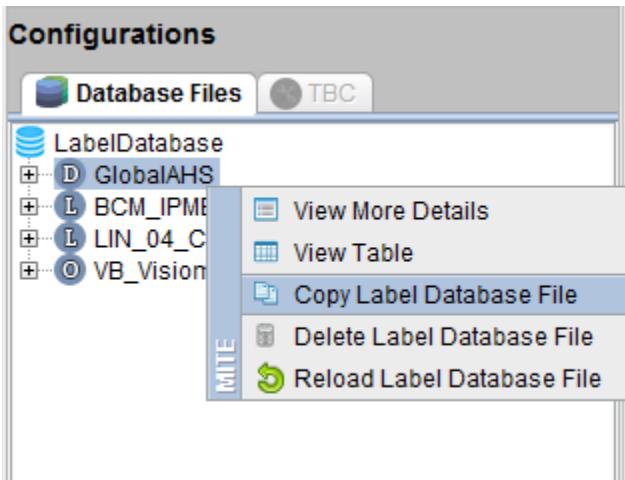


Figure 24: "Copy Label Database File" option

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

7.7 On Paste Label Database File

Paste Label Database File appears only when copy Label Database operation occurs and on "Label Database Right Click" options

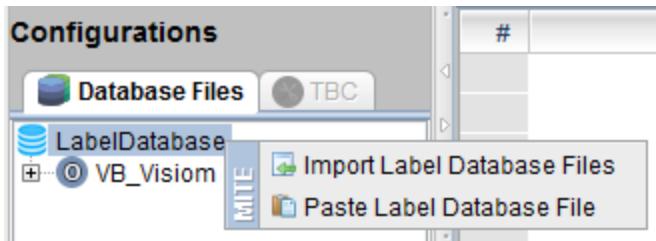


Figure 25: "Paste Label Database File" option

Created with the Personal Edition of HelpNDoc: [Single source CHM, PDF, DOC and HTML Help creation](#)

7.8 On Delete Label Database File

Deletes the selected file from the tree for the currently selected test suite

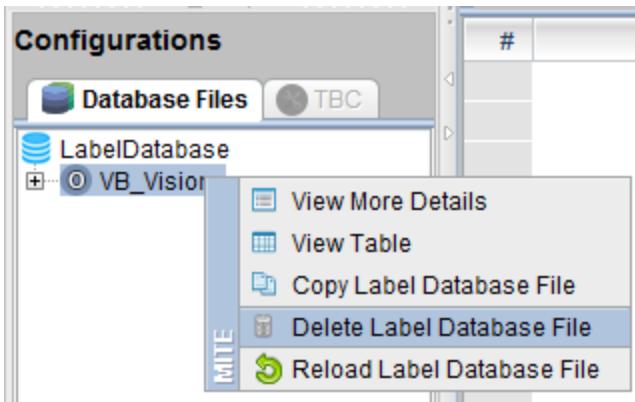
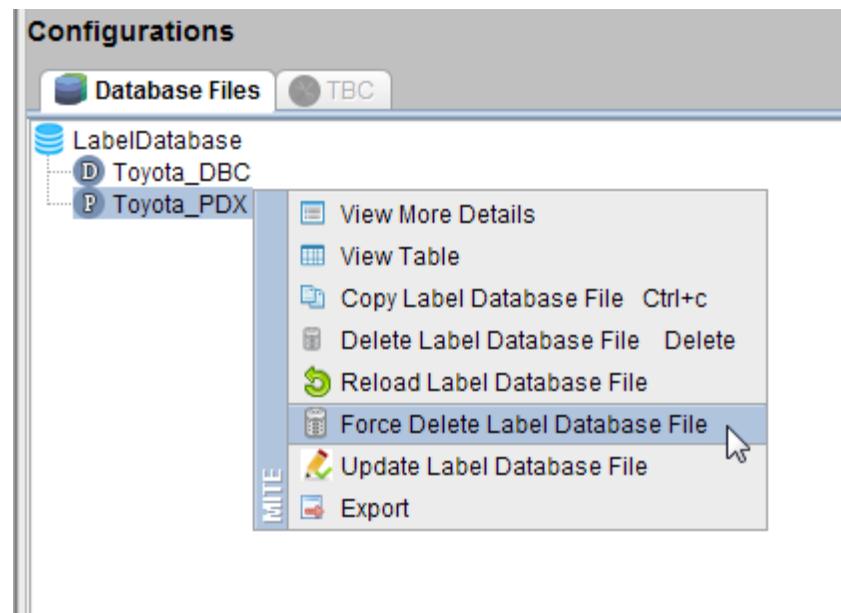


Figure 26: "Delete Label Database File" option

Force Delete of the Label Data Base Files:

This option in MITE is made available from MITE - v 2.1.6.0. Generally, up to then if the labels from a particular label data base file are used in test case authoring then user is not allowed to delete that particular label data base file. But, this option of "Force Delete Label Database File", allows user to delete those type of used files also.

But, if those used labels are not present in any of the other label database files, then the labels are shown in pink color in the Labels tab in the Test suite level. These labels can be either synchronized by adding appropriate Label Database file or they can be updated using the "Update Labels" option.



Created with the Personal Edition of HelpNDoc: [News and information about help authoring tools and software](#)

7.9 On Reload Label Database File

It provides the opportunity to reload the newly added labels of the same file into the tree for the currently selected label database file just by providing the file path as shown below :

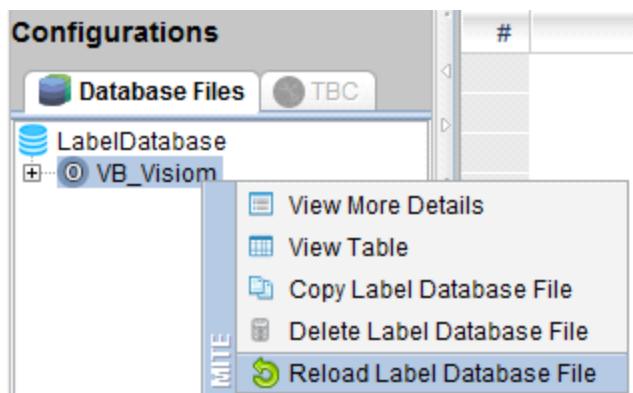


Figure 27: "Reload Label Database File" option

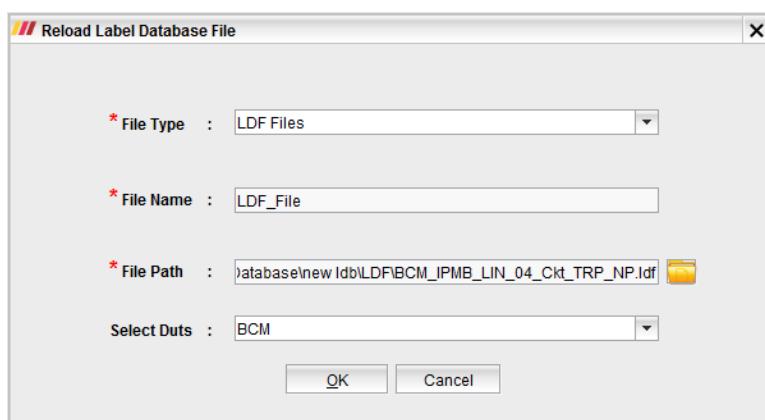


Figure 28: File path to Reload Label Database File

Created with the Personal Edition of HelpNDoc: [Free PDF documentation generator](#)

7.10 On Update Label Database File

It allows the user to update Diagnostic Vehicle and Logical Link of an already imported diagnostic base variant as shown below:

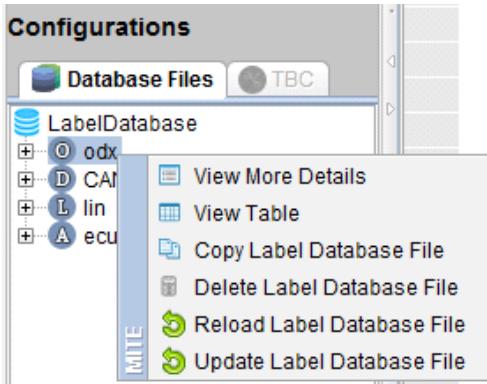


Figure 29: "Update Label Database File" option

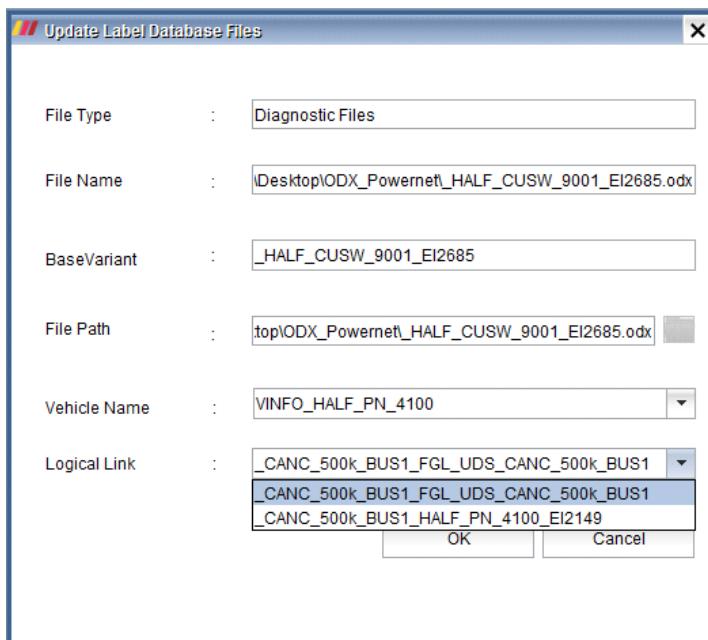


Figure 30: Select Vehicle Name and Logical link from drop down list

NOTE – Update operation is only available for Diagnostic Files such as ODX,PDX

Created with the Personal Edition of HelpNDoc: [Full-featured Documentation generator](#)

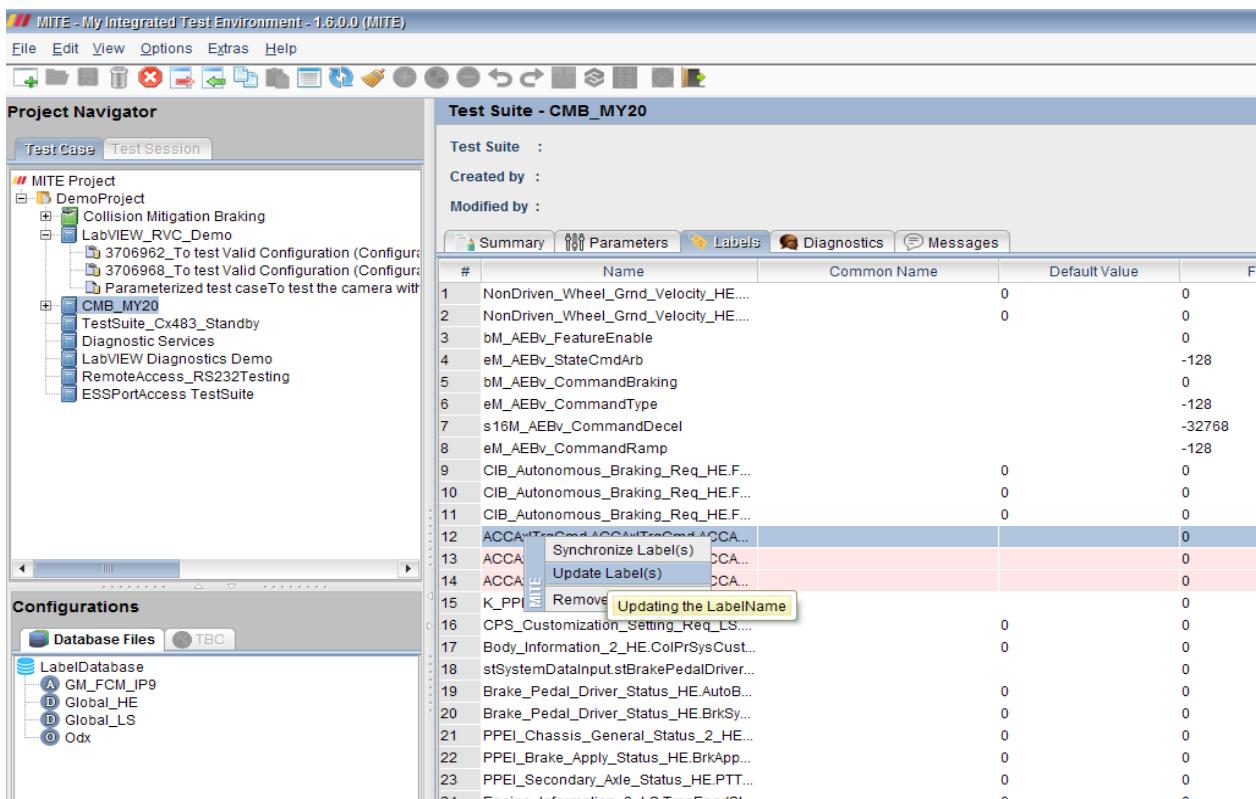
7.11 Update and Synchronize options for Diagnostics and Labels

Update and Synchronize options for Diagnostics and Labels

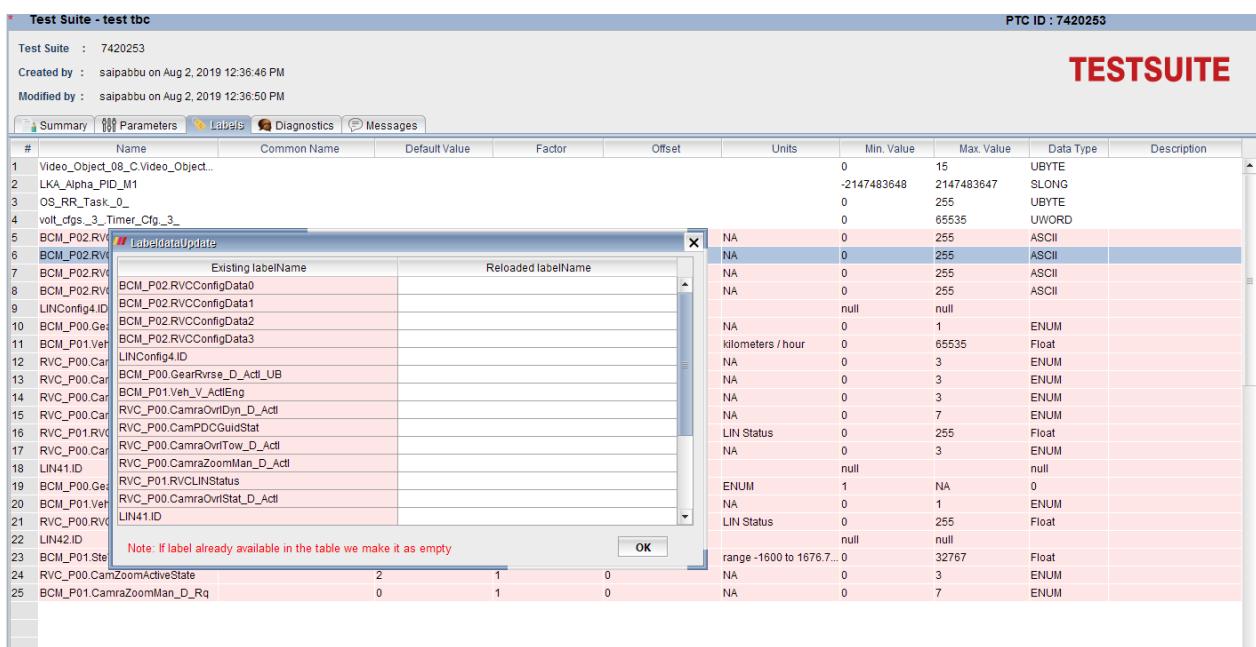
(1)Update Labels:-

- Whenever User wants to change/modify any labels then he/she can use update labels as option in Test Suite level. If User edits any labels at Test Case editor those labels can be automatically used in Test Suite levels.. For example in

- recorder,labels,diagnostics etc.
- It can be performed in various scenarios for example while copy and paste of test cases or while reloading any label database file. The following scenarios will help a User on how to update the labels while performing the action:-

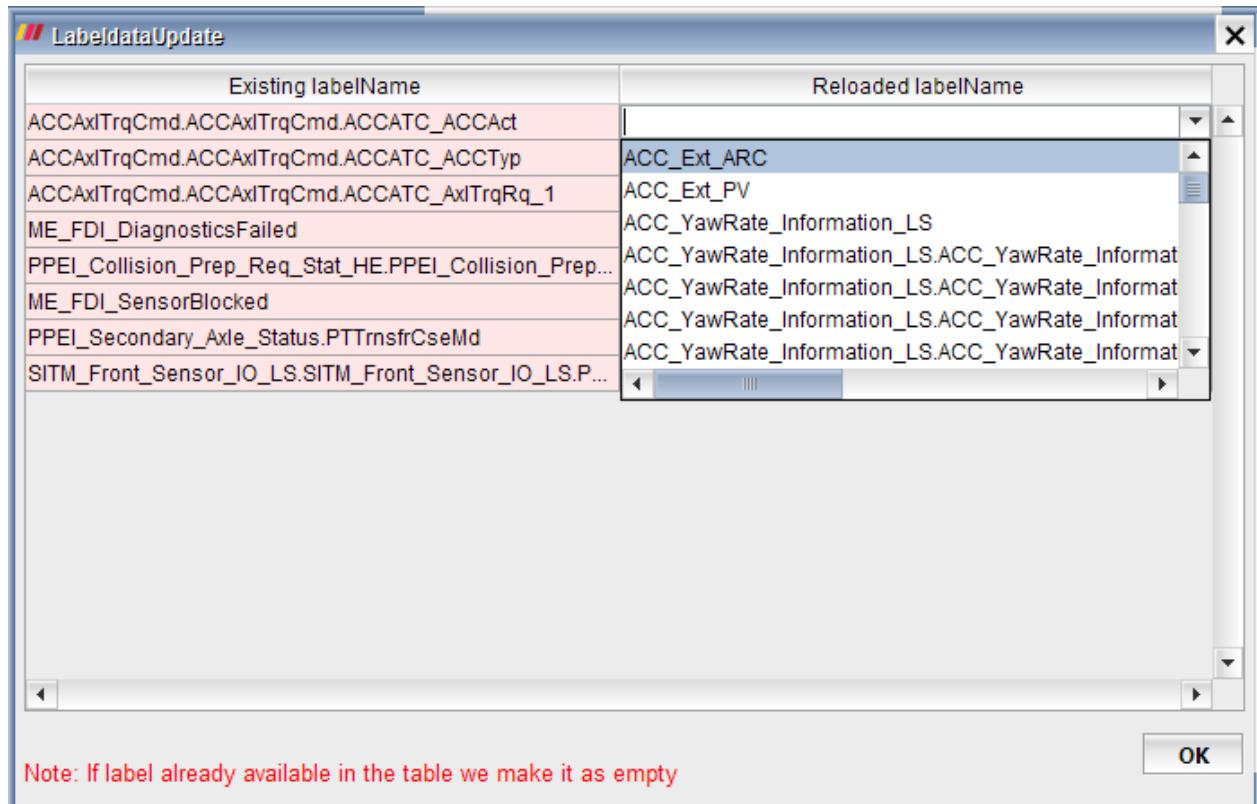


User can click on that Update Label option and then he/she will get below dialog box as shown:



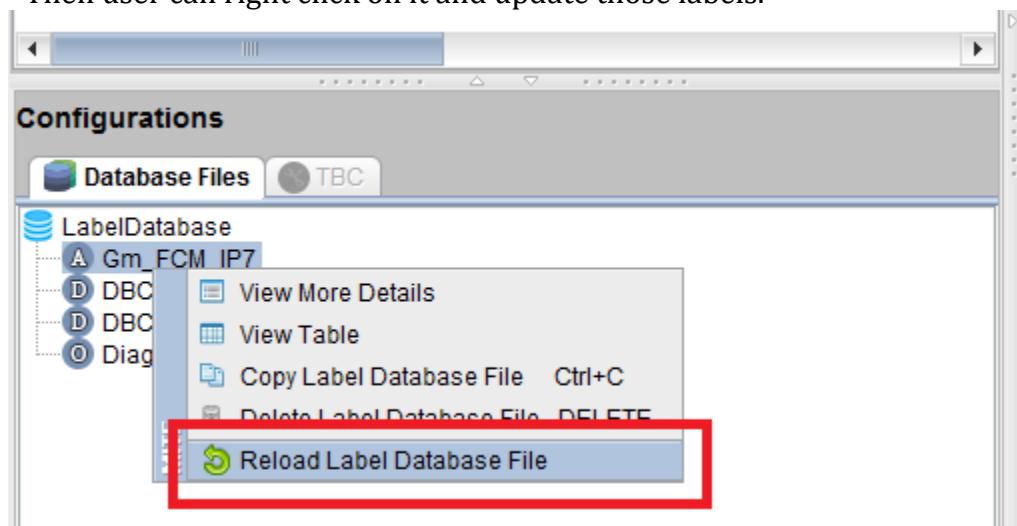
- Here Existing labels means which are already present in Test Suite panel.

- Reload label names means which User wants to update labels in the respective Test Suite.
- Select the labels in that combo box, if it is already used those labels are not allowed to use for second time otherwise it will allow and click ok option.

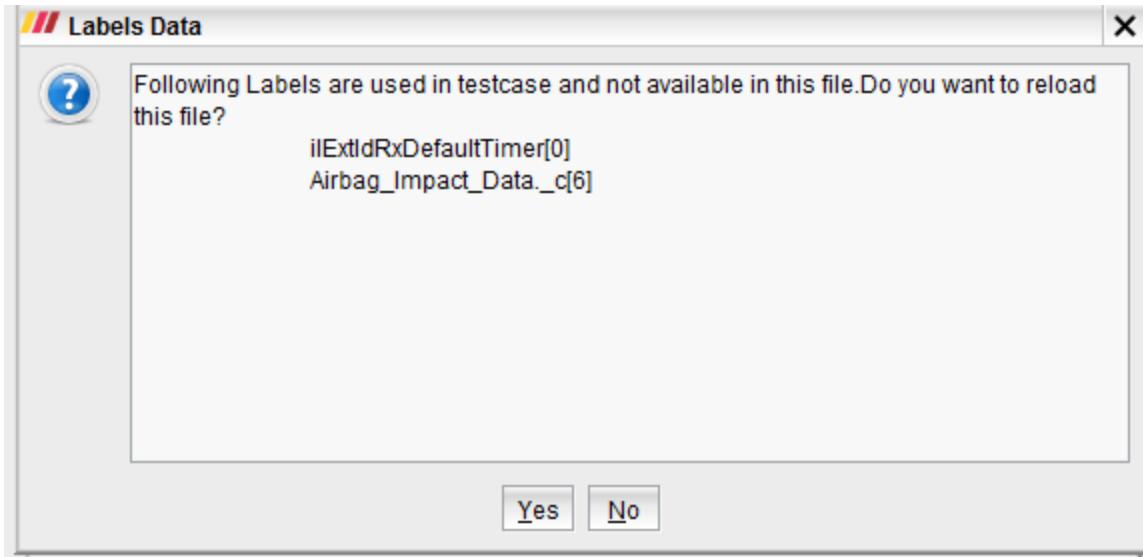


Reload Label database:

- When User perform Reload label database file option, at that time label database file will update, if it is already used in the Test Cases.
- Those labels will be changed to red color in the Test Suite panel, at that time if User wants to update the updated label data base file then User can use that update labels option and it is available in the Test Suite right side label panel.
- Then user can right click on it and update those labels.



The following pop up will occur if the labels are not available and User wants to reload:-

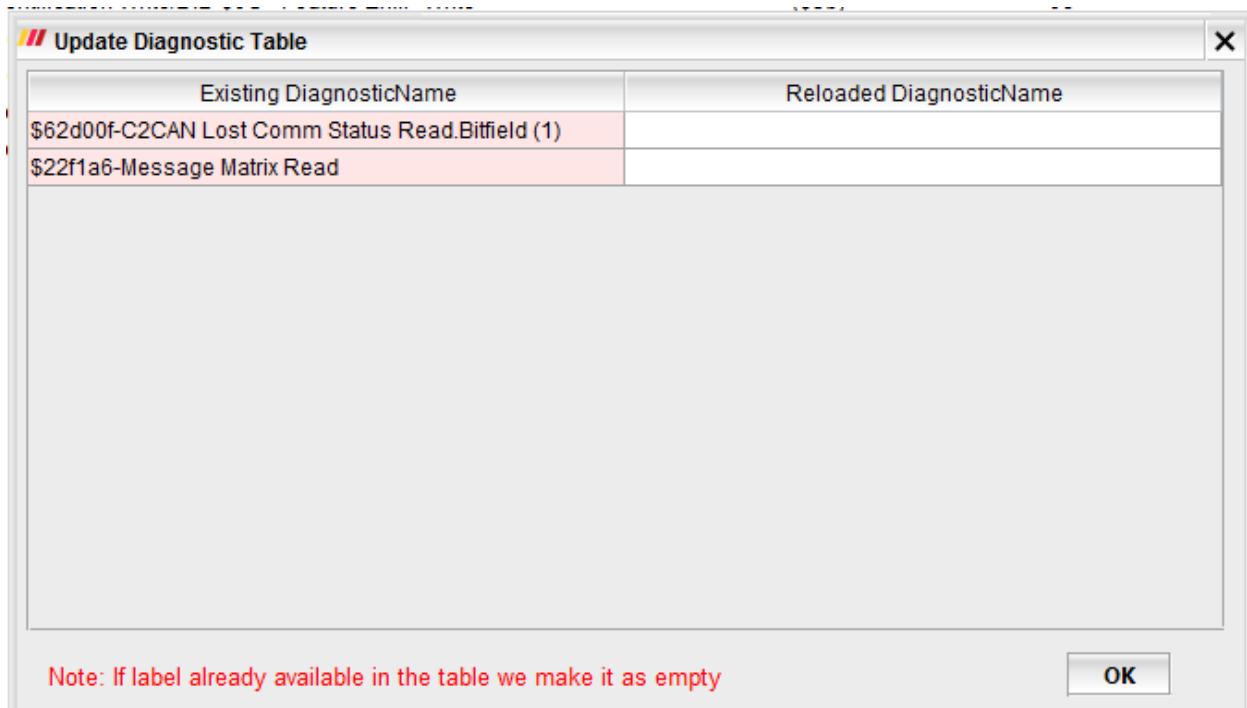


(2)Update Diagnostics:

If User wants to change/modify any diagnostics then he/she can use update Diagnostics as option in Test Suite level.

#	Name	Service Name	Service Id	DID Id	SubFunction Id	Signal Name	Bit Length	By	
1	Synchronize Diagnostic(s)	Status Read Bitfield (1)	Read	(\$62)	(\$d000)	NA	Bitfield (1)	8	3
2	Update Diagnostic(s)	DID 96E - TACI SP-TA.. Write	Write	(\$22)	(\$f1a5)	NA	NA	16	1
3	Remove Diagnostic	DID SOC - Feature En.. Write	Write	(\$3b)	(\$0c)	NA	SLIF Enable	1	2
4	\$3b0-Ecu identification Write	DID S01 - DTC Enabl.. Write	Write	(\$3b)	(\$01)	NA	DTC B356A 02	1	0
5	\$4a0-Ecu identification Read	DID S01 - DTC Enabl.. Read	Read	(\$5a)	(\$01)	NA	DTC B356A 02	1	0
6	\$3b0-Ecu identification Write/DID S0C - Feature En.. Write		Write	(\$3b)	(\$0c)	NA	FDI Enable	1	3
7	\$4a0-Ecu identification Read/DID S0C - Feature En.. Read		Read	(\$5a)	(\$0c)	NA	FDI Enable	1	3
8									

- Here Existing labels means which are already present in Test Suite panel.
- Reload label name means which User wants to update labels in the respective Test Suite.
- Select the labels in that combo box, if it is already used those labels are not allowed to use for second time otherwise it will allow and click ok option.



3) Synchronize option for Labels and Diagnostics:-

Whenever User wants to Sync any Labels/Diagnostics which has occurred as red color(indicates there needs to have some modification based on requirements) then he/she can use synchronize option on right clicking that particular case based on User Requirements.

#	Name	Common Name	Default Value	Factor	Offset	Units	Min. Value
1	Synchronize Label(s)	ton.VehS...	0	0	Enum	1	1
2	Update Label(s)	: ColP Sys...	0	0	Enum	7	1
3	Remove Label	Status_HE_A...	0	0		1	1
4	PPEL_Pedal_Driver_Status_HE_B...		0	0		1	1
5	PPEL_Chassis_General_Status_2...		0	0		1	1
6	PPEL_Brake_Apply_Status_HE_Brk...		0	0		1	1
7	PPEL_Trans_General_Status_2_H...		15.0	0	Enum	15	1
8	bM_AEBv_CommandBraking		0	0	Enum	3	1
9	eM_AEBv_CommandType		-128	0	SBYTE	127	
10	bM_AEBv_CommandTorque		0	0	UBYTE	255	
11	eM_AEBv_CommandRamp		-128	0	SBYTE	127	
12	s16M_AEBv_CommandDecel		-32768	0	SWORD	32767	
13	K_AEBv_UseCICustomMenu		0	0	__UBYTE_S	255	
14	bM_AEBv_CustomizationValue		0	0	UBYTE	255	
15	CPS_Collision_Preparation_IO_H...		0	0	Enum	7	1
16	ACCAxTrqCmdACCAxitTrqCmdA...		0	0	UBYTE	1	0

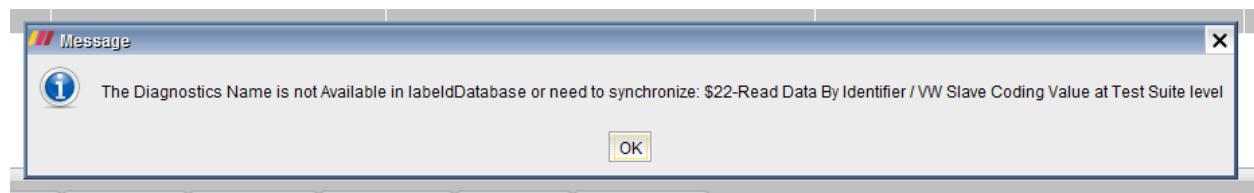
#	Name	Common Name	Default Value	Factor	Offset	Units	Min. Value	Max. Value	Data Type	Description
1	NonDriven_Wheel_Grnd_Velocity...		0	0.03125	0	km / h	0	511.96875	Float	Wheel Ground Velocity
2	NonDriven_Wheel_Grnd_Velocity...		0	0.03125	0	km / h	0	511.96875	Float	Wheel Ground Velocity
3	bM_AEBv_FeatureEnable						0	255	UBYTE	
4	eM_AEBv_StateCmdArb						-128	127	SBYTE	
5	bM_AEBv_CommandBraking						0	255	UBYTE	
6	eM_AEBv_CommandType						-128	127	SBYTE	
7	s16M_AEBv_CommandDecel						-32768	32767	SWORD	
8	eM_AEBv_CommandRamp						-128	127	SBYTE	
9	CIB_Autonomous_Braking_Req...		0	1	0		0	7	Enum	Forward Collision Alert C...
10	CIB_Autonomous_Braking_Req...		0	1	0		0	1	UBYTE	Forward Collision Alert C...
11	CIB_Autonomous_Braking_Req...		0	1	0		0	15	Enum	Forward Collision Alert C...
12	bM_FCM_Available						0	255	UBYTE	
13	eM_AEBr_RequestRamp						-128	127	SBYTE	
14	s16M_AEBr_RequestDecel						-32768	32767	SWORD	
15	CIB_Autonomous_Braking_Req...		0	0.01	0	m/s^2	-20.48	20.47	Float	Forward Collision Alert C...
16	ACCAxTrqCmd ACCAxTrqCmd A...						0	1	UBYTE	
17	ACCAxTrqCmd ACCAxTrqCmd A...						0	3	UBYTE	
18	ACCAxTrqCmd ACCAxTrqCmd A...						0	255	UBYTE	
19	Control_Power_Liftgate_LS_PwrL...		0	1	0		0	255	Float	Power Liftgate Inclination...
20	Driver_Door_Status DDAjgSwAtM		0	1	0		0	1	Enum	Driver Door Ajar Switch A...
21	CIB_Autonomous_Braking_Req...		0	1	0		0	1048575	Float	Forward Collision Alert C...
22	PPEI_Engine_General_Status_1...		0	1	0		0	1	Enum	Engine Run Active
23	Body_Information_2_HE_CoPrSys...		0	1	0		0	7	Enum	Collision Preparation Sys...

After Clicking on Synchronize Label Option User will get the following Result as shown below:-

#	Name	Common Name	Default Value	Factor	Offset	Units	Min. Value	Max. Value	Data Type	Description
1	NonDriven_Wheel_Grnd_Velocity...		0	0.03125	0	km / h	0	511.96875	Float	Wheel Ground Velocity Ri...
2	NonDriven_Wheel_Grnd_Velocity...		0	0.03125	0	km / h	0	511.96875	Float	Wheel Ground Velocity L...
3	bM_AEBv_FeatureEnable						0	255	UBYTE	
4	eM_AEBv_StateCmdArb						-128	127	SBYTE	
5	bM_AEBv_CommandBraking						0	255	UBYTE	
6	eM_AEBv_CommandType						-128	127	SBYTE	
7	s16M_AEBv_CommandDecel						-32768	32767	SWORD	
8	eM_AEBv_CommandRamp						-128	127	SBYTE	
9	CIB_Autonomous_Braking_Req...		0	1	0		0	7	Enum	Forward Collision Alert C...
10	CIB_Autonomous_Braking_Req...		0	1	0		0	1	UBYTE	Forward Collision Alert C...
11	CIB_Autonomous_Braking_Req...		0	1	0		0	15	Enum	Forward Collision Alert C...
12	bM_FCM_Available						0	255	UBYTE	
13	eM_AEBr_RequestRamp						-128	127	SBYTE	
14	s16M_AEBr_RequestDecel						-32768	32767	SWORD	
15	CIB_Autonomous_Braking_Req...		0	0.01	0	m/s^2	-20.48	20.47	Float	Forward Collision Alert C...
16	ACCAxTrqCmd ACCAxTrqCmd A...						0	1	UBYTE	
17	ACCAxTrqCmd ACCAxTrqCmd A...						0	3	UBYTE	
18	ACCAxTrqCmd ACCAxTrqCmd A...						0	255	UBYTE	
19	Control_Power_Liftgate_LS_PwrL...		0	1	0		0	255	Float	Power Liftgate Inclination...
20	Driver_Door_Status DDAjgSwAtM		0	1	0		0	1	Enum	Driver Door Ajar Switch A...
21	CIB_Autonomous_Braking_Req...		0	1	0		0	1048575	Float	Forward Collision Alert C...
22	PPEI_Engine_General_Status_1...		0	1	0		0	1	Enum	Engine Run Active
23	Body_Information_2_HE_CoPrSys...		0	1	0		0	7	Enum	Collision Preparation Sys...

For Diagnostics:-

User will get pop up in test case editor shown below:



File Edit View Options Extras Help

Project Navigator

Test Case Test Session

MITE Project

Collision Mitigation Braking

To check transition from ENABLED State to OFF State when FCM is not in Enabled State
To check transition from ENABLED State to OFF State when FCM is not in Enabled State
To check the variables in ENABLED State
To check signals CPS_Collision_Preparation, IO_CpPgsCmrtSng and CPS_Collision_Enab and CPS_Collision_Disabl with Vss sleep is Enabled
To test the camera with valid Configuration with Vss sleep is Enabled
To test the camera with invalid Configuration with Remote start status equal to 1
To check transition from DISABLED State to ENABLED State when FDI feed is available
To check transition from ENABLED State to OFF State when FCM is not in Enabled State
To check transition from OFF State to Enabled State when FCM is not in Enabled State
To check the variables in OFF State
To check signals CPS_Collision_Preparation, IO_CpPgsCmrtSng and CPS_Collision_Enab and CPS_Collision_Disabl with Vss sleep is Enabled
To test the camera with valid Configuration with Vss sleep is Enabled
To test the camera with invalid Configuration with Remote start status equal to 1
To check transition from DISABLED State to Enabled State when FDI feed is available

Test Suite - Collision Mitigation Braking

Test Suite : 7502492

Created by : naridan on Aug 23, 2019 3:26:05 PM

Modified by : bharat on Aug 26, 2019 5:06:05 PM

Labels

#	Name	Service Name	Service Id	DID Id	SubFunction Id	Signal Name	Bit Len
1	Synchronize Diagnostic(s)	Read Bitfield(1)	(\\$62)	(\\$d00)	NA	Bitfield(1)	8
2	Update Diagnostic(s)	Read	(\\$22)	(\\$f1a5)	NA	NA	16
3	\$300-Ecu Identification Write	DID : TAC/SP-TA, Write	(\\$3b)	6e	NA	cameralinfo_ta2_focal_16	
4	\$3b01-Ecu Identification Write	DID : 01-DTC Enabl., Write	(\\$3b)	0c	NA	SLIF Enable	1
5	\$3b01-Ecu Identification Read	DID : 01-DTC Enabl., Read	(\\$5a)	(\\$01)	NA	DTC B356A.02	1
6	\$3b01-Ecu Identification Write	DID : 01-DTC Enabl., Write	(\\$3b)	(\\$01)	NA	DTC B356A.02	1
7	\$3b01-Ecu Identification Read	DID : 01-DTC Enabl., Read	(\\$5a)	(\\$0c)	NA	FDI Enable	1
8	\$5a0c-Ecu Identification Read	DID : 0c-Feature En., Read	(\\$5a)	(\\$0c)	NA	FDI Enable	1

After Clicking on Synchronize Diagnostics Option User will get the following Result as shown below:-

#	Name	Service Name	Service Id	DID Id	SubFunction Id	Signal Name	Bit Length	Byte Position
1	\$3b6-Ecu identification Write/DID \$6E - TAC/ SP-TA... Write		(\$3b)	6e	NA	cameraInfo_tac2_focal...	16	24
2	\$3b0c-Ecu identification Write/DID \$0C - Feature En... Write		(\$3b)	0c	NA	SLIF Enable	1	2
3	\$3b01-Ecu identification Write/DID \$01 - DTC Enabl... Write		(\$3b)	(\$01)	NA	DTC B356A 02	1	0
4	\$5a01-Ecu identification Read/DID \$01 - DTC Enabl... Read		(\$5a)	(\$01)	NA	DTC B356A 02	1	0
5	\$3b0c-Ecu identification Write/DID \$0C - Feature En... Write		(\$3b)	(\$0c)	NA	FDI Enable	1	3
6	\$5a0c-Ecu identification Read/DID \$0C - Feature En... Read		(\$5a)	(\$0c)	NA	FDI Enable	1	3

Created with the Personal Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

7.12 Label data base file with respect to Dut's

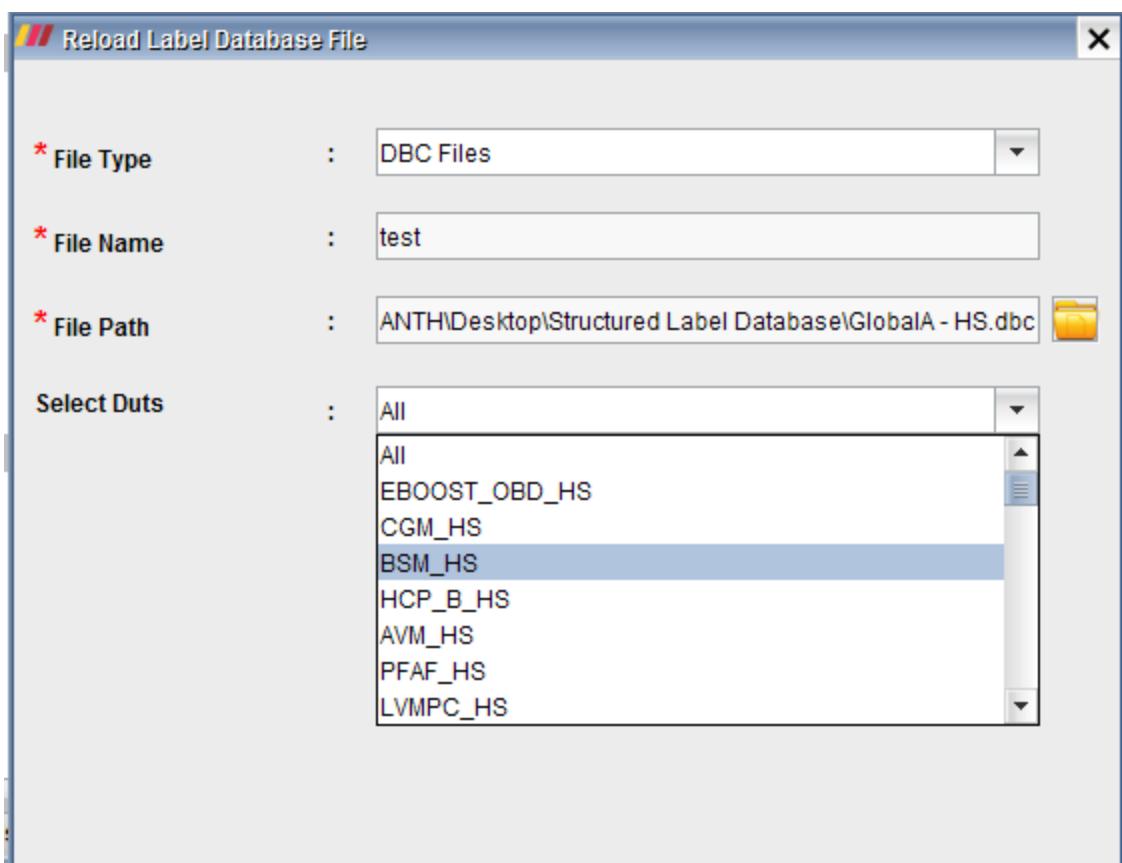
Description :

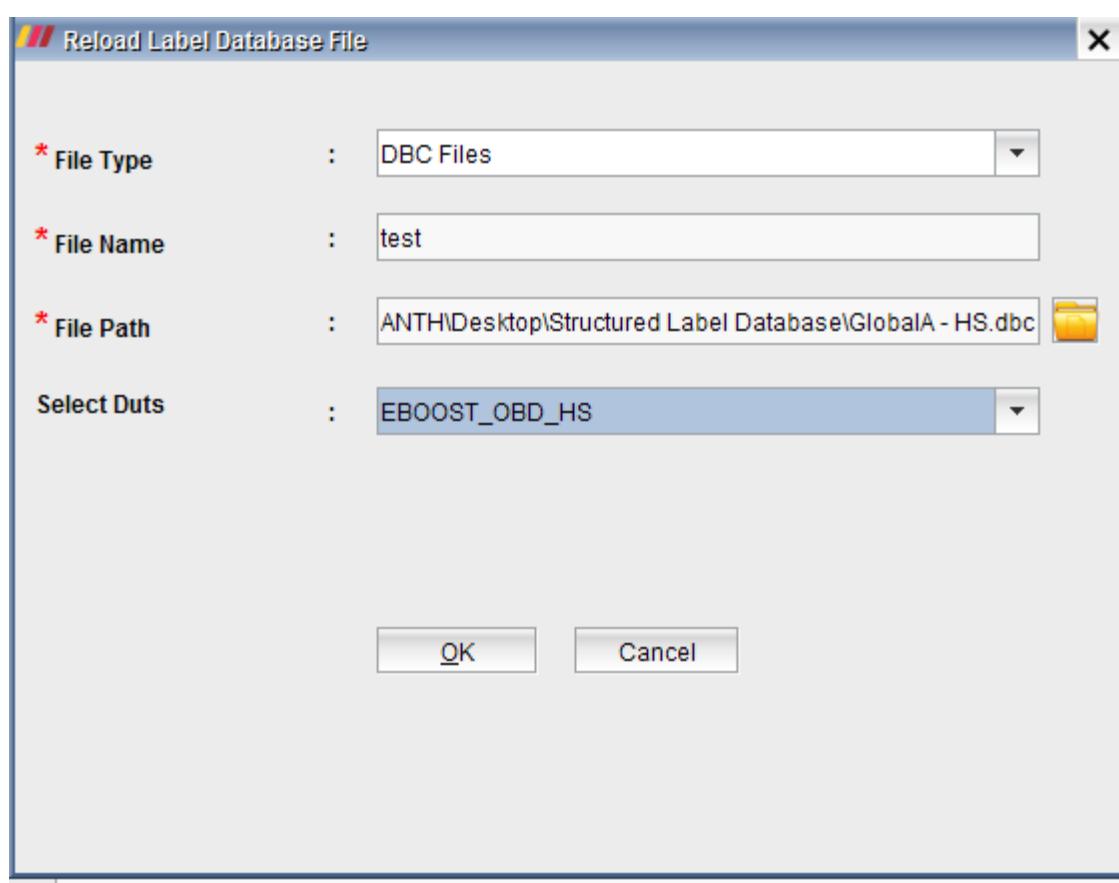
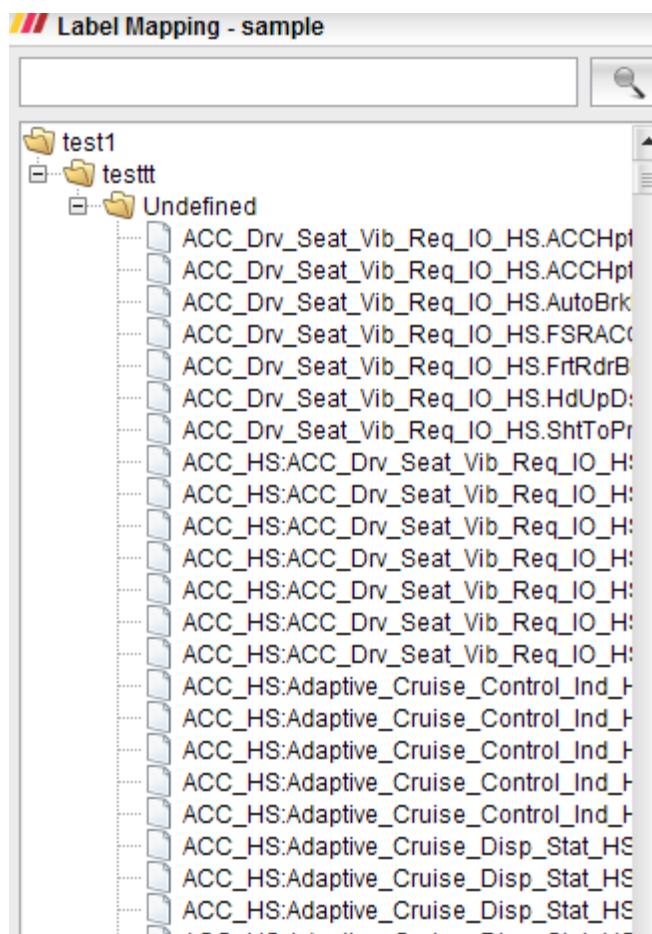
While extracting a label database file User is allowed to select a DUT from the drop down which consists of "ALL" along with the list of DUT's. When user selects "All", the type of label cannot be determined as MITE cannot detect whether the label is routing from a transmitter or receiver.

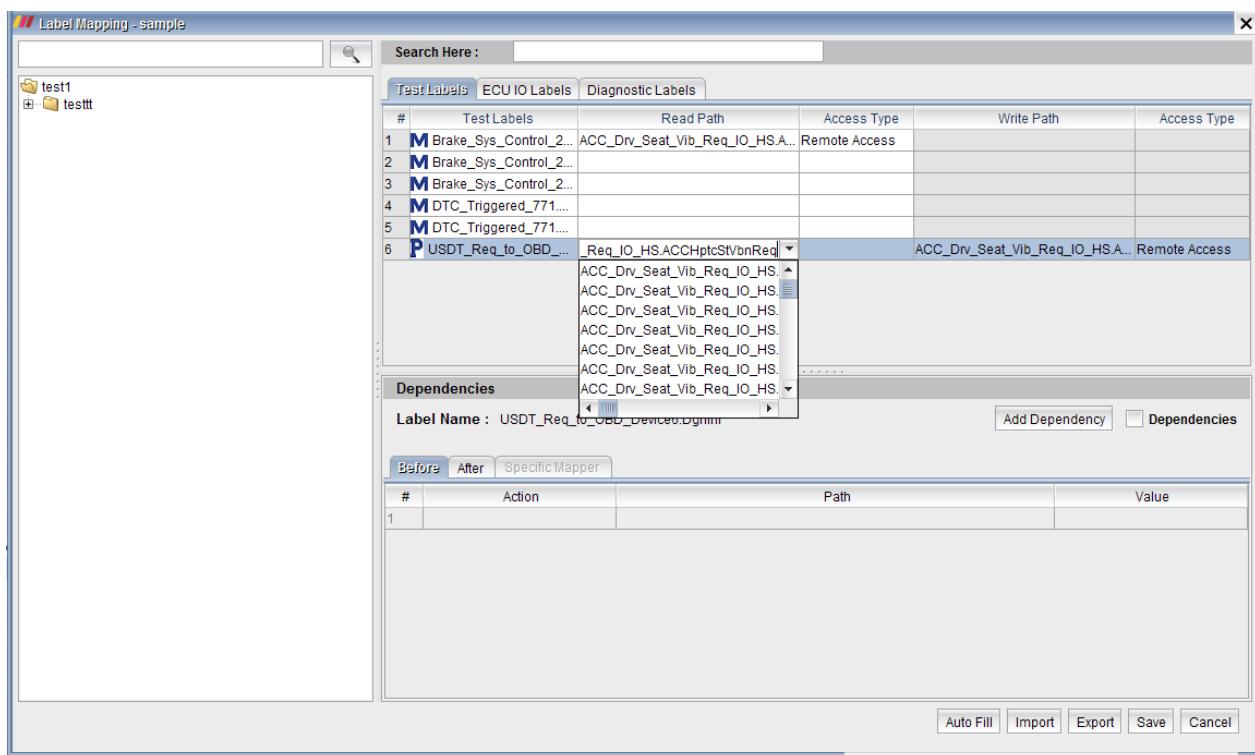
So, when user selects "All", all the labels are considered as undefined which allows user to enter both read path and write path in the Label Mapping Frame.

Instead of this, when user selects a particular DUT, label type is set to the label as "Measurement" or "Parameter" and allowing user to edit respective paths in the Label Mapping Frame.

To over come this problem user can select single DUT from the DUT's list instead of "ALL" as MITE allows import of same file multiple times with different Dut's.







Created with the Personal Edition of HelpNDoc: [Easily create EPub books](#)

7.13 PDX Extraction Changes in the version MITE - v2.1.6.0

Description:

Some labels are missing when PDX file has been imported till MITE - v2.1.6.0. In this version, the whole extraction process has been changed which was implemented following the spec rules and regulations. Due to this change, already used PDX labels will be no more present when user imports the same file with the version MITE - v2.1.6.0. Here user has to follow the below steps to make everything set again.

- 1) Reload the PDX file in the Label Data Base panel side.
 - 2) Already mapped old PDX labels will be shown in pink color in the Test suite labels tab. User has to update these old labels with the new labels from the reloaded file. This update option can be done for multiple labels using "Update Labels" in a single window in one go.
 - 3) User has to reload the same PDX file in the Test Bench Configuration panel side. Now, already mapped diagnostic read paths and write paths will become in pink color in the diagnostic labels tab.
 - 4) User can use the auto fill option here and save the data.

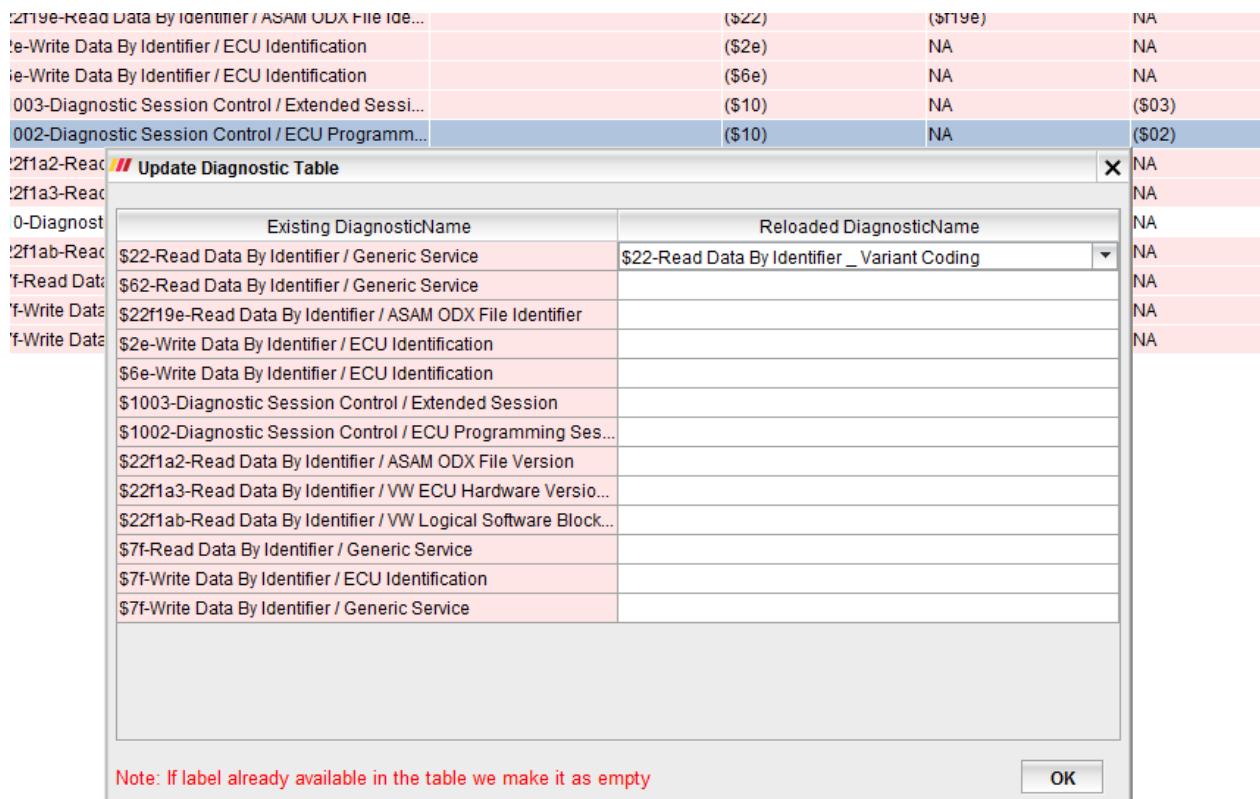
With this new extraction almost all the services are made available. Mainly, we have tested the execution part also for the services 1a, 3b, 22, 2e, 31 and 19. The corresponding test reports and the test suites that we have tested will be made available to the users in the PTC.

Created with the Personal Edition of HelpNDoc: Easily create CHM Help documents

7.14 PDX Extraction Changes in the version MITE -v 2.4.0.1

Diagnostic Extraction:

From MITE Version, MITE-v2.4.0.1, the format of PDX labels has been changed for better understanding and better performance. From now onwards, if Label Name contains “/” or “.” in the file we are replacing it with the “ ”.



With this change, all the PDX Labels will be shown in the pink colour in the labels tab as well as in the Label Mapping Frame. To set everything back to normal, user has to follow the below shown steps:

- 1) Reload the PDX file in the Label Data Base panel side.
- 2) User has to update these old labels with the labels from the reloaded file. This update option can be done for multiple labels using "Update Diagnostics" in a single window in one go.
- 3) User has to reload the same PDX file in the Test Bench Configuration panel side and user can use the option of Auto fill to get the mapping done automatically.

With this new extraction almost all the services are made available.

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

7.15 ARXML/FIBEX Arrayed Labels

ARXML/FIBEX Arrayed Labels

Generally, ARXML and Fibex files will have more arrayed labels than the original labels. So, due to this lengthy files get created in the back end which is sometimes responsible for MITE Screen becoming blank and becoming stuck.

Ex: if the label contains array size of 255, then we are generating the 255 labels with the change of the array length in the label name.

```
MapControl::showDestinations.tripDefinition.tripSegment[13].interpolationPointList[13].interpolationPointData.highwayEntranceExit.
geoLocation.entrancePointList[3].baseLocation.textDescription.textDescriptorList[7].value.phonetic
```

This label alone has approx. 100 labels with the different array size number.

To overcome this problems, in the new implementation, user has to enter the array size manually

for any arrayed label that is coming from ARXML/FIBEX files. In the extraction we will bring only the single/original label with '0' array as an example for the user to author the TestCases.

While writing in the Editor, we will validate if the entered label is in the valid range or not.
If not we will show the pop up that it is not in the valid range.

Parameter/Description

MapControl::showDestinations.tripDefinition.tripSegment[13].destination.optionLocation.geoLocation.baseLocation.textDescription.textDescriptorList[3].key

MapControl::showDestinations.tripDefinition.tripSegment[13].interpolationPointList[13].interpolationPointData.highwayEntranceExit.geoLocation.entrancePoin...

MapControl::showDestinations.tripDefinition.tripSegment[7].interpolationPointList[5].interpolationPointData.highwayEntranceExit.geoLocation.entrancePointLi...

MapControl::showDestinations.tripDefinition.tripSegment[7].interpolationPointList[5].interpolationPointData.highwayEntranceExit.geoLocation.entrancePointLi...
--

MapControl::showDestinations.tripDefinition.tripSegment[7].interpolationPointList[5].interpolationPointData.highwayEntranceExit.geoLocation.entrancePointLi...

MapControl::showDestinations.tripDefinition.tripSegment[7].interpolationPointList[5].interpolationPointData.highwayEntranceExit.geoLocation.entrancePointLi...

MapControl::showDestinations.tripDefinition.tripSegment[7].interpolationPointList[5].interpolationPointData.highwayEntranceExit.geoLocation.entrancePointLi...

Created with the Personal Edition of HelpNDoc: [Free Web Help generator](#)

8 MITE Parameters

Test Parameterization and label mapping are two powerful features that improve the reuse of test artifacts.

In MITE, Complete test data is categorized into two types,

1. Test Labels: Signals, variables and constants from database files like (.a2l, .dbc , .ldf , .odx etc.)
2. Test Parameters : User defined constants

Created with the Personal Edition of HelpNDoc: [Create iPhone web-based documentation](#)

8.1 Test Parameterization

1.1 **O**

verview

Test parameterization is a testing method utilizing parameters rather than fixed values. So one can execute the same test with different data sets (test cases) used as inputs or expected outputs that are stored separately from the test steps. Test Parameterization is a powerful practice to enhance the testing scenarios and reuse the same. During the execution of a test case the parameters are filled in with their actual value, so a parameterized variation of the original test case is produced.

The main benefits of parameterization are as follows :

- By externalizing the changing parts of a test case(s) as parameters it is easier to manage complex test cases
- User can automatically execute multiple variations of each test. If a parameter has multiple values provided that will cause several executions of each tests for each variations of the parameters.
- Test Parameterization allows sharing information between multiple test cases.

Sharing is easily achieved by defining parameter values in the test sets, which effectively and automatically shares these values in all test cases participate in the test set.

Created with the Personal Edition of HelpNDoc: [Easily create Qt Help files](#)

8.2 MITE interface

- Test Parameters Table Interface @Test case Definition Level.

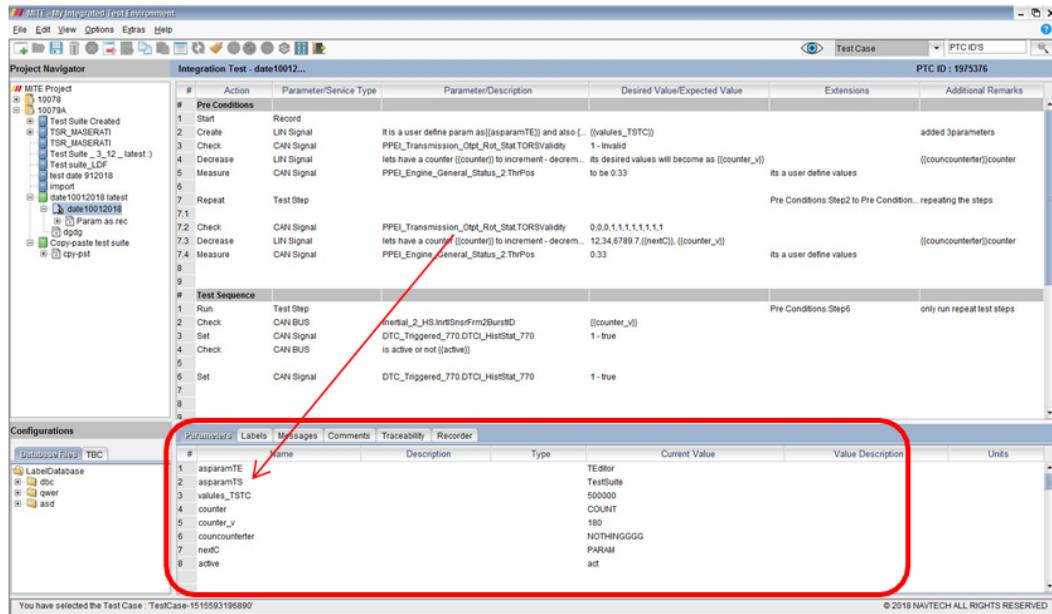


Figure 30: Test Parameters Table with data

- Parameters Table Editor Interface @Test suite Definition Level

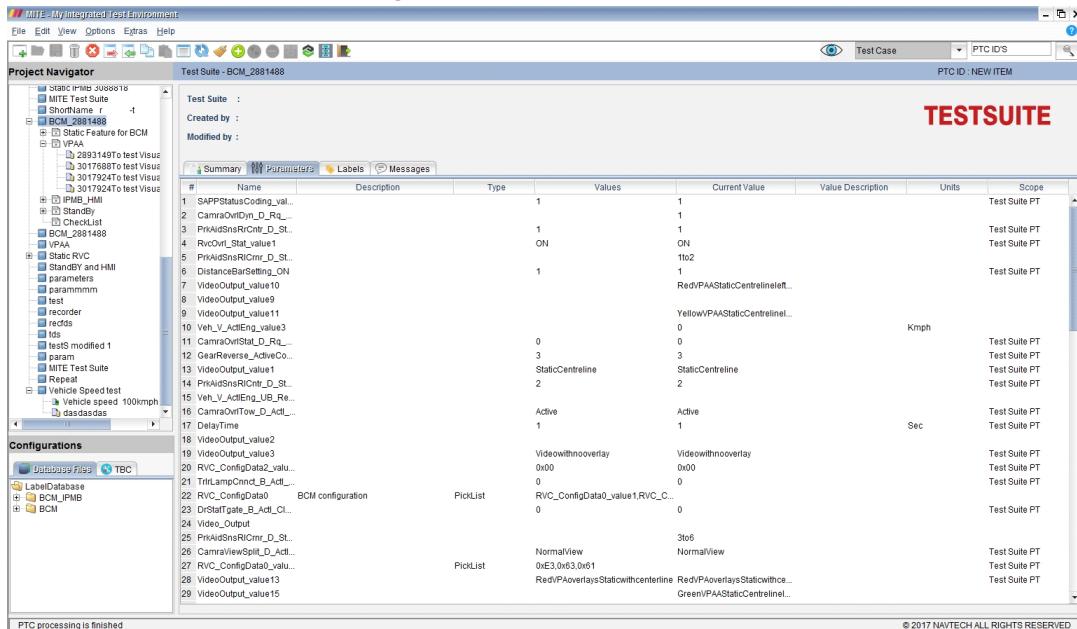


Figure 31: Test Parameters in Project Level

- Parameter Table Editor will have fields as shown below

S.No:	Field Description	Availability (Access type)
-------	-------------------	----------------------------

		@ Test Case	@ Test suite
1	Name	Yes (Read Write)	Yes (Read Write)
2	Description	Yes (Read Write)	Yes (Read Write)
3	Type(Pick List, String)	Yes (Read Only)	Yes (Read Write)
4	Values	Yes (Read Only)	Yes (Read Write)
5	Current Value	Yes (Read Write)	Yes (Read Write)
6	Value Description	Yes (Read Write)	Yes (Read Write)
7	Units	Yes (Read Write)	Yes (Read Write)
8	Scope	NA	Yes (Read only)

Table 8: Parameter Table Editor Fields

Created with the Personal Edition of HelpNDoc: [Free help authoring tool](#)

8.3 Using Parameters in Test Cases

Every test case can be parameterized by simply using parameters when editing the test case's definition. Parameter(s) added at this level(s) in a particular project will be reflected in all the levels of the project, since the parameter(s) consolidation is developed at project level also.

Irrespective of where/how the parameter is defined, list gets consolidated at test suite level and so it can be further used in any test case of that test suite with a different value or same value.

As user start typing the parameter name, MITE will show the list of matching parameters already defined in that test suite. And it will automatically add a parameter entry in the test case parameter table.

Created with the Personal Edition of HelpNDoc: [Easily create Help documents](#)

8.3.1 Adding a New Parameter

In MITE, A new parameter can be defined in the following ways:

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

8.3.2 Import Parameters from another Test Suite

This enables the user to import all the parameters which are present in another Test Suite into the currently selected test suite.

Select a Test-Suite and click on "Parameter" tab as shown in below figure,

- ✓ Click on "**Import Parameters**" icon in the tool bar
- ✓ A pop-up appears asking for "TestSuite ID" ; TestSuite ID from which user wants to import parameters.
- ✓ Provide Valid TestSuite ID and Click 'OK'
- ✓ All the available parameters will be imported into the currently selected Test Suite



Figure 32: Import Parameters at Test Suite Level

Note: This is only possible where there are no parameters present in the test suite

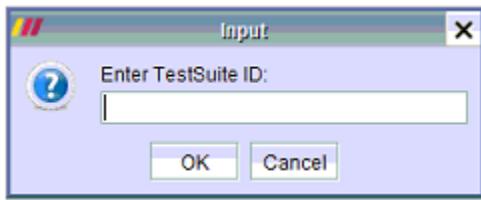


Figure 33: Enter Test suite ID to Import Parameters

Created with the Personal Edition of HelpNDoc: [Write eBooks for the Kindle](#)

8.3.3 Add / Remove parameter(s) at Test Suite Level

Select a Test-Suite and click on “Parameter” tab as shown in below figure, To add parameters:

- ✓ Right Click on Test Suite – Parameter tab
- ✓ Select on Add Parameter
- ✓ Fill in the particulars and Click ‘OK’

- **Add Parameters using “+” icon from Tool bar menu also**

To delete parameters:

- ✓ Right Click on Test Suite – Parameter tab
- ✓ Select on Remove Parameter

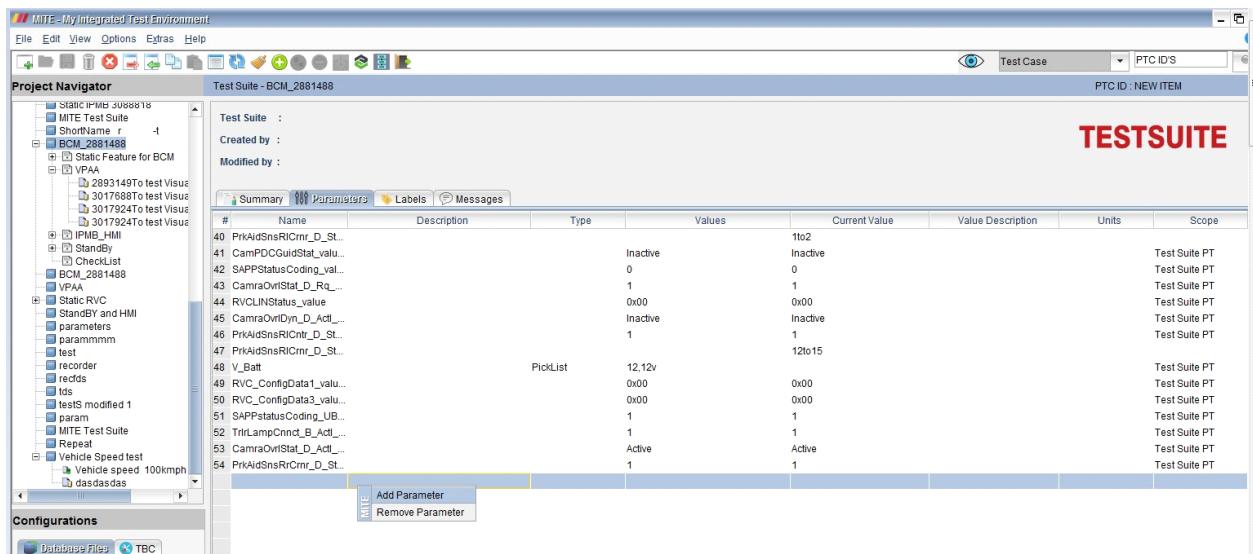


Figure 34: Add / Remove Parameters at Test Suite Level

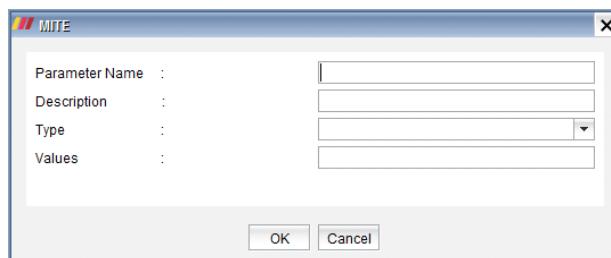


Figure 35: Parameter details

Created with the Personal Edition of HelpNDoc: [Full-featured EPub generator](#)

8.3.4 Add / Remove parameter(s) at Test Case Level

Test Case Parameter table: Right Click on Test

Case – Parameter Tab To add parameters:

- ✓ Select Add Parameter
- ✓ Fill in the particulars and Click 'OK' as shown

in above Figure:28 To delete parameters:

- ✓ Right Click on Test Case
- ✓ Select on Remove Parameter

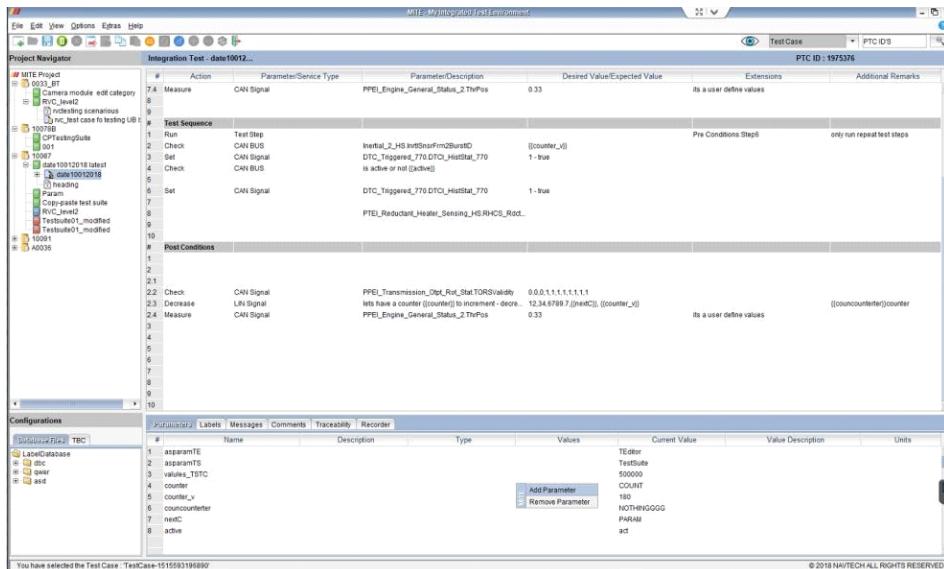


Figure 36: Add Parameters at Test Case Table Level

Created with the Personal Edition of HelpNDoc: Easily create EBooks

8.3.5 Add parameter(s) in Test Case editor

1. While authoring a test case user can define or create a parameter(s). By following the parameter naming conventions (mentioned at section - Parameter Naming Convention).
2. In test step, when user enters a new string which is not available in label database files(.A2L,.DBC,.LDF) and in the existing list of parameters in that project, MITE creates a parameter and automatically adds it to the existing list of parameters.
3. Parameters can be used in the following locations, while authoring the test case
 - i. Description of the test-case
 - ii. Pre-Conditions , Post- Conditions and Test Sequence sections of a test-case
 - iii. In Parameter/Description, Expected/Desired Value , Extension and Additional- Remarks fields of each test-step/test-line

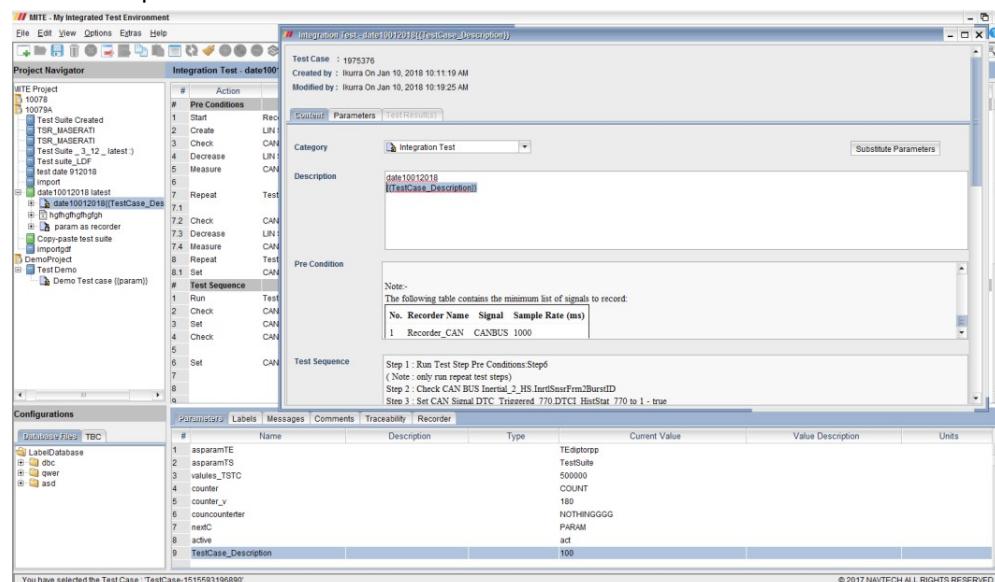
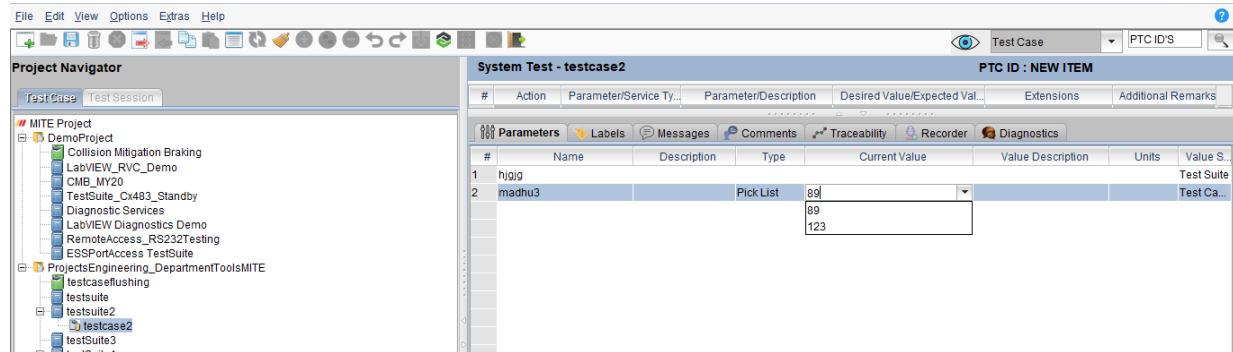


Figure 37: Add Parameters in Description of a Test Case

Created with the Personal Edition of HelpNDoc: Full-featured EPub generator

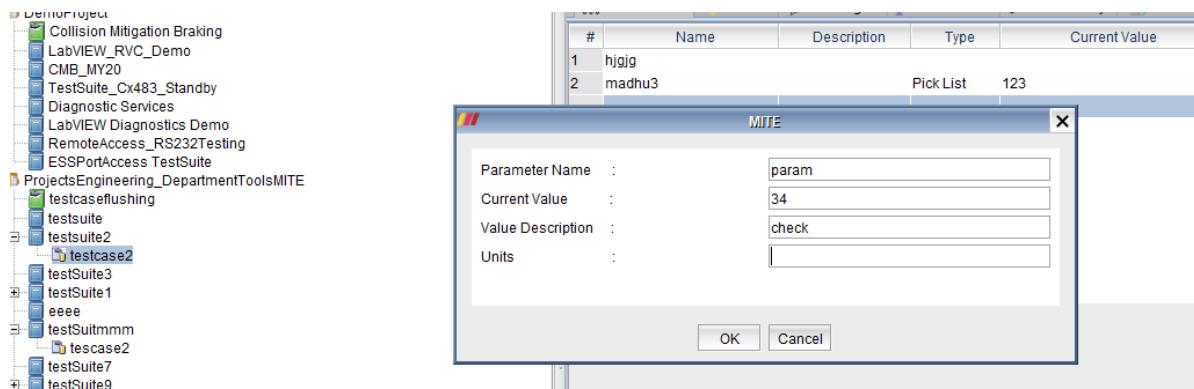
8.3.6 Edit parameter save list option

Step 1:-If User changes parameter's current value in test case level or suite level.

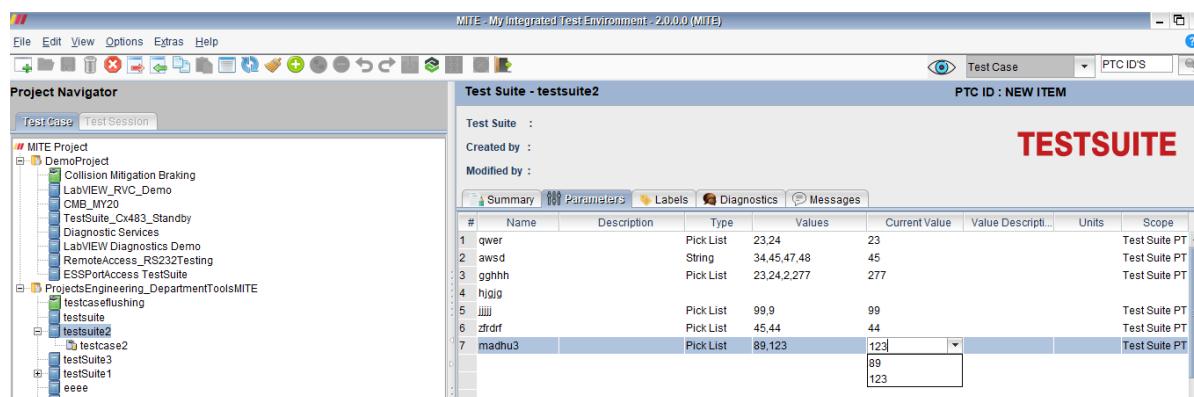


(Or)

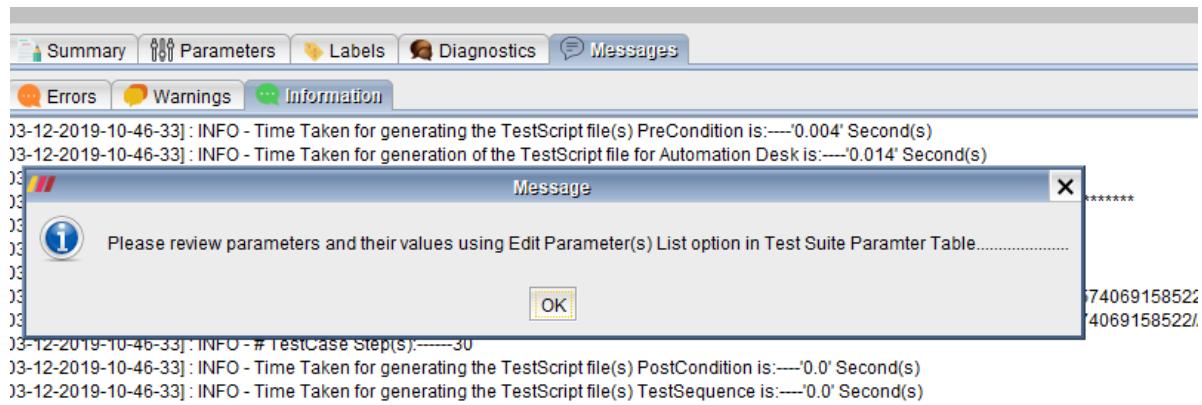
When User add new parameter by right click in test case or suite level then following pop up will be shown:



Step 2:-If user add or update current value in test case level after that he/she have to save(ctrl+s) and then go to suite level here user have to save again.

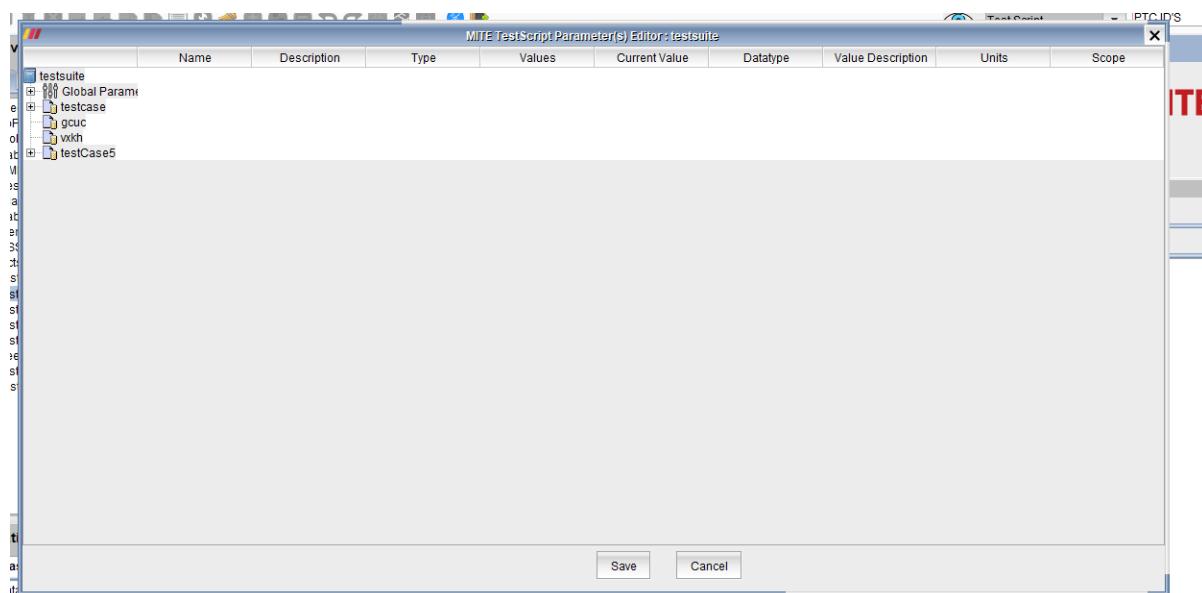


Step 3:-When User is adding new parameter in Test Case or Test Suite level for the execution of Script generation then he/she will get the following popup.

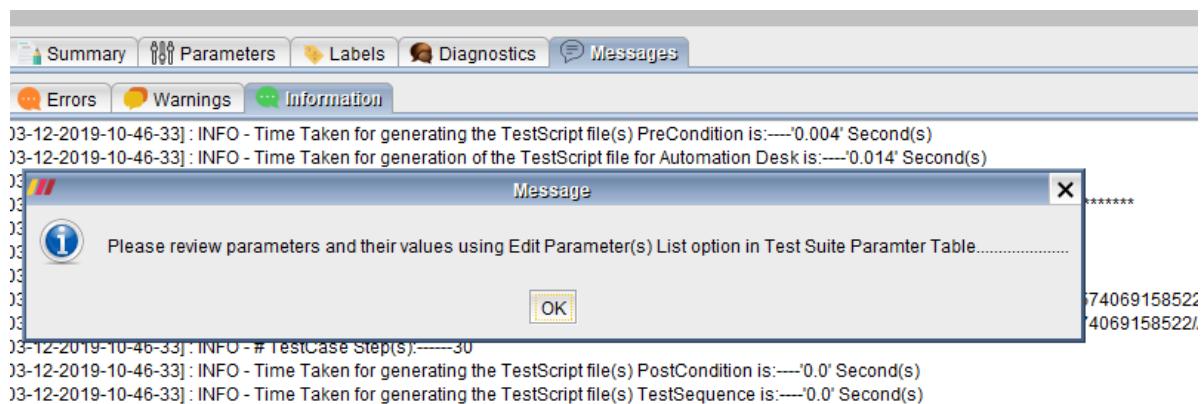


Step 4:- After clicking on ok button it should go to the Edit Parameter Dialog Box as shown below.

If User click on Cancel button without saving the task then he/she will give a pop message but after saving if he/she clicks cancel it should close the dialog Box.



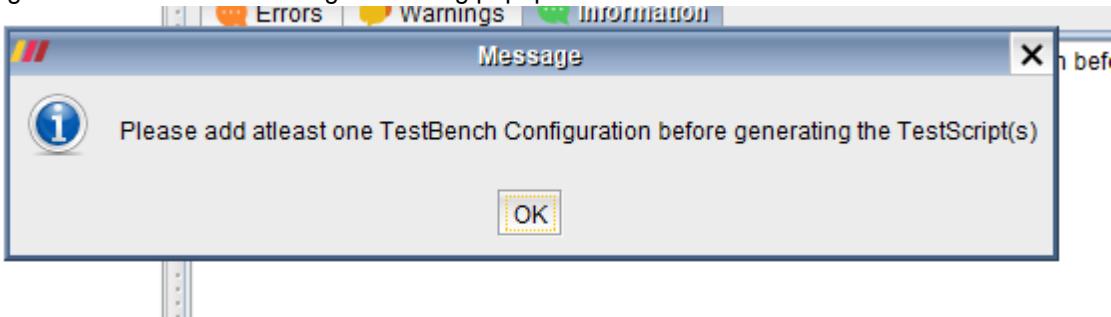
Step 5:- If User edits current value in Test case or Suite level then at the time of script generation he/she will get the following popup shown below:



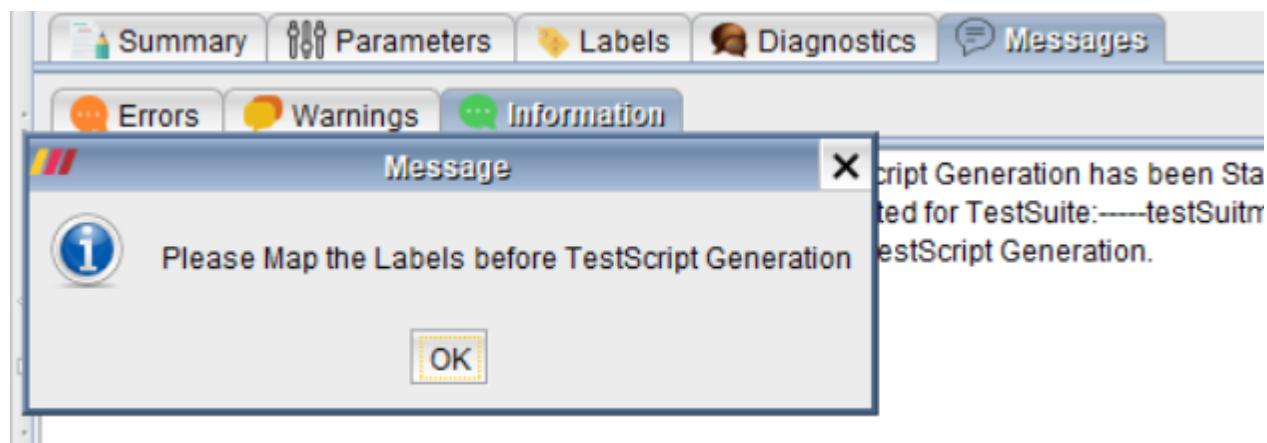
Label Mapping:-

Step 1:- If User is having Test Cases and shifts to script level at that time he/she have to add

TBC configuration files otherwise will get following popup:



Step 2:-If there is any label mapping file and are not mapped in script level then at that time of script generation User will get the following pop message.



Step 6:-After map the Label user get generate the script .

Created with the Personal Edition of HelpNDoc: [Free EBook and documentation generator](#)

8.3.7 Parameters Value SetIn

User define parameters can assigned with different values as per system requirement , Value can be used from Test Suite or Test Case across a test case, this will be automatically updated in test case under “Value SetIn” column in test case Parameter table, as shown in below picture.

A parameter values deals with the following characteristics in a test case:-

- i. Parameter value can be defined at Test suite level and also at Test Case level
- ii. A value defined at test suite level will have its “Value SetIn as Test Suite”
- iii. A value defined at test case level will have its “Value SetIn as Test Case”
- iv. When “Value SetIn is Test Suite” , parameter value will have all the properties as defined at test suite level
- v. When “Value SetIn is Test Case” , parameter value will have all the properties as defined at test case level
- vi. User can switch between Value SetIn test case /test suite as per test case requirement

Note:

1. Value SetIn as Test Case is local to the respective test case
2. Value SetIn as Test Suite is global and can be accessed across all the test cases of the test suite
3. Once the value changed from “Value SetIn is Test Case” to “Value SetIn is Test Suite” – parameter value given at test case level cannot be retrieved back

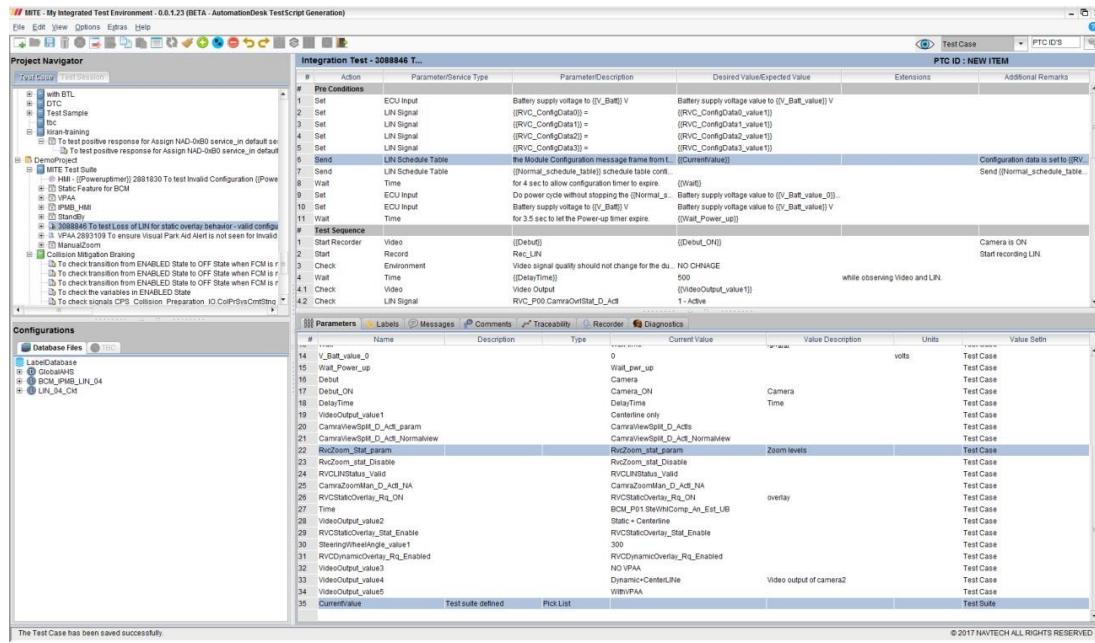


Figure 38: Parameters Value-SetIn

Created with the Personal Edition of HelpNDoc: [Free HTML Help documentation generator](#)

9 Parameter Naming Convention

To define a new parameter, the following are the rules for the naming convention Parameter Name,

1. Should be a single string.
2. Parameter syntax should start with Double Open braces '{' and close with Double Closed braces '}' only.
3. Parameter Name should start only with alphabet(s).
4. Parameter Name should be more than three(3) characters.
5. Numeric and Special characters '.' and '_' are allowed(only in between the Parameter Name)
6. No blank space(s) allowed in between the Parameter Name.
7. Length of string should not exceed 40 characters.

Created with the Personal Edition of HelpNDoc: [Produce online help for Qt applications](#)

9.1 Providing Parameter values in Test Cases or Test Sets

Assuming that user have entered some parameters in the Test Cases definition(s), some of those parameters still don't have values filled at test suite level table and test case level table, their current value and the list of possible values of the parameters can be provided at each test case level by editing it in the "Test Parameters" section of the respective test case. Values can also be assigned to it at test suite level, the similar "Test Parameters" section also appear at a test suite level. These values provided on a test suite will be shared and available in all test cases which are part of that suite.

When a parameter is used in two or more test cases with different current value, MITE will automatically consolidate the list of current values and adds them to value list of that test case. After that, if that parameter is used in other test case, dropdown of value list will be shown in parameter table to enable quick selection.

Created with the Personal Edition of HelpNDoc: [Write eBooks for the Kindle](#)

9.2 Rename parameter(s)

Already defined parameter can be re-named as per user requirement. This operation is possible at Test-Suite level and will reflect this change in all the Test-Case(s) present within the Test-Suite.

Steps to re-name a Parameter:-

- i. Select a Test Suite and click on its "Parameter" tab window
- ii. Click on a parameter which need to be re-named and start rename/edit it directly
- iii. Press Tab-key or Enter-key ; a pop-up will appear to confirm parameter rename operation through-out the test suite entries
- iv. Click on 'Yes' to apply the changes made

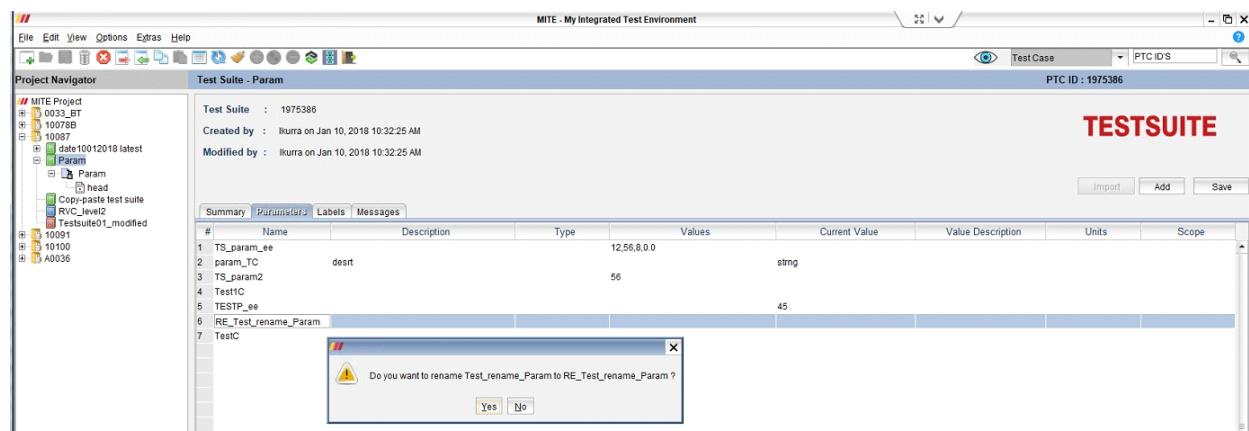


Figure 39: Re-name Parameter

Created with the Personal Edition of HelpNDoc: [Create cross-platform Qt Help files](#)

9.3 Reviewing and managing Parameter configurations

Once user starts using parameters and provide values, user might come across few questions, such as:

- What are the parameters used in the current test case or test suite?
- What are the actual values user has provided to the current test case or test suite?

The "Parameter" section shown on the Test case editor page and test suite editor page shows this information.

Test Case Level Parameter Table:

1. Test case parameter table will show list of parameters used in respective test case.
2. For Parameters which are already defined at test suite level, MITE will pick all the field values from test suite table by default. User can modify some of field values as per the definition shown in table 1.
3. For the parameters which are newly defined, user will have to fill the required field values. Test Suite Level Parameter Table:

1. Test suite parameter table will show list of parameters used in all the test case under that test suite.
2. User can import the parameters which are defined in other test suite just by providing the Test- Suite ID, of the test suite which is not imported in MITE.
3. User can copy selected parameters from other test suites
4. User can change the scope of any parameter definition between Test case, Test suite and Project using simple selection list.
5. User can add multiple values to a parameter which are available as a picklist in current value field at test suite and test case.

Created with the Personal Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

9.3.1 Label/User-define Parameters in MITE:

All the used Label parameters used while Test case authoring will be listed in **Tab -Label**

The screenshot shows the MITE software interface with the following details:

- Project Navigator:** Shows the project structure with nodes like 'MITE Project', '10078', '10079A', 'Test Suite Created', 'TSK_MASDPA1', 'TSK_MASDPA1', 'TestSuite_3_12_(latest)', 'Test suite_LDF', 'Test date 912018', 'Import', 'date10012018 latest', 'date10012018', 'Param as rec', 'ddpg', 'Copy-paste test suite', and 'cpy-pst'.
- Test Case:** The current test case is 'Integration Test - date10012...' with PTC ID: 1975376.
- Table Headers:** The main table has columns for '#', 'Action', 'Parameter/Service Type', 'Parameter/Description', 'Desired Value/Expected Value', 'Extensions', and 'Additional Remarks'.
- Table Data:** The table contains several rows of actions, such as Start, Create, Check, Decrease, Measure, Repeat, and Check, each with its corresponding parameters and descriptions.
- Configurations:** A separate section shows a table of parameters with columns for '#', 'Name', 'Common Name', 'Default Value', 'Factor', 'Offset', 'Units', 'Min. Value', 'Max. Value', 'Data Type', and 'Description'.
- Message Log:** The bottom pane shows a log with entries like 'You have selected the Test Case : TestCase-1515593196890' and copyright information '© 2018 NAVTECH ALL RIGHTS RESERVED'.

Figure 40: Label parameters

Created with the Personal Edition of HelpNDoc: [Free help authoring environment](#)

9.3.1.1 Label Parameters

All the used Label parameters used while Test case authoring will be listed in **Tab -Label**

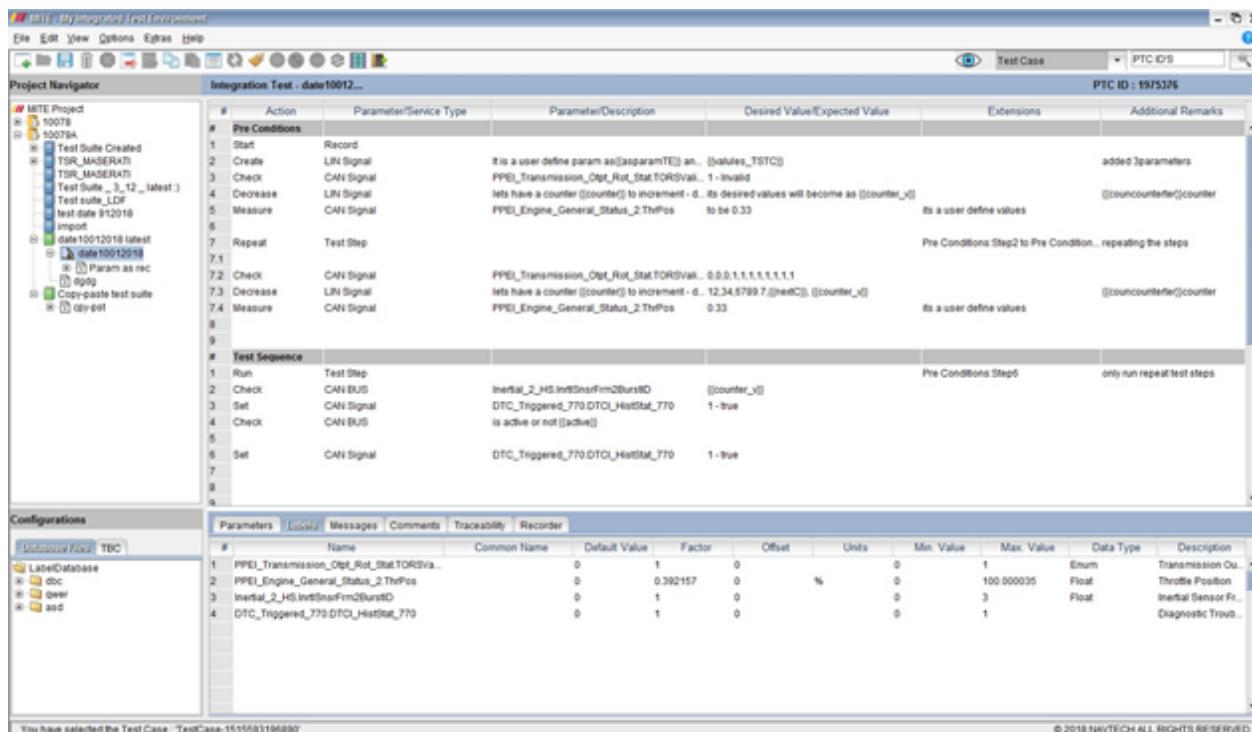


Figure 40: Label parameters

Created with the Personal Edition of HelpNDoc: [Free Kindle producer](#)

9.3.1.2 User Define Parameters

The User defined parameters used while Test case authoring will be listed in **Tab - Parameter**

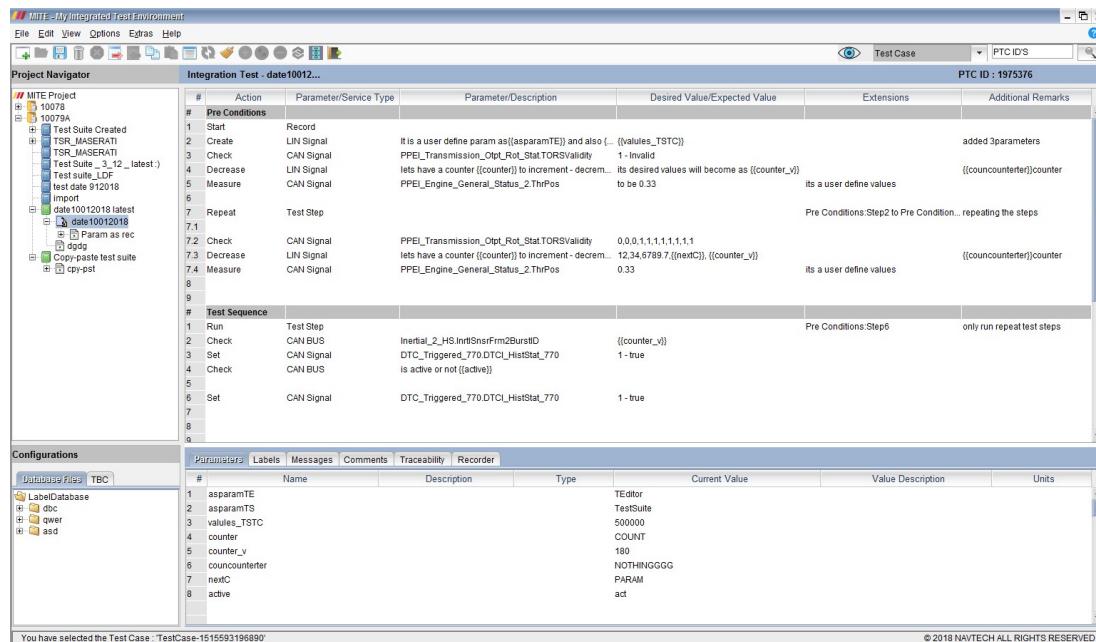


Figure 41: User defined parameters

Created with the Personal Edition of HelpNDoc: [Easy EPub and documentation editor](#)

9.4 Parameter Common Name

This provides a provision to append a “common name” to use/reuse Parameters and Labels together in a test case.

Steps to add common name:

1. Define a parameter and a label in same test case
2. Attach parameter name to the label in “Common Name” column from that drop-down list as shown in below figure
3. Select the required parameter as common name and press Tab-key
4. This operation will change/replace the “Current Value” of the selected parameter as common name in parameter table
5. On using same Common name for more than one Label , a warning pop-up to confirm from user

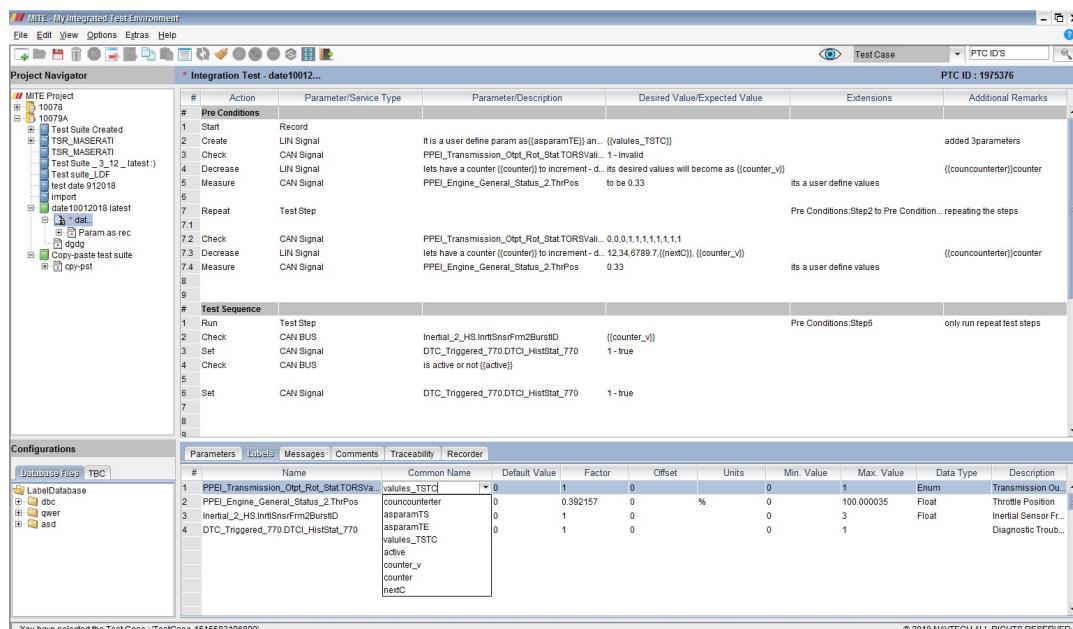


Figure 42: Common Name Parameter

Created with the Personal Edition of HelpNDoc: [Free EPub and documentation generator](#)

9.5 PTC Content:

When content is pushed to PTC,

1. The parameter will be automatically recognized by MITE in the test case content, and the "{{parametername}}" placeholder will be automatically placed in the content that is pushed to PTC.
2. Parameter Value Table at test case level, will be populated with the parameters which are used only in that respective test case.
3. Parameter Value Table at test suite level, will be populated with the parameters which are used in two or more test cases with same value across the test suite.
4. Parameter Table at test suite level, will be populated with the parameters which are used in two or more test cases with different values across the test suite or which has multiple values after consolidation at test suite level.

5. The following hierarchy is followed while updating the PTC parameters content, so that remaining fields updates are automatically taken care by the process and PTC.

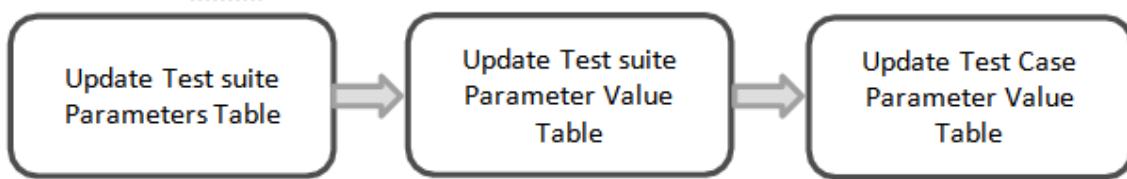


Figure 43: PTC Contents

User will have option to review the PTC test case content in MITE itself (Content, Parameters, Test results).

Substitute parameter option will be available in MITE also, to see actual test case content in PTC format after replacing parameters with their values.

Created with the Personal Edition of HelpNDoc: [Write eBooks for the Kindle](#)

10 Create Test Case

Under selected Test Suite only a Test Case can be added.

Steps to Add Test Case:

1. Right Click on the selected Test-Suite
2. Select option – “Create Test Case”

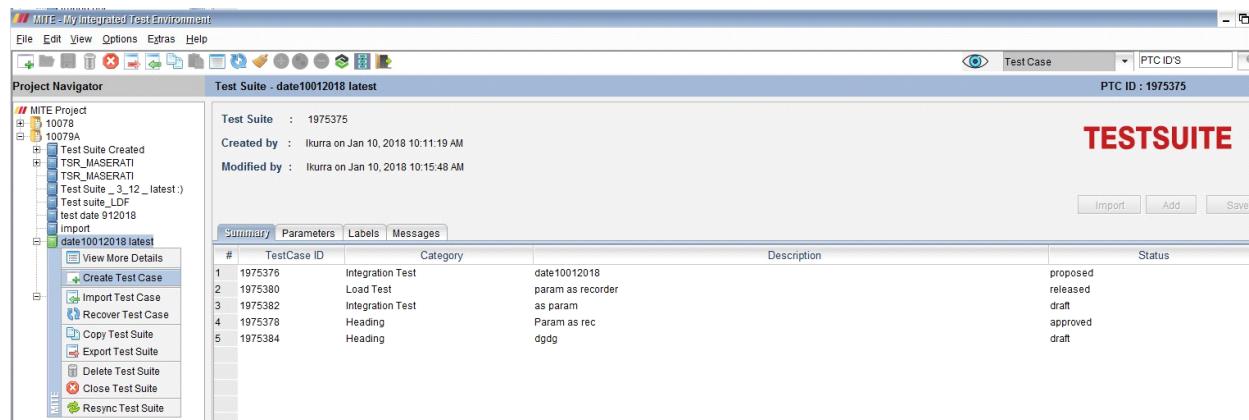


Figure 44: Add Test Case

Created with the Personal Edition of HelpNDoc: [Easily create Qt Help files](#)

10.1 Fill Add Test Case particulars

- i. For adding Test Case successfully the below fields are require to fill and then click on “OK” button.

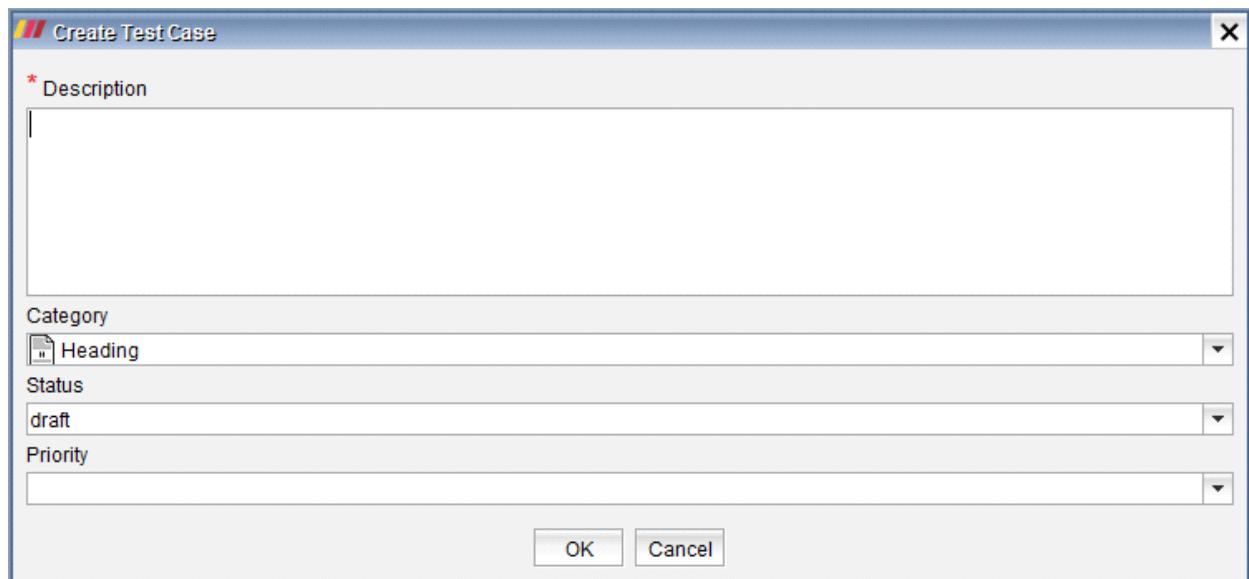


Figure 45: Test Case particulars

On adding Item	Fields	Operation performed
Create Test Case	Description	Description in words
	Category	Select the required Category from the drop-down list
	Status	Select the required Status for the same
	Priority	Select the required Priority for the same

Table 9: Create Test-Case(s) Particulars

- ii. The added Test Case will be listed under Test Suite
- iii. Expand the Test Suite (by clicking on the '+' symbol) to view all the Test case(s) present under it.

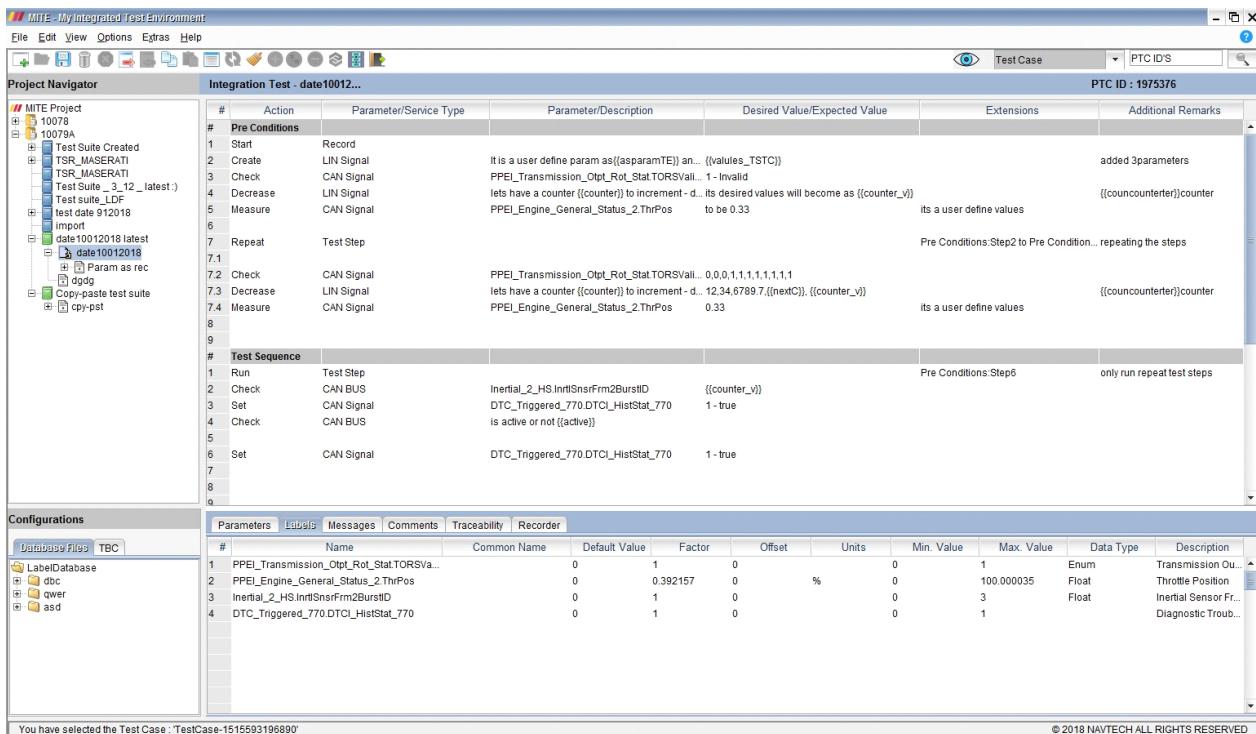


Figure 46: Test Case Editor

Created with the Personal Edition of HelpNDoc: [Full-featured EPub generator](#)

10.2 Options available on Test Case-Right click

The options available on Right click on “Test Suite” are as below:

Select on Item	Options available on Right click	Operation performed
Test Case	View More Details	Test Case details
	Create Test Case	Creates a new test case(as its child node)
	Copy Test Case	To copy a Test Case
	Paste Test Case	To paste the copied Test Case
	Export Test Case	Export a Test Case into Zip file
	Delete Test Case	Delete selected Test Case. Note – This option is only available for Test case(s) with NO PTC-ID

Table 10: Test Case-Right click options

Created with the Personal Edition of HelpNDoc: [Easy CHM and documentation editor](#)

10.3 Delete - Recover operations on Test-Case

Delete/Recover options are available in right-click options only. A deleted Test-Case can only be added back into the respective tree by using "Recover" option only

[For more information Please refer to section 13 Delete/Recovery Operations]

Delete - Recover operations can be performed on two levels : Test-Suite and Test-Case

i. Test-Case Delete :

To delete a Test-Case , select the respective Test-Case from its respective Test-Suite folder/tree. Right-Click and click on "Delete Test-Case",Test-Case will be deleted from the project.

On successful deletion, the Test-Suite folder/tree and its summary will perform a refresh operation which will update the Test-Suite tree and its summary table accordingly.

ii. Test-Case Recover :

To recover a Test-Case, select the respective Test-Suite. Right-Click and click on "Recover Test-Case", a dialog box will appear with the list of recently deleted Test-Case(s) select the require Test-Case and click OK. Test-Case will be added to the Test-Suite tree.

On successful recovery, the Test-Suite tree and its summary will perform a refresh operation which will update the Test-Suite tree and its summary table according to the latest tree structure.

Created with the Personal Edition of HelpNDoc: [Easily create Web Help sites](#)

10.4 Add the capability in MITE to insert images to test cases

Contents

1.Please do follow to invoke additional remarks window

2

2.Adding single image file into Additional remarks window

3

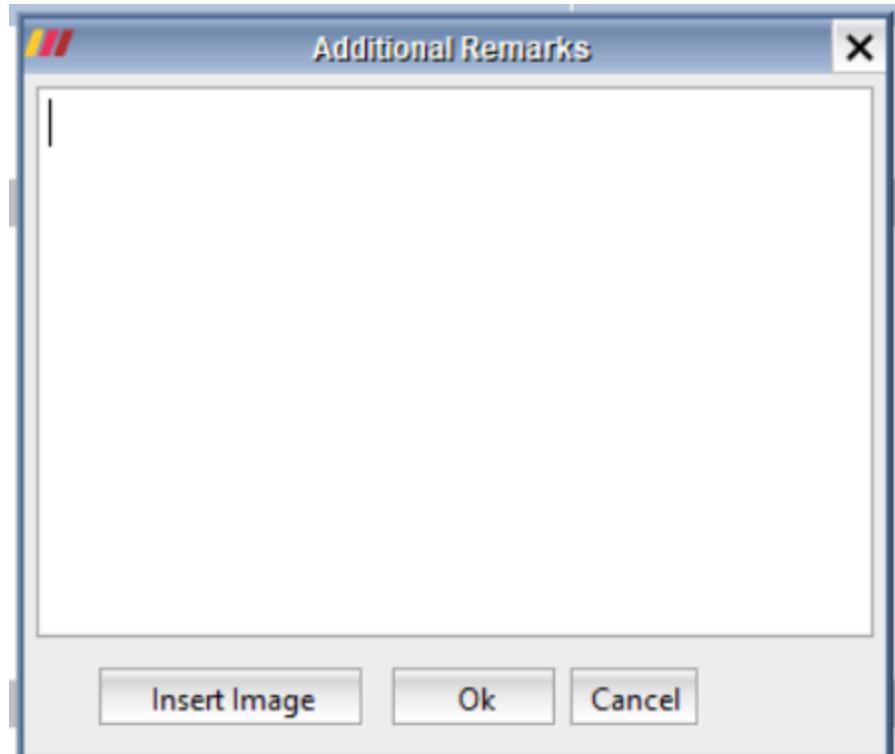
3.Adding image file into test case editor

4

Add the capability in MITE to insert images to test cases

Earlier user can add only description. But through this enhancement user can able to save image files and can add reference image file path locations in test case editor.

1.Please do follow to invoke additional remarks window

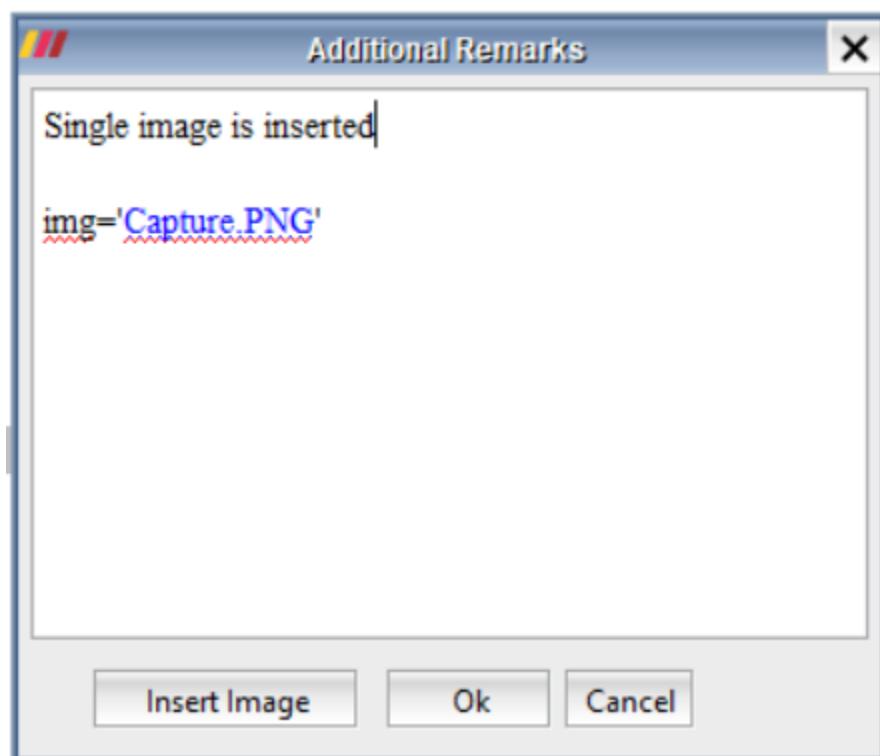


IMG: Additional remarks window will invoke

2.Adding single image file into Additional remarks window

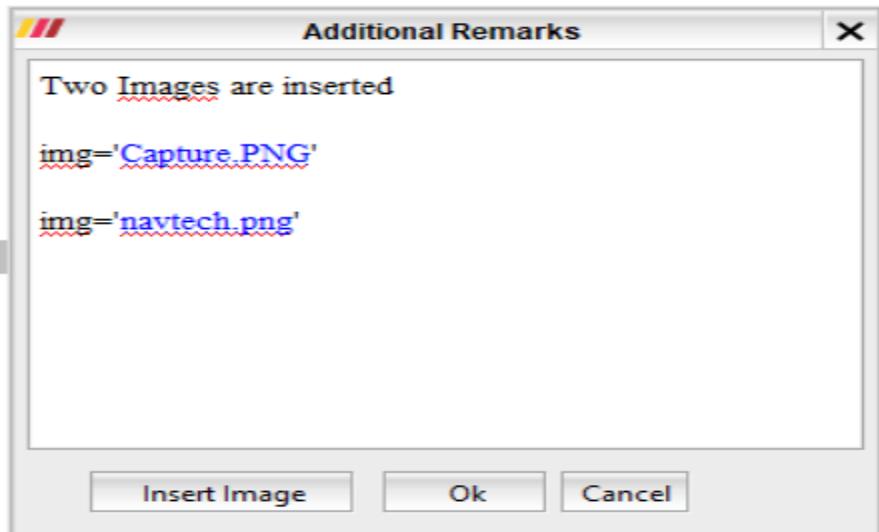
To insert images should click on **insert image** button in window. After that a file chooser will invoke. We should select an image file from that file chooser.

Then we can see a image file path in that window as like below .User can write his/her customized messages about belonging images.



IMG: Window with image file

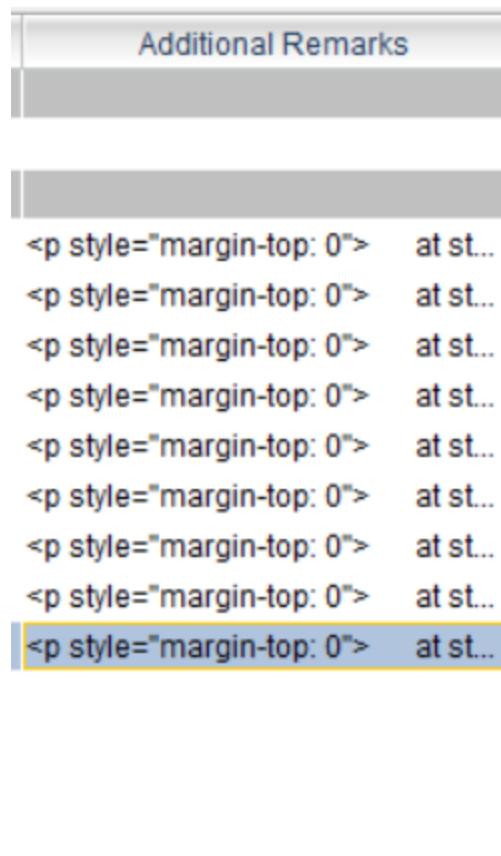
To insert multiple image files we need to follow above procedure multiple times. Then we can add multiple images as like below



IMG: Window with multiple image file

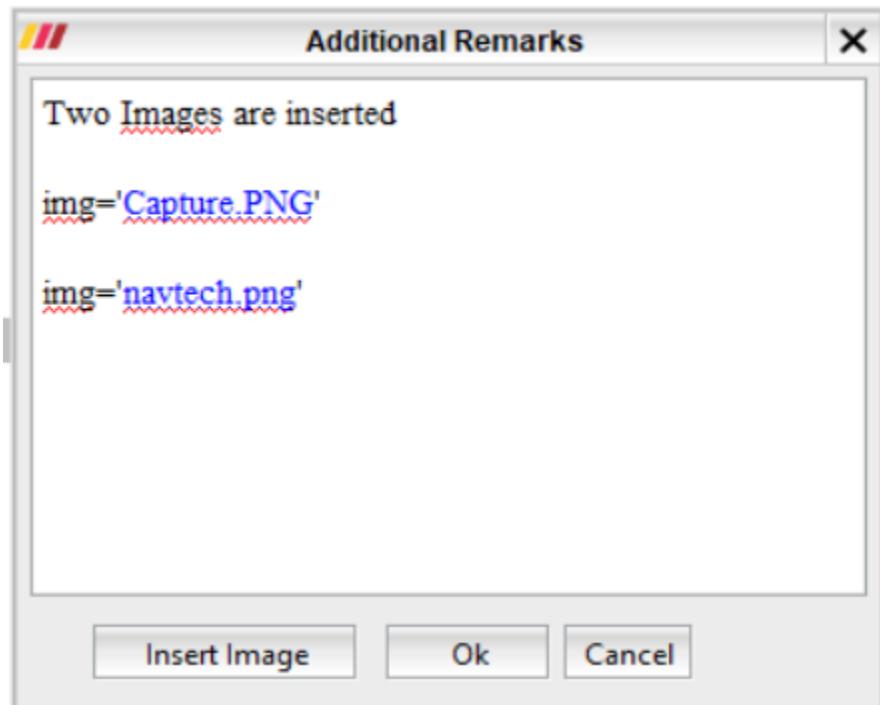
3.Adding image file into test case editor

To add content from additional remarks window to test case editor we should use **OK BUTTON** beside of **Insert image button**. Then the data will add into test case editor.



IMG: Add content into test case editor

While user double click in additional remarks column from test case editor, If data is already exists then window will invoke with existed content including image files as like below.



IMG: Window with multiple image file.

Created with the Personal Edition of HelpNDoc: [Easily create iPhone documentation](#)

11 Test Case Editor

On selecting Test case the MITE Editor will be shown. Using this editor only, Test Case authoring can be done.

Steps to follow for Test Case authoring:

1. Select the Test Case
2. Click on the test case editor window (which displays on right side)
3. Select the required authoring section and start entering the data/information into the editor
4. Click on the “Save” icon(on top left corner of the MITE window) or “Ctrl+S” to save the data present in editor window

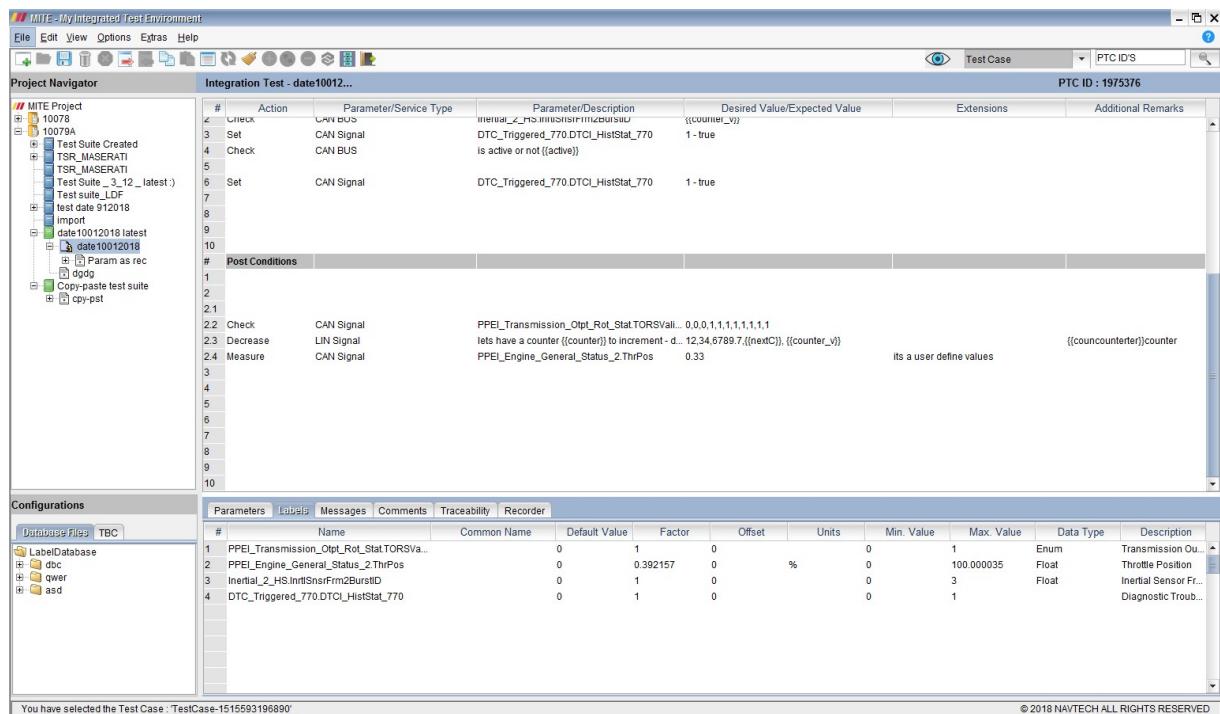


Figure 47: MITE /Test Case Editor

Created with the Personal Edition of HelpNDoc: [Free CHM Help documentation generator](#)

11.1 Elements of Test Case Editor

Select on Item	Elements	Purpose
Test Case Editor	MITE project(s)	List of project mapped and Test-Suite(s)/Test-Case(s) under it, in a tree structure
	Configurations	Imported Database files list
	Action	List of all the available actions that can be performed/used
	Parameter/Service Type	List of all the services that can be performed/used
	Parameter/Description	Labels and parameters definition and description used in test case authoring
	Desired Value/Expected Value	Values definition to the respected parameter/label used
	Extensions	Information definition require for some Actions selected
	Additional Remarks	Remarks added to that particular test line will appear in braces as "(additional remarks)"
	Pre Conditions	Defines the test pre-condition section
	Test Sequence	Defines the test sequence section
	Post Conditions	Defines the test post-condition section
	Parameters	List of parameter(s) used in test case authoring

Bottom Table	Labels	List of label(s) used in test case authoring
	Messages	Error(s), Warning(s), Information messages appeared during Test case authoring and PTC submit
	Comments	Comments added to a particular Test step/test case
	Traceability	List of requirement documents and requirement IDs of the project
	Recorder	List of recorders used in test case authoring
Configurations	Label Database	LDF,DBC,A2L files can be imported

Table 11: Elements of Test Editor

Created with the Personal Edition of HelpNDoc: [Easily create HTML Help documents](#)

11.2 Key-Words for Test-Case(s) Authoring

To make Test-Case(s) authoring more effective and simple “Drop-Down” list(s) are created as keywords for “Action” and “Parameter/Service Type”.

Using these keywords a Test-Case can be developed in a

well-mannered way. The below tables contains all the

available keywords and its functionality:

Test Case Editor	Item	Function
Action	Check	To verify the result in expected results
	Set	To set value(s) to parameters
	Create	To create new object
	Increase	To increase from one value to another by some factor
	Run	To run the mentioned test-step(s)
	Decrease	To decrease from one value to another by some factor
	Wait	To keep waiting for ms or sec at a particular test-step
	Measure	To measure the parameter current value
	Start	To start
	Stop	To stop
	Send	To send an instruction such as Diagnostic request
	Pause	To pause recording a video temporarily
	While	To use for loops to repeat test steps for a particular count
	Repeat	To repeat multiple test-step(s)

Table 12: Drop-Down list of Action

Test Case Editor	Item	Function
Parameter/Service Type	CAN Signal	Available CAN signals from DBC file
	CAN BUS	Available CAN bus from DBC file
	CAN Message	Available CAN messages from DBC file
	LIN Signal	Available LIN signals from LDF file
	LIN Message	Available LIN Tables from LDF file
	LIN Bus	Available LIN bus from LDF file
	LIN Schedule Table	Available LIN Schedule Tables from LDF file
	ECU Variable	Available variables from A2L file
	ECU Calibration	Available variables Calibration from A2L file
	Image	Already saved image as validation
	Output	Output value
	DiagRequest	Request for diagnostics service
	DiagResponse	Response of diagnostics service request sent
	Test Case	Authored test case
	Time	To set/give time at a test step
	Test Step	Authored test step
	Environment	Set of steps
	Function	A particular function call
	ECU Output	Available measurement from A2L file
	ECU Input	Available measurement from A2L file
	Recorder	To enable Recorder(s) from the recorders' list
	Fault	To mention fault category
	Video	Operations on video to record start/stop/pause

Table 13: Drop-Down list of Parameter/Service Type

Created with the Personal Edition of HelpNDoc: [Qt Help documentation made easy](#)

11.3 Additional Remarks

Remarks can be added to each test step by “Double-Click” on the test line Additional remarks column. A pop-up window will display to add remarks and click “OK”

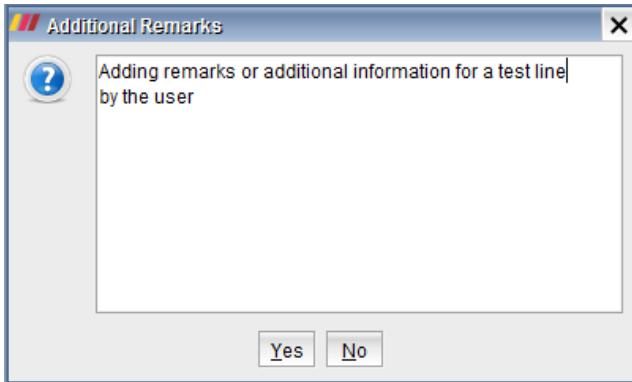


Figure 48: Additional Remarks in Test-Case Editor

Created with the Personal Edition of HelpNDoc: Easy EBook and documentation generator

11.4 Comments

This option is provided to add “User Comments” to the Test-Step(s) of a selected Test-Case. Comments option will only be enabled when a particular Test-Step of a Test-Case is selected. The below figure shows the Comments section in MITE Editor:

Figure 49: Comments in Test-Case Editor

Features of Comments:

- Comments can be added to any Test-Step or Test-Line available inside a Test-Case
- Add comments is only possible to the Test-Step(s) which has data/information
- Comments can be add to any number of Test-Steps present in the Test-Case with data
- There is no restriction on Comments length or number of words
- Cannot add Comments to an empty Test-Step(s)

- f. Comments can be added for the selected 'Test Case' as well
- g.

Created with the Personal Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

11.5 Traceability

Traceability allows the user to add or tag the requirement(s) to the developed Test-Case(s).

- I. This functionality provides the scope to identify the Test-Case(s) with its respective requirement document(s) and requirement-ID(s).
- II. This functionality provides the traceability between the Test-Case(s) and the project requirement(s)

Traceability feature will only be enabled when a Test-Case is selected. This window is available at the bottom of the MITE screen under Tab-Traceability.

There are two ways in which a Requirement ID can be tagged or added to the Test Case from Traceability window:

1. Using “Load Documents” option in Requirement ID’s tab

2. Using “Add Requirements” directly in Relationships tab

Using “Load Documents”:

The below figure shows the MITE Traceability window:

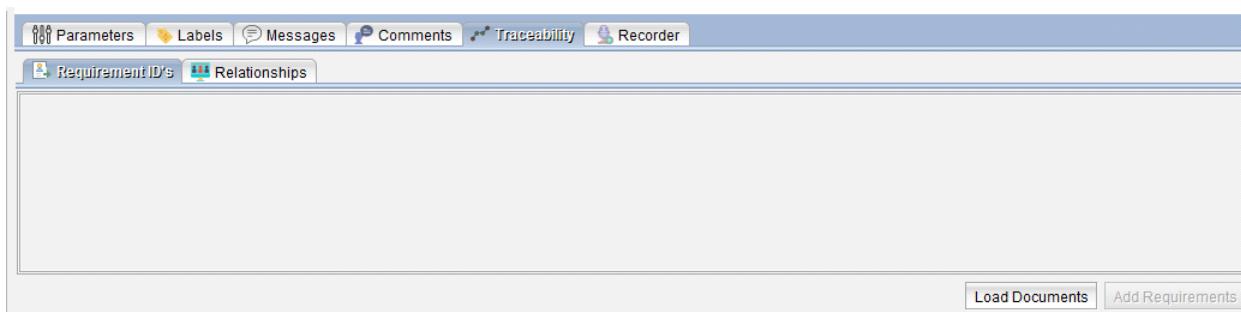


Figure 50: MITE Traceability window

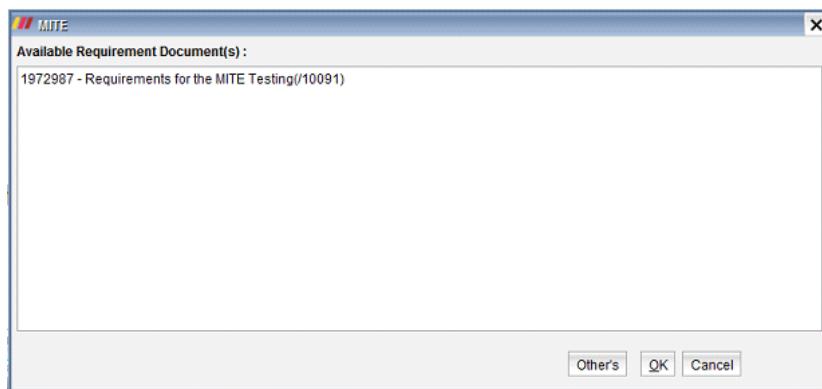


Figure 51: Load same project Documents window

On “others” button click: Lists all the available project, Select required project and Click OK

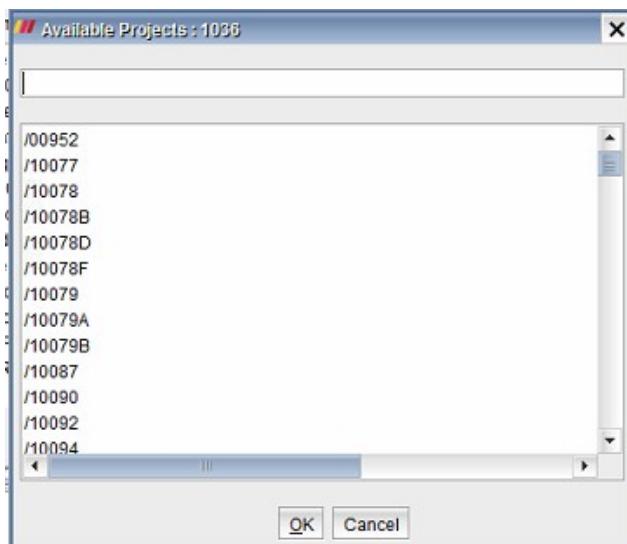


Figure 52: Load other projects

New Requirement documents will be listed as shown below along with the existing Requirement documents

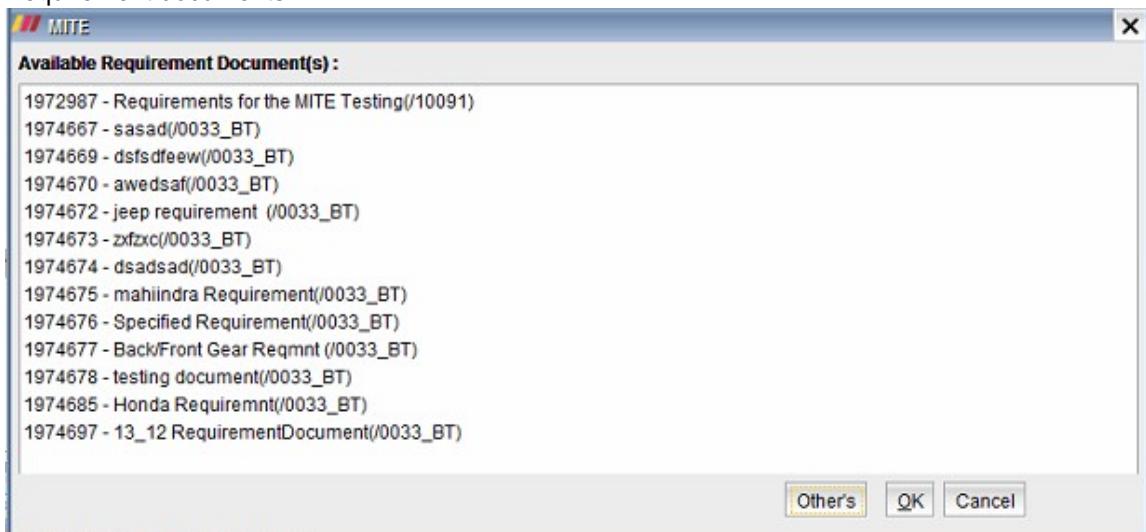


Figure 53: Load Documents of selected Project

Select the requirements document which will appear as a Tree enabling the user to select the requireID using Check-Box

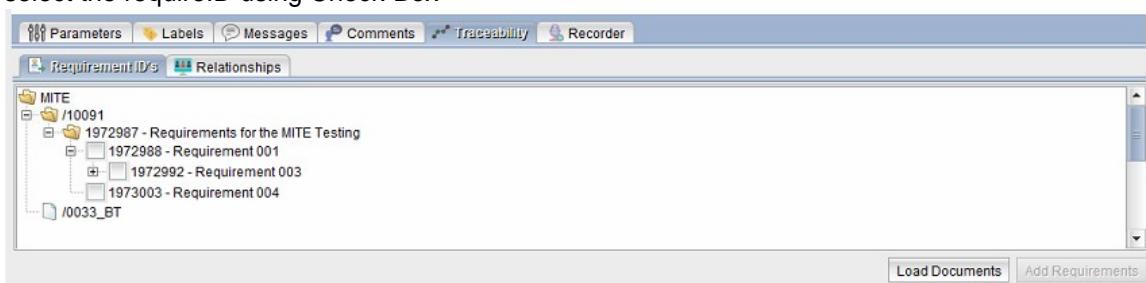


Figure 54: Load Requirement IDs

#	Type	Summary
1	Requirements	1972988 - Requirement 001
2	Requirements	1972992 - Requirement 003
3	Requirements	1972990 - Requirement 002
4	Requirements	1973003 - Requirement 004

Figure 55: Requirement(s) added into Relationships tab

Using “Add Requirements” :

1. Traverse to “Relationships” tab of MITE Traceability window
2. Right click and select “Add Requirements”

#	Type	Summary
1	Requirements	1974700 - doc2
2	Requirements	1974702 - doc3
3	Requirements	1974698 - doc1

Figure 56: Directly adding Requirement(s) into Relationships tab

Then click OK to append to the already existing list of requirement IDs



Figure 57: Requirement IDs

3. After adding the new Requirement IDs

#	Type	Summary
1	Requirements	1974700 - doc2
2	Requirements	1974702 - doc3
3	Requirements	1974698 - doc1
4	Requirements	1974704 - doc4
5	Requirements	1974706 - doc5

Figure 58: Requirement(s) appended into Relationships tab

Delete the Added Requirement:

1. Traverse to “Relationships” tab of MITE Traceability window
2. Select requirement ID
3. Right click and select “Delete Requirements”

11.6 Recorders

This functionality is used to enable recording measurements of a particular (CAN/LIN) signal which requires testing with the recorded data while Test case execution starts.

Recorders will only be available for Test-Case(s) not for Test-Suite(s) Steps to use Recorders:

1. Select the Test-Case
2. Click on the “Recorders” tab shown at the bottom of the Test-Case window as shown in below figure:
3. Click on “Add Recorder” at the bottom right corner , this will enable the user to enter ‘Recorder Name’ , ‘Signal Name’ , ‘Sample Rate’ and ‘Enable/Disable’ as shown in below figure:

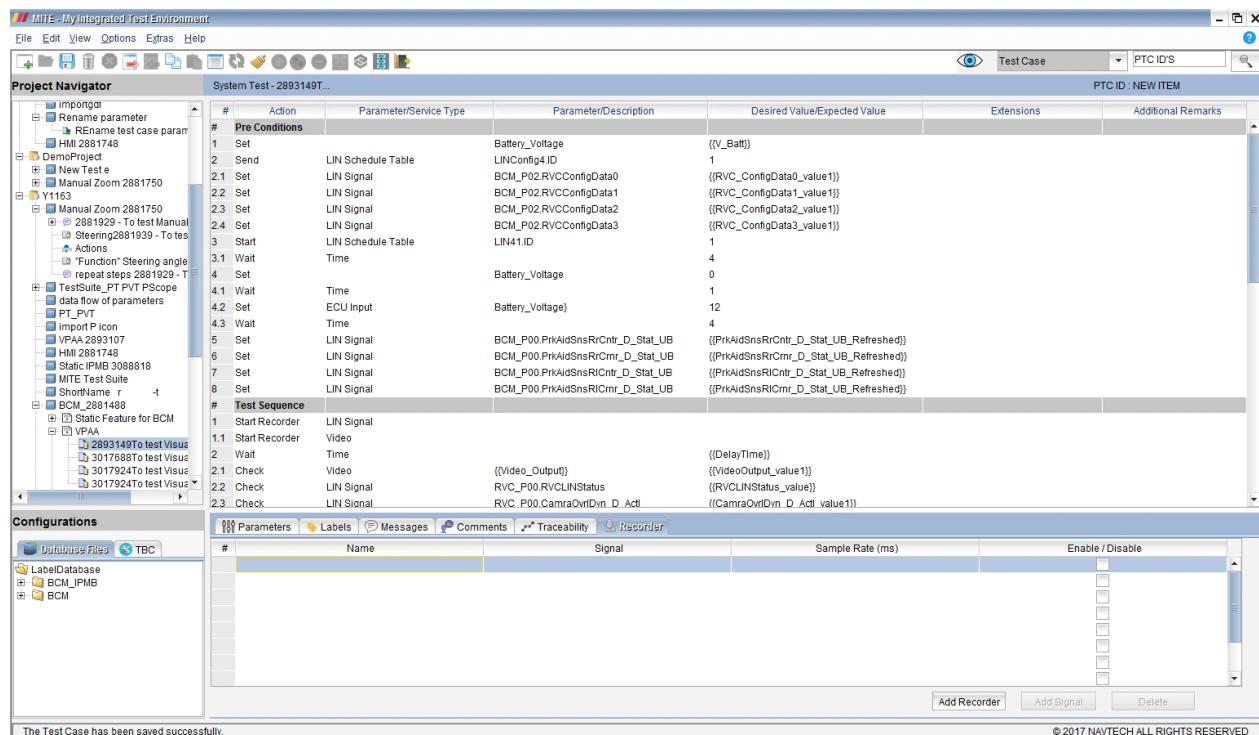


Figure 59: Add Recorder(s) in Test-Case Editor

4. Click “Save or Ctrl+S” to save the recorder(s) contents
5. Option ‘Add Signal’ allows the user to add only signal to the defined recorder , as shown in figure:

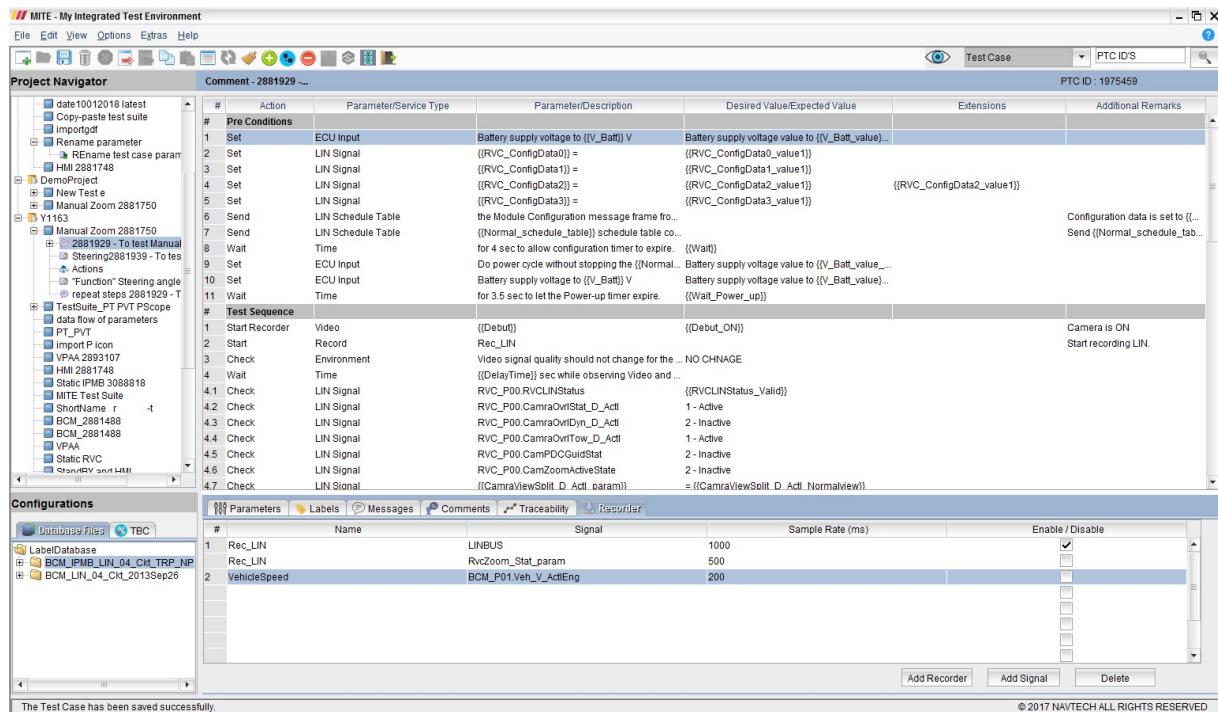


Figure 60: Add signal to the Recorder

Created with the Personal Edition of HelpNDoc: [Free Kindle producer](#)

11.7 Extensions

This is a “Secondary Actions” set by the user to include scenarios to complete connecting conditions. User can add conditions or combination of conditions as shown in below figure(s) :

1. **Condition – AND :- To concatenate two signals of same service type**
2. Range of steps :- List of test steps with serial numbers along with section(s) will be displayed
3. Range of values :- Allows float/integer values with units concatenation(shall be used for mentioning range of values, increment\decrement operations and to mention the previous and current values with combination of stepsize).
4. Tolerance :- To specify the tolerance that shall be allowed for the desired\expected values. Allows float/integer values for percentage(%) and Offset.
5. Step size:-Step size to increment\decrement the desired\expected value from value mentioned in From field to the value mentioned in To field.
6. Reference step:-Refers to particular Desired/Expected Value observed in the step mentioned in this field.
7. Count: - To mention the number iterations the corresponding step needs to be executed. Allows only integer values.
8. Time :
 - a. Response :- It can be used to measure output response time as “with in” time.
 - b. Duration :- It can be used to measure output for some time duration appending a key word “for”.

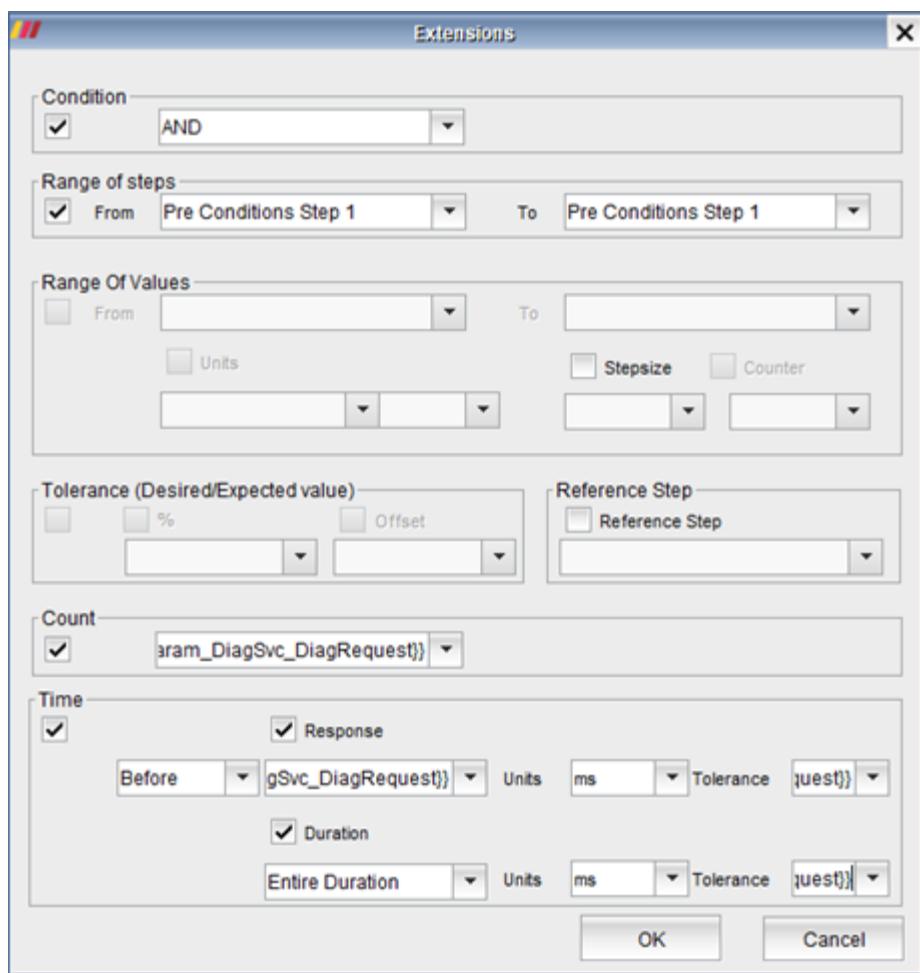


Figure 61: Extensions operation

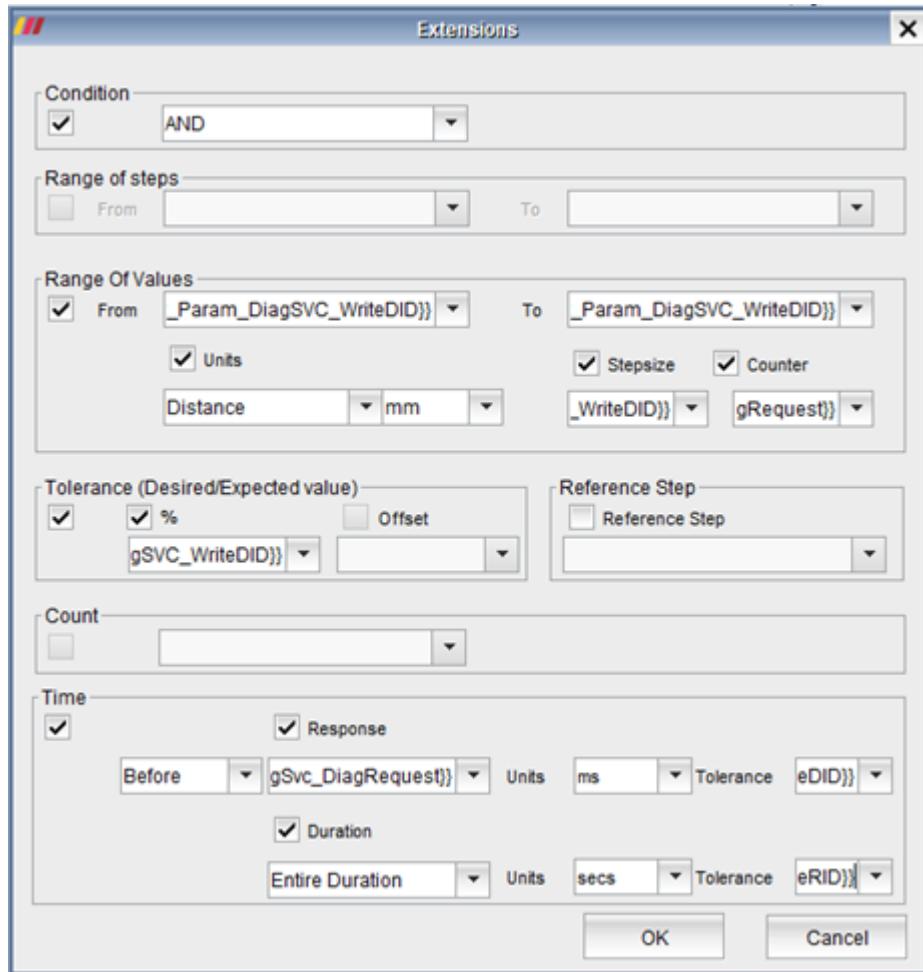


Figure 62: Extensions operation

2. Condition – :Then

To check in particular step and verdict the immediate next step signals and service type.

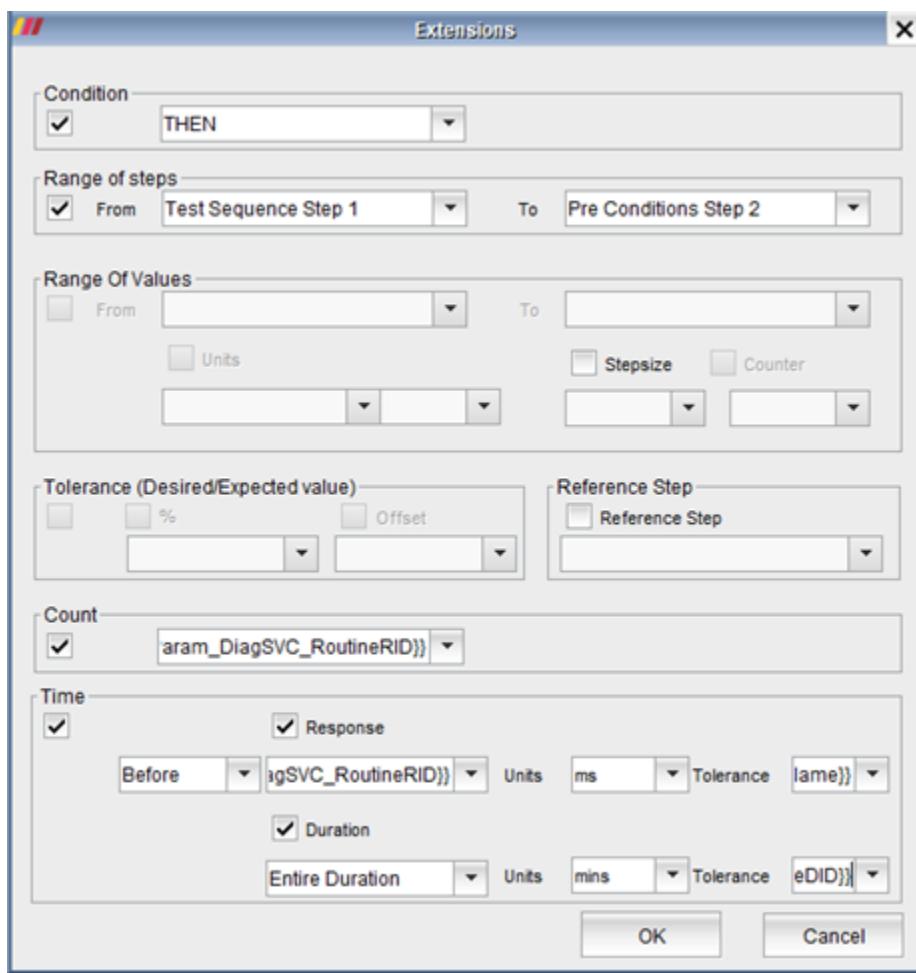


Figure 63: Extensions operation

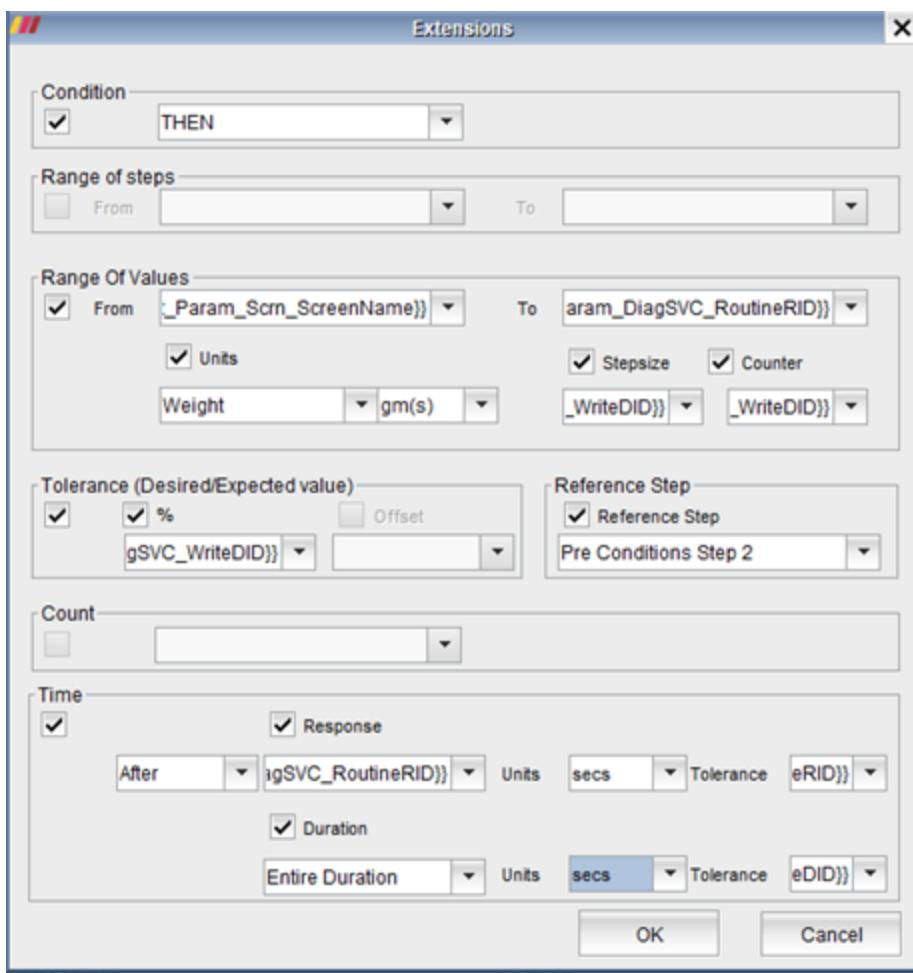


Figure 64: Extensions operation

Created with the Personal Edition of HelpNDoc: [Free CHM Help documentation generator](#)

11.8 After Extensions in MITE

"AFTER" Extensions in MITE

Description:

The extensions **AFTER & BEFORE** are used in MITE test steps to check the value of the signal or variable after/before a specified duration from the recorded results. These extensions are only applicable inside **WHILE Loop**.

The latest modification in the MITE library for these extensions are explained below.

1. If the value of a particular signal or variable is expected to be "X" **after** a specified duration but the value "X" is observed before the specified duration then the verdict will be given as "**Failed**"

Example:

1. Value of a signal or variable is expected to be 4 after 800 ms.
 - a. If the value is observed as 4 after 800ms then the verdict is given as **PASSED** for the test step.
 - b. If the value is observed as 4 before 800 ms then the verdict is given as **FAILED** for the test step.

Test case Structure

1. TC 01

#	Pre C...			
1	Run Test Case	6697201		
#	Test ...			
1	Set Ethernet Signal VelocityAndVehicleConditionProvision:velocityVehicleLongitudinal:velocityVehicleLongitu... >20 Kph			
2	Set Ethernet Signal SteeringAngleProvision:steeringAngleCondition.steeringAngleCondition	1		
3	Wait Time	0.03 sec		
4	Send DiagRequestPhy... \$1906-ReadDTCInformation 06 reportDTCExtendedDataRecordByDTCNumber	0x05 87 78 01		
5	Check DiagResponse:P... Positive resp	0x59 06 05 87 78 2...		
6	Check Ethernet Signal HeadwayObservationPPP:.headwayObservationPPP.qualifierFunctionPPP	6.0 - Funktion_mel...		
7	Wait Time	0.7 sec		
8	While Test Step (s)			From Test Sequence Step 1 To Test Sequence Step 7
8.1	Check Ethernet Signal SteeringAngleProvision:steeringAngleCondition steeringAngleCondition	1 THEN		
8.2	Check Ethernet Signal DisplayCheckControl:DisplayCheckControlMessageStandard.numberCheckControlMes...	2272 After 800ms		
8.3	Check Ethernet Signal DisplayCheckControl:DisplayCheckControlMessageStandard.transmittingFormCheckCo...	5		
8.4	Check Ethernet Signal DisplayCheckControl:DisplayCheckControlMessageStandard.statusCheckControlMes...	1		
8.5	Check Ethernet Signal DisplayCheckControl:DisplayCheckControlMessageStandard.statusIndicateDirectionF...	0		
9	Set Ethernet Signal SteeringAngleProvision:steeringAngleCondition.steeringAngleCondition	0		
10	Check Ethernet Signal DisplayCheckControl:DisplayCheckControlMessageStandard.statusCheckControlMes...	0		
#	Post ...			
1	Run Test Case	6697203		

2. As per new modification in library if value of signal at test step 8.2

- a. is observed as 2272 after 800 ms then verdict is given as "PASS"
- b. is observed as 2272 before 800 ms then verdict is given as "FAILED"

Report

Scenario 1: Test Step is PASSED.

53.2029998302 sec: Check Ethernet Signal DisplayCheckControl:DisplayCheckControlMessageStandard.numberCheckControlMessageStandard is 2272 After 800ms			
✓ Passed PostProcessingVerdict			
Expected value/conditions were satisfied for DisplayCheckControl:DisplayCheckControlMessageStandard.numberCheckControlMessageStandard and actual value is 2272			
Record Data Details of 'ENV_CCM_numberCCM'			
Value	StartTime	End Time	Duration
65535	0.0	1.223	1.223
2272	1.223	1.345	>=0.122

As value is observed as 2272 after 800 ms test step is PASSED.

Scenario 2: Test step is FAILED.

169.1.1.5 Test Step - 4 While Loop
71.6210000515 sec: Check Ethernet Signal DisplayCheckControl:DisplayCheckControlMessageStandard.numberCheckControlMessageStandard is 2272 After 800ms
✗ Failed PostProcessingVerdict
Expected value/conditions weren't satisfied for DisplayCheckControl:DisplayCheckControlMessageStandard.numberCheckControlMessageStandard

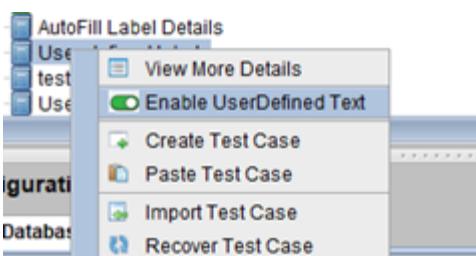
As value 2272 is observed before 800 ms test step is FAILED.

Created with the Personal Edition of HelpNDoc: [Single source CHM, PDF, DOC and HTML Help creation](#)

11.9 User Defined Labels.

User Defined Labels

In mite testcase editor, user can write his own text in Parameter/Description and Desired Value/Expected value. User write some text in testcase1 and again user wants to write same content in the Testcase2, that time we are Allowing user to Generate user defined index list with all the text used in test cases and add them to drop down list to choose them quickly from existing text while authoring test cases.



If user enables the above option for a Testsuite, MITE starts saving the user defined data that is given in the Parameter/Description and Desired Value/Expected columns. So, if user wants to use the same user defined

text later, MITE will show the saved data in the combobox. This makes user to avoid the redundant data in the Test Case editor and the Label Mapping frame.

#	Pre Conditions	
1	Set	ECU Input
2	Send	CAN Message
3	Set	CAN Signal
4	Set	CAN Signal
5	Wait	Time
#	Test Sequence	
1	Send	DiagRequestPhysical
2	Check	DiagResponsePhysical

Created with the Personal Edition of HelpNDoc: [Produce Kindle eBooks easily](#)

11.10 Extensions Pre Processing

Documentation of Extensions Pre Processing

Pre Processing of Extensions.

Case 1:

A variable is referred in Desired Expected value without reference step number in the extension.

If there is no reference step in the extension and gave the desired value in desired/expected value column then prepare the “measure step” and Parameter/description will be desired value of original step(Note: without Parameter/Service type).

1	Send	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1:drivingDirection.drivingDirectionVehicleConfirmed ConfigurationDriverAssistanceFunc...	
2	Check	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1:drivingDirection.drivingDirectionVehicleConfirmed controlCrossTrafficBrake	
#	Post Conditions			
1	Measure		ConfigurationDriverAssistanceFunctionsLegacy::controlActiveSafety.controlActiveSafety	
1.1	Send	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1:drivingDirection.drivingDirectionVehicleConfirmed ConfigurationDriverAssistanceFunc... reference step is Post Conditi...	
2	Measure		controlCrossTrafficBrake	
2.1	Check	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1:drivingDirection.drivingDirectionVehicleConfirmed controlCrossTrafficBrake	reference step is Post Conditi...
3				

Case 2:

A variable is referred in desired Expected value with reference step number in the extension and variable is mentioned in Parameter/Description column in the referred step.

If variable is referred in Parameter/Description, Desired Expected Value and extensions is referred with reference step then it will be considered as a complete step so we no need to write any extra step.

#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	Additional Remarks
	Pre Conditions					
Run	Test Case	9898172				
	Test Sequence					
	Send	Ethernet Signal	ConfigurationDriverAssistanceFunctionsLegacy::controlActiveSafety,controlActiveSafety	2		
	Send	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	ConfigurationDriverAssistanceFunctionsLegacy::controlCrossTrafficBrake	reference step is Test Sequence Step 1	
	Check	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	controlCrossTrafficBrake	reference step is Test Sequence Step 1	
	Post Conditions					
1	Send	Ethernet Signal	ConfigurationDriverAssistanceFunctionsLegacy::controlActiveSafety,controlActiveSafety	2		
2	Measure		controlCrossTrafficBrake			
	Send	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	ConfigurationDriverAssistanceFunctionsLegacy::controlCrossTrafficBrake	reference step is Post Conditions Step 1	
	Check	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	controlCrossTrafficBrake	reference step is Post Conditions Step 1	
0						
1	Check	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	controlCrossTrafficBrake234	reference step is Post Conditions Step 1	

Case 3:

A variable is referred in desired Expected value with reference step number and variable is not mentioned in Parameter\Description column in the referred step.

#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	Additional Remarks
	Pre Conditions					
Run	Test Case	9898172				
	Test Sequence					
	Send	Ethernet Signal	ConfigurationDriverAssistanceFunctionsLegacy::controlActiveSafety,controlActiveSafety	2		
	Send	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	ConfigurationDriverAssistanceFunctionsLegacy::controlCrossTrafficBrake	reference step is Test Sequence Step 1	
	Check	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	controlCrossTrafficBrake	reference step is Test Sequence Step 1	
	Post Conditions					
1	Send	Ethernet Signal	ConfigurationDriverAssistanceFunctionsLegacy::controlActiveSafety,controlActiveSafety	2		
2	Measure		controlCrossTrafficBrake			
	Send	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	ConfigurationDriverAssistanceFunctionsLegacy::controlCrossTrafficBrake	reference step is Post Conditions Step 1	
	Check	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	controlCrossTrafficBrake	reference step is Post Conditions Step 1	
0						
1	Check	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	controlCrossTrafficBrake234	reference step is Post Conditions Step 1	

If Parameter/Description and Desired Expected value is referred with variable and Extensions is referred with reference step then in the referenced step if the Parameter/Description is empty then it should be referred with current Desired Expected value (original step).(Even if the referenced step is empty)

If referenced step's Parameter/Description is not same as desired value of original step then we need to create a "Measure" step with Parameter/Description value as Desired value.

Case 4:

Desired Expected value is empty with reference step number and variable is present in Parameter\Description column in the referred step.

	Pre Conditions					
Run	Test Case	9898172				
	Test Sequence					
	Send	Ethernet Signal	ConfigurationDriverAssistanceFunctionsLegacy::controlActiveSafety,controlActiveSafety	2		
	Check	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection		reference step is Test Sequence Step 1	
	Post Conditions					
	Send	Ethernet Signal	ConfigurationDriverAssistanceFunctionsLegacy::controlActiveSafety,controlActiveSafety	2		
	Send	Ethernet Signal	VEHICLEDYNAMICS::VelocityAndVehicleConditionProvision::1::1::drivingDirection,drivingDirection	ConfigurationDriverAssistanceFunctionsLegacy::controlCrossTrafficBrake	reference step is Post Conditions Step 1	

When Desired Expected value is empty but Parameter/Description is referred with variable and reference step is referred in extensions then the referenced step's Parameter/Description should be referred in

current Desired Expected value.

There no action dependency in this process i.e., Action Type can be anything. This extension process should happen for every Action type.

The order of Pre-processing will be

1. Parameter replacing with current values.
2. While.
3. Extension Pre Processing.
4. Repeat.

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

12 Copy/Paste Operations

Copy and Paste contents/data can be performed using this operation.

- Copy operation will copy the contents and Paste operation will paste the recently copied contents anywhere within the MITE editor only.
- Paste operation will only be enabled when copy operation has performed.
- This is operation is provided with single Test-Step copy and also multiple Test-Steps copy. The recently copied Test-Step or Test-Steps will be pasted accordingly only upon selection the Paste operation

The copy/Paste operations can be performed using Right-click option, as explained below:

Item	Operation performed	Brief Explanation
Copy Test Suite	Right click on Test suite and select Copy Test Suite or Ctrl+C	
Paste Test Suite	Right click on Project and select Paste Test Suite or Ctrl+V	The copied TestSuite(s) will be pasted in selected (same/another) Project.
Copy Test Case	Right click on Test Case and select Copy Test Case or Ctrl+C	
Paste Test Case	Right click on Test suite and select Paste Test Case or Ctrl+V	The copied TestCase(s) will be pasted in selected (same/another) Test suite.

Table 14: Copy-Paste operations

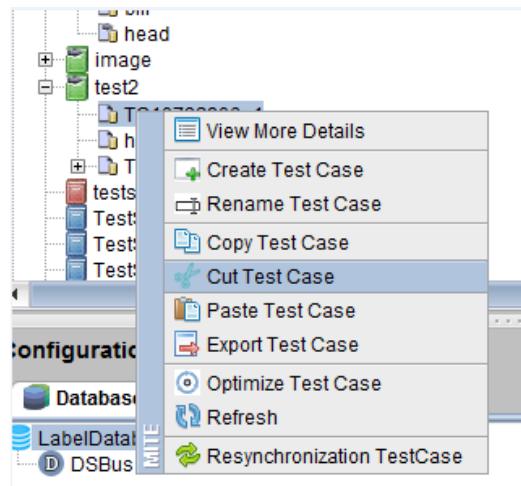
Created with the Personal Edition of HelpNDoc: [Easily create PDF Help documents](#)

12.1 Cut and Paste operation

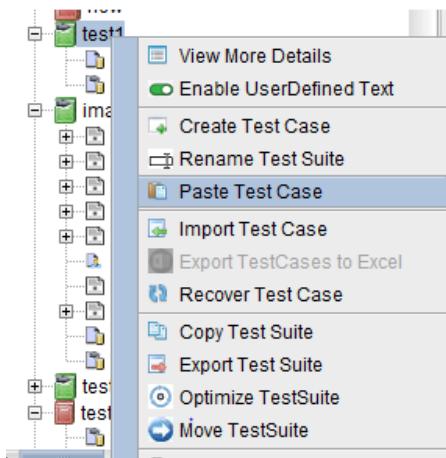
Cut and Paste Functionality

- When user wants to move the selected case to another location, Cut removes the item from its current location and places it into the desired location. Paste inserts the current selected test case contents into the desired location.
- However, they are referred as "cut and paste," which is relocation or "move" and not a "copy."

Cut Test Case: Select a test case of a suite and do right click on it and select the cut test case option, the selected test case will get cut

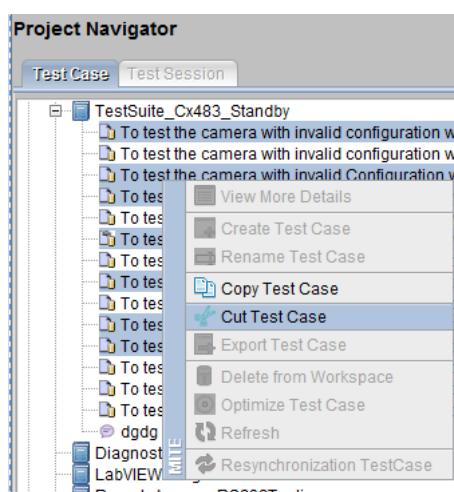


Paste Test Case: Select a Test Suite/Test Case and do right click on it and select the paste test case option, the test case which is cut previously would pasted

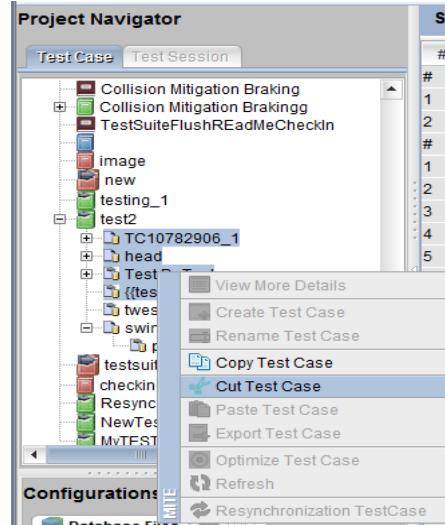


Keyboard shortcuts:

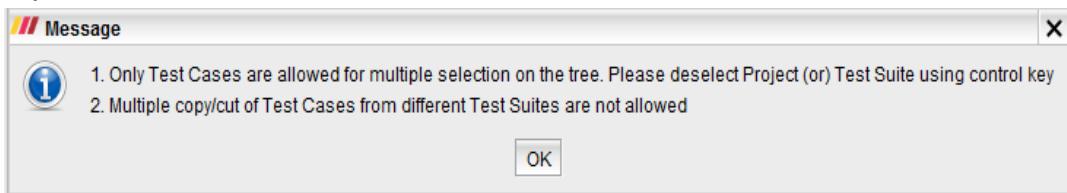
1. Ctrl + X= Used to cut the test case
2. Ctrl + V= Used to paste the test case
3. Multiple/random test cases can be selected by holding Ctrl key



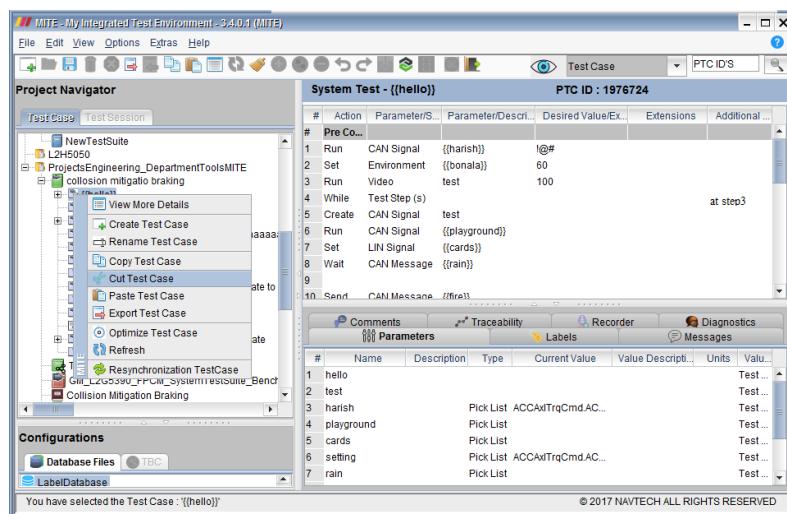
4. Multiple sequential test cases can be selected by holding Shift key

**Note:**

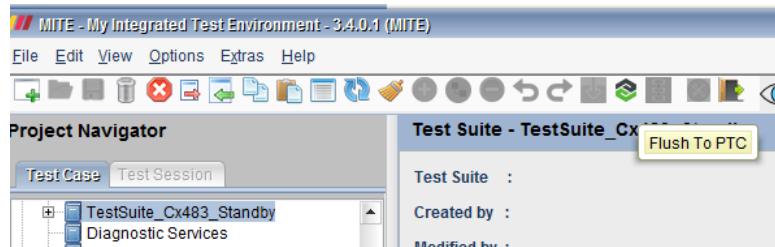
1. Multiple cut of test cases from different test suites are not allowed.



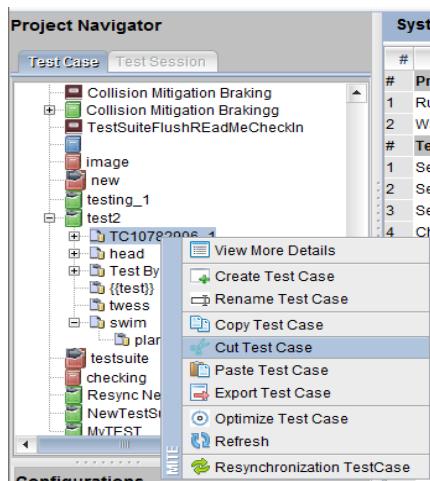
2. Cut test case option is only available for test cases and in the test case role itself.



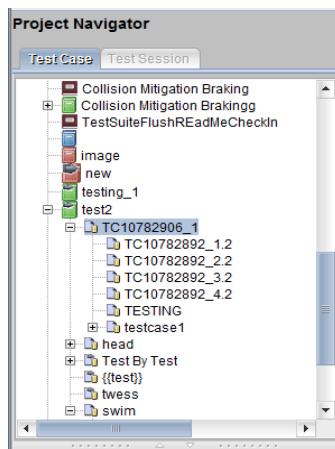
3. After a cut and paste action, the user must flush both (Cut- test case from where it was cut and Paste- test case/test suite to where it was pasted) test cases/test suite to the PTC to reflecting the changes in PTC



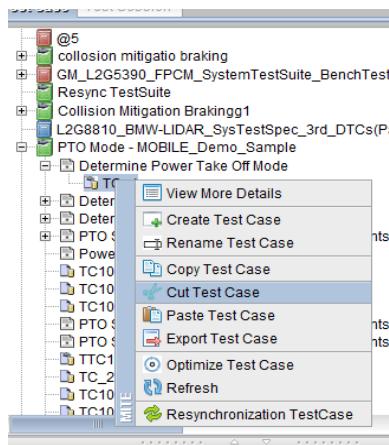
4. If there are child test cases in a particular test case, to get all of the test cases in the other test suite/test case, without any tree expansion, the user must choose the parent node test case and can do a cut-and-paste operation.



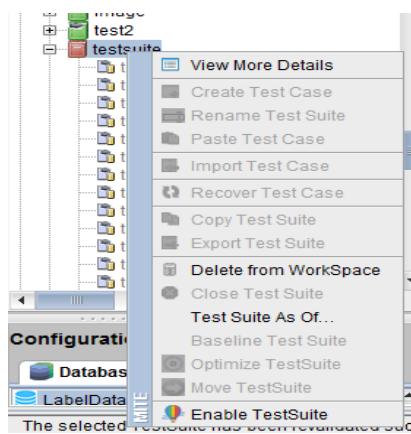
5. If user expands the parent node test case then if he perform cut and paste operation then user will get only that particular selected parent test case, he will not get child test cases



6. If the test case is cut-pasted from one test suite to another, the PTC id is likewise copied to that test suite



- If the user cut the test case from normal test suite and if he tries to paste it in baseline test suite level then it won't be done.



Created with the Personal Edition of HelpNDoc: [Easily create EPub books](#)

13 Delete/Recovery Operations

Delete: This operation is provided to delete or remove the data/ contents which are not required for the project.

Recover: This operation is provided to recover the data/contents which has been deleted from the project. The Recovery operation can only work until clean-up is not performed on the MITE-Project.

The Delete/Recovery operations can be performed using Right-click option, as explained below:

Item	Operation performed	Brief Explanation
Delete Test Suite	Right click on Test suite and select Delete Test Suite	The Selected Test Suite can be Deleted
Recover Test Suite	Right click on Project and select Recover Test Suite	The selected Testsuite(s) will Recover in same/another Project.
Delete Test Case	Right click on Test Case and select Delete Test Case	

Recover Test Case	Right click on Test suite and select Recover Test Case	The selected Testcase(s) will Recover in same/another Testsuite(s).
-------------------	--	---

Table 15: Delete-

Recovery operations NOTE:- The TestSuite(s)/TestCase(s) with PTC IDs cannot be deleted

Created with the Personal Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)

14 Import/Export Operations

Import and Export options provide the facility to save the developed MITE-format Test-Suite(s) or Test-Case(s) data into a “ZIP file” format.

This operation can be carried out between different users in sharing data/contents and also for re-usability of the data/contents.

Import: Allows the Test-Suite(s) and Test-Case(s) to display in MITE-editor

Export: Allows the Test-Suite(s) and Test-Case(s) to save in Zip file from MITE

The Import/Export operations can be performed using Right-click option, as explained below:

Item	Operation performed	Brief Explanation
Import Test Suite	Right click on Project and select Import Test Suite	The selected Testsuite(s) will imported from Zip file
Export Test Suite	Right click on Test Suite and select Export Test Suite	The selected Testsuite(s) will exported into Zip file
Import Test Case	Right click on Test Suite and select Import Test Case	The selected Testcase(s) will imported from Zip file
Export Test Case	Right click on Test Case and select Export Test Case	The selected Testcase(s) will exported into Zip file

Table 16: Import-Export operations

Created with the Personal Edition of HelpNDoc: [Easily create Help documents](#)

15 Shortcut Keys & Mouse operations

The below listed operations and their combinations are possible on MITE Test-Case Editor :

S.no	Shortcut Key-board	Function Description	Selection item
1.	Ctrl+Enter	Adds a New test line in the test case editor	Select any column in editor and press “Ctrl+Enter”
2.	Ctrl+D	Removes single or multiple selected test line(s) in the test case editor	Select any column in editor and press “Ctrl+D”
3.	Ctrl+A	Select all the test lines present in the editor	Select any column in editor and press “Ctrl+A”
4.	Ctrl+C	Copy the selected test line or lines	Copies the selected lines. The selection of lines is possible by selecting from the numbers on the editor only.
5.	Ctrl+V	Paste the selected test line or lines	Paste the copied lines

6.	Delete-Key	Deletes the contents	Select any row or column In the test case editor
7.	Esc – Key	Retains back the change present before edit operation and before save operation	Edit any cell in test case editor and hit Esc key
8.	Mouse operation	Multiple line selection can be done using mouse	Drag the mouse pointer along the number section(on left side) on the test case editor to select multiple lines

Table 17: Shortcut Key and Mouse operations

Created with the Personal Edition of HelpNDoc: [Full-featured EPub generator](#)

16 Flushing into PTC server

Created with the Personal Edition of HelpNDoc: [Free Kindle producer](#)

16.1 About Flushing

Flushing is a method developed to submit Test-Suite/Test-Cases/Heading/Comments which are in MITE compatible format into PTC Server.

Created with the Personal Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

16.2 Environment for Flushing into PTC server

To carry-on the flushing procedure successfully the below pre-requisites should complete:

1. PTC server should be logged-in and server should be up
2. MITE application and PTC server should be up/open in the same PC/System

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

16.3 Contents which can be flushed into PTC

S.No	MITE contents	Flush to PTC (Yes/No)	Brief explanation
1.	MITE Project	No	-
2.	Project mapped into MITE(from PTC)	No	-
3.	Database Files	No	-
4.	MITE Test Suite alone	Yes	Only MITE compatible format tests can be flushed into PTC server(at a time).
5.	MITE Test Case alone	No	-
6.	MITE Test Suite along with MITE Test Case(s)	Yes	Only MITE compatible format test case(s) which are under MITE Test Suite can be flushed into PTC server

Table 18: MITE contents flushing

Created with the Personal Edition of HelpNDoc: [Full-featured multi-format Help generator](#)

16.4 How to flush the contents into PTC server?

For Flushing the MITE compatible format contents into PTC server can be done only through MITE - "Submit" operation. This operation is indicated using "S" icon on top left corner of the MITE window as shown below:

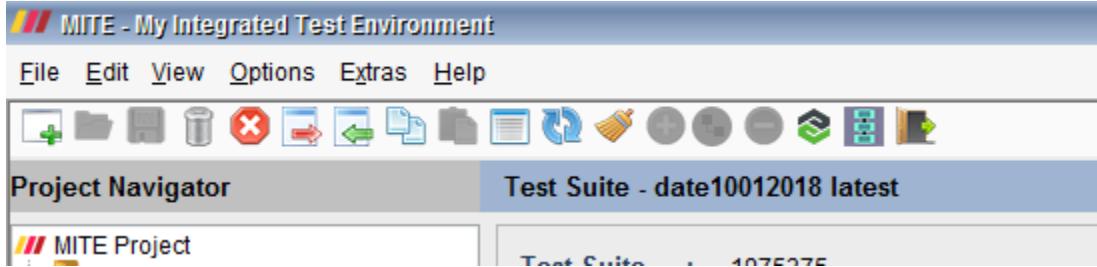


Figure 63: PTC Submit operation

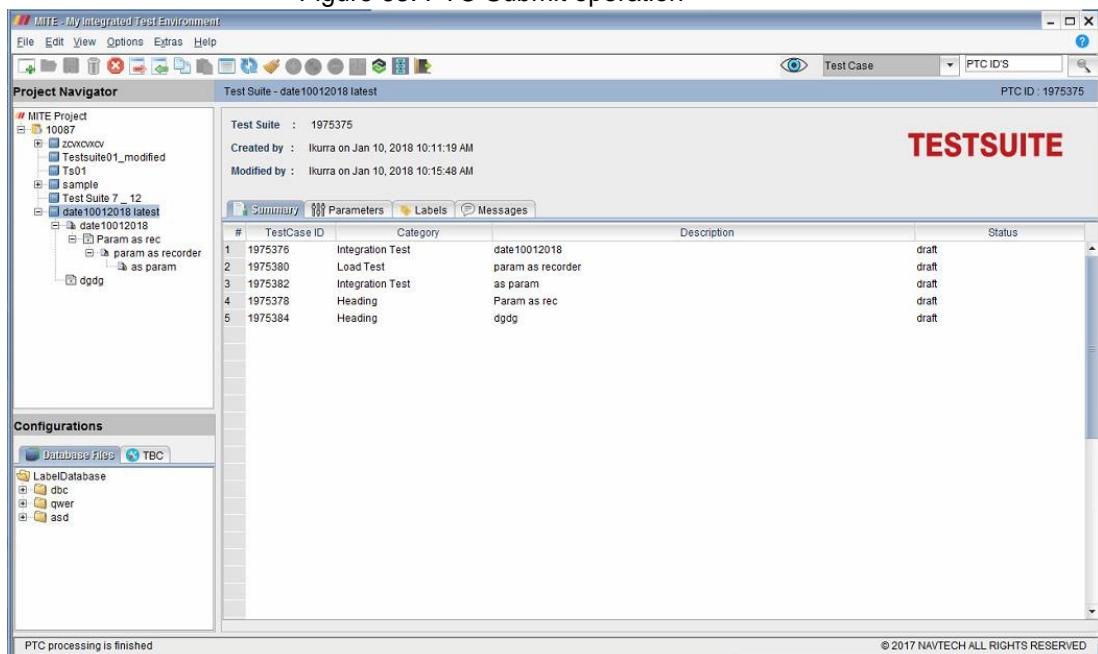


Figure 64: PTC IDs

Note: This Submit option only displays on selecting a Test Suite otherwise Submit icon will be in disable mode

While Flushing Test-Suite, the selected Test-Suite will added into PTC's respective project.

Created with the Personal Edition of HelpNDoc: [Free Qt Help documentation generator](#)

16.5 PTC Test Suite state validation

Description:-

Previously while Flushing/Resync the Test Suite we were not considering the state of the suite(in review/published/Restricted) and user was able to flush the test suite properly .

Now we will flush the test suite based on the state of the suite in PTC.Only if the state is Open we are allowing the Test Suite to be Flushed.

Step 1:-User need to check the state of the suite in PTC as shown below:-

Process

Category: Suite

Project: /Projects/Engineering_Department/Tools/MITE (Out of Phase)

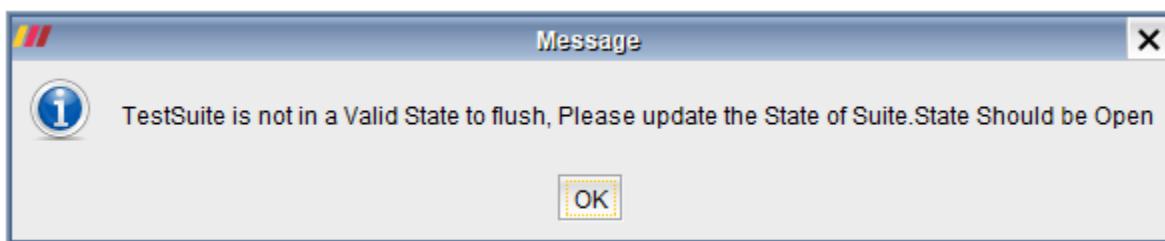
State: Restricted (Constrained)

Options

Step 2:-On clicking the Flush icon as shown below:-



Step 3:- When User will flush the Test suite following pop up will be shown as below:-



Created with the Personal Edition of HelpNDoc: [Easily create CHM Help documents](#)

16.6 MITE Non-Dependency on Sandbox

With MITE-v2.8.0.0 we are introducing a complete independent and a unique solution for PTC related functionalities by removing the sandbox dependency.

Previously while doing any PTC actions with respect to MITE there is a mediator between MITE and PTC which is Sandbox. Now the main Usage with this new implementation is there is no dependency with sandbox. This will basically help users with less time consumption and increased performance for PTC related operations.

So, as a result there is no back ward MITE version compatibility with respect to PTC related functionalities from MITE-v2.8.0.0 i.e., user should never use older MITE versions for PTC related functionalities after start using MITE-v2.8.0.0.

Users need to follow below instructions while using MITE-v2.8.0.0

1) Before start using MITE-v2.8.0.0, user should Resynchronize all the available Test Suites with the current MITE Version that is been used.

2) Before start using MITE-v2.8.0.0, user should Update all the shared/Reuse Test Suites with the current MITE Version that is been used by using the option "Update Shared/Reuse Data from Source Test Suite".

3) After doing all the necessary Updates with the current MITE Version, the very first operation related to PTC from MITE-v2.8.0.0 should be flushing only.

4) For this, a new option is added for every Test Suite in the right click as "Flush Whole Test Suite" which is an added advantage. With this, the whole Test Suite is converted to required format by dropping the sand

box and from then the rest of the PTC operations will go smoothly. This operation is an one time operation for one Test Suite.

5) If multiple users are using same test suite, any one user must ensure that it is completely flushed to PTC with MITE-v2.8.0.0. Also, all the corresponding team members should update their MITE version to MITE-v2.8.0.0

Note:-User must follow above instructions for smoother operations ,lesser redundancy and better usage.

Created with the Personal Edition of HelpNDoc: [Free Kindle producer](#)

16.7 MITE User Scenarios

Scenario 1:-

For flushing brand new Test suite consisting of 1149 test cases it took 23 minutes to flush to PTC successfully as seen in below image:-

```
[23-03-2021-06-34-22]: PROCESSING : Time Taken to create New Test Case: '0' Secs
[23-03-2021-06-34-22]: **** New TestCase Create Operation ****
[23-03-2021-06-34-22]: PROCESSING :Total number of TestCases :----- 1149
[23-03-2021-06-34-22]: PROCESSING :Current running TestCase: -----1148
[23-03-2021-06-34-22]: PROCESSING :Testcase Inforamtion: ----- NEW ITEM - TC10782866_3
[23-03-2021-06-34-23]: PROCESSING : Time Taken to create New Test Case: '1' Secs
[23-03-2021-06-34-23]: **** New TestCase Create Operation ****
[23-03-2021-06-34-23]: PROCESSING :Total number of TestCases :----- 1149
[23-03-2021-06-34-23]: PROCESSING :Current running TestCase: -----1149
[23-03-2021-06-34-23]: PROCESSING :Testcase Inforamtion: ----- NEW ITEM - TC10782866_3.2.1
[23-03-2021-06-34-24]: PROCESSING : Time Taken to create New Test Case: '1' Secs
[23-03-2021-06-34-27]: ****
[23-03-2021-06-34-27]: PROCESSING :Update Information For PTC TestSuite:12154614
[23-03-2021-06-34-27]: PROCESSING :Update Information For PTC TestCases
[23-03-2021-06-36-03]: FINAL: PTC SUBMIT PROCESS IS SUCCESSFULLY COMPLETED
[23-03-2021-06-36-03]: FINAL: TimeTaken for PTC Submit Process Is '23.0 Minutes'
```

Scenario 2:-

For flushing 15 modified test cases it took 5 minutes to flush to PTC.

```
[23-03-2021-06-58-48]: PROCESSING :TestCase Inforamtion: -----12155773 - TC10784861_2.2, TC10784863_2.2
[23-03-2021-06-58-49]: PROCESSING : Time Taken to Update Test Case: '1' Secs
[23-03-2021-06-58-49]: **** TestCase Edit Operation ****
[23-03-2021-06-58-49]: PROCESSING :Total number of TestCases :----- 1149
[23-03-2021-06-58-49]: PROCESSING :Current running TestCase: -----566
[23-03-2021-06-58-49]: PROCESSING :TestCase Inforamtion: -----12155775 - TC10784883_MOBILE
[23-03-2021-06-58-50]: PROCESSING : Time Taken to Update Test Case: '0' Secs
[23-03-2021-06-58-50]: **** TestCase Edit Operation ****
[23-03-2021-06-58-50]: PROCESSING :Total number of TestCases :----- 1149
[23-03-2021-06-58-50]: PROCESSING :Current running TestCase: -----568
[23-03-2021-06-58-50]: PROCESSING :TestCase Inforamtion: -----12155779 - TC10784883_2.2
[23-03-2021-06-58-51]: PROCESSING : Time Taken to Update Test Case: '0' Secs
[23-03-2021-06-58-53]: ****
[23-03-2021-06-58-53]: PROCESSING :Update Information For PTC TestSuite:12154614
[23-03-2021-06-58-53]: PROCESSING :Update Information For PTC TestCases
[23-03-2021-07-00-42]: FINAL: PTC SUBMIT PROCESS IS SUCCESSFULLY COMPLETED
[23-03-2021-07-00-42]: FINAL: TimeTaken for PTC Submit Process Is '5.0 Minutes'
```

Scenario 3:-

For flushing 50 modified test cases it took 5 minutes to flush to PTC.

```
[23-03-2021-07-13-37] : PROCESSING : TestCase Inforamtion: -----12156833 - TC10782830_1.27
[23-03-2021-07-13-38] : PROCESSING : Time Taken to Update Test Case: '0' Secs
[23-03-2021-07-13-38] : ***** TestCase Edit Operation *****
[23-03-2021-07-13-38] : PROCESSING : Total number of TestCases :----- 1149
[23-03-2021-07-13-38] : PROCESSING : Current running TestCase: -----1069
[23-03-2021-07-13-38] : PROCESSING : TestCase Inforamtion: -----12156835 - TC10782830_1.26
[23-03-2021-07-13-39] : PROCESSING : Time Taken to Update Test Case: '0' Secs
[23-03-2021-07-13-39] : ***** TestCase Edit Operation *****
[23-03-2021-07-13-39] : PROCESSING : Total number of TestCases :----- 1149
[23-03-2021-07-13-39] : PROCESSING : Current running TestCase: -----1070
[23-03-2021-07-13-39] : PROCESSING : TestCase Inforamtion: -----12156837 - TC10782830_1.28
[23-03-2021-07-13-40] : PROCESSING : Time Taken to Update Test Case: '0' Secs
[23-03-2021-07-13-40] : ***** TestCase Edit Operation *****
[23-03-2021-07-13-40] : PROCESSING : Total number of TestCases :----- 1149
[23-03-2021-07-13-40] : PROCESSING : Current running TestCase: -----1071
[23-03-2021-07-13-40] : PROCESSING : TestCase Inforamtion: -----12156839 - TC10782830_1.29
[23-03-2021-07-13-40] : PROCESSING : Time Taken to Update Test Case: '0' Secs
[23-03-2021-07-13-42] : *****
[23-03-2021-07-13-42] : PROCESSING :Update Information For PTC TestSuite:12154614
[23-03-2021-07-13-43] : PROCESSING :Update Information For PTC TestCases
[23-03-2021-07-15-21] : FINAL: PTC SUBMIT PROCESS IS SUCCESSFULLY COMPLETED
[23-03-2021-07-15-21] : FINAL: TimeTaken for PTC Submit Process is '5.0 Minutes'
```

Scenario 4:-

For flushing 200 modified test cases it took 6 minutes to flush to PTC.

```
[23-03-2021-07-23-21] : PROCESSING : TestCase Inforamtion: -----12156350 - TC10782830_1.48
[23-03-2021-07-23-22] : PROCESSING : Time Taken to Update Test Case: '1' Secs
[23-03-2021-07-23-22] : ***** TestCase Edit Operation *****
[23-03-2021-07-23-22] : PROCESSING : Total number of TestCases :----- 1149
[23-03-2021-07-23-22] : PROCESSING : Current running TestCase: -----845
[23-03-2021-07-23-22] : PROCESSING : TestCase Inforamtion: -----12156352 - TC10782830_1.49
[23-03-2021-07-23-23] : PROCESSING : Time Taken to Update Test Case: '0' Secs
[23-03-2021-07-23-23] : ***** TestCase Edit Operation *****
[23-03-2021-07-23-23] : PROCESSING : Total number of TestCases :----- 1149
[23-03-2021-07-23-23] : PROCESSING : Current running TestCase: -----846
[23-03-2021-07-23-23] : PROCESSING : TestCase Inforamtion: -----12156354 - TC10782830_1.25
[23-03-2021-07-23-24] : PROCESSING : Time Taken to Update Test Case: '1' Secs
[23-03-2021-07-23-24] : ***** TestCase Edit Operation *****
[23-03-2021-07-23-24] : PROCESSING : Total number of TestCases :----- 1149
[23-03-2021-07-23-24] : PROCESSING : Current running TestCase: -----847
[23-03-2021-07-23-24] : PROCESSING : TestCase Inforamtion: -----12156356 - TC10782830_1.26
[23-03-2021-07-23-25] : PROCESSING : Time Taken to Update Test Case: '0' Secs
[23-03-2021-07-23-27] : *****
[23-03-2021-07-23-27] : PROCESSING :Update Information For PTC TestSuite:12154614
[23-03-2021-07-23-28] : PROCESSING :Update Information For PTC TestCases
[23-03-2021-07-24-59] : FINAL: PTC SUBMIT PROCESS IS SUCCESSFULLY COMPLETED
[23-03-2021-07-24-59] : FINAL: TimeTaken for PTC Submit Process is '6.0 Minutes'
```

Points to be noted:-

- The time taken for flushing is completely depends up on the network band width. The more the bandwidth, the less time it takes for flushing.
- We suggest users not to use mobile network or weaker networks while doing any PTC related functionalities like flushing or resynchronization.
- The decent number of Test Cases that can be flushed into the PTC in one go are 800-1000 Test Cases.
- We also suggest users to flush the Test Suites from time to time to reduce the flushing over all time.

- Please use versions 2.8.1.1 and above for PTC related functionalities. Using lower versions for these operations is strictly prohibited.

Created with the Personal Edition of HelpNDoc: [Easily create EPub books](#)

17 Test Suite Resynchronization

Resync Test Suite provides opportunity to update the current selected test suite with respect to PTC test suite. Test Suite gets update or resync number of test cases under the test suite along with the tree structure as per PTC. Using this option a user can get the latest content updates which are present in PTC for the selected Test suite on a single click in MITE.

Resync will keep the user updated with the changes being done on the selected or currently working test suite by another user or in PTC.

Resync option is present as one of the right-click operations on Test Suite, as shown below:

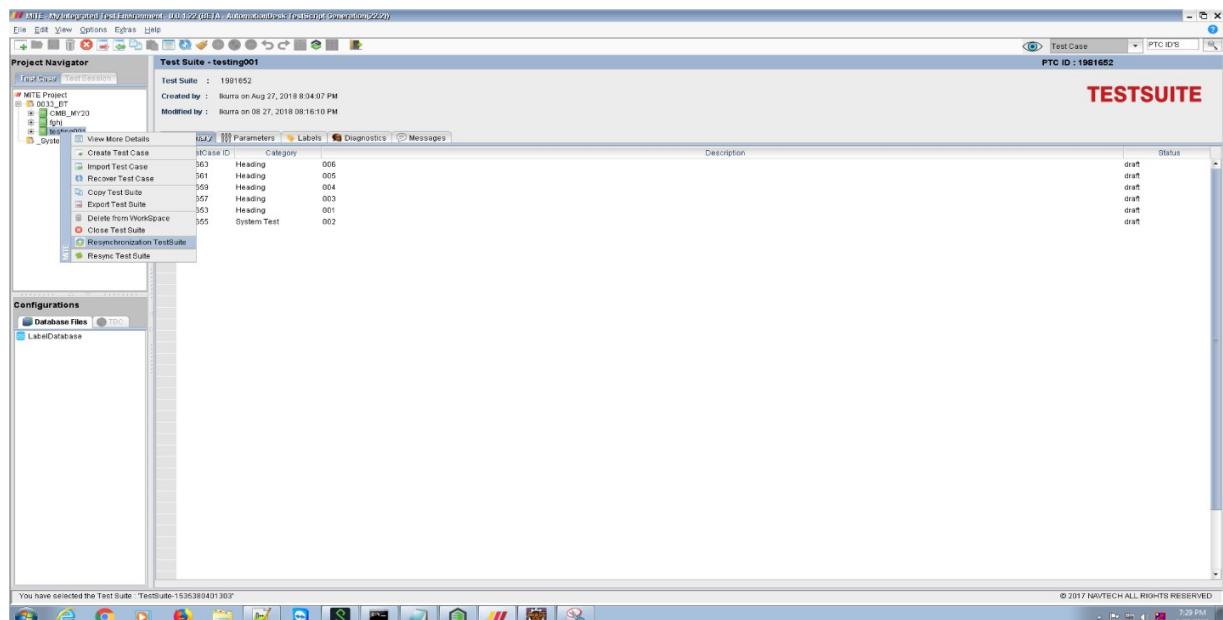


Figure 65: Resynchronization Test Suite Option on Right Click

Steps to follow to perform Resynchronization Test Suite:

1. Make sure PTC connection is up and running
2. Select a Test suite from a project which is already submitted to PTC lately
3. Right Click on selected test suite
4. Resynchronization Test Suite appears
5. Click on Resync option -- Resync operation initiates
6. Resync window as shown below : It has the following information
 - i. Test suite Tree structure as in MITE
 - ii. Number of test cases under a test suite
 - iii. Details about the test case

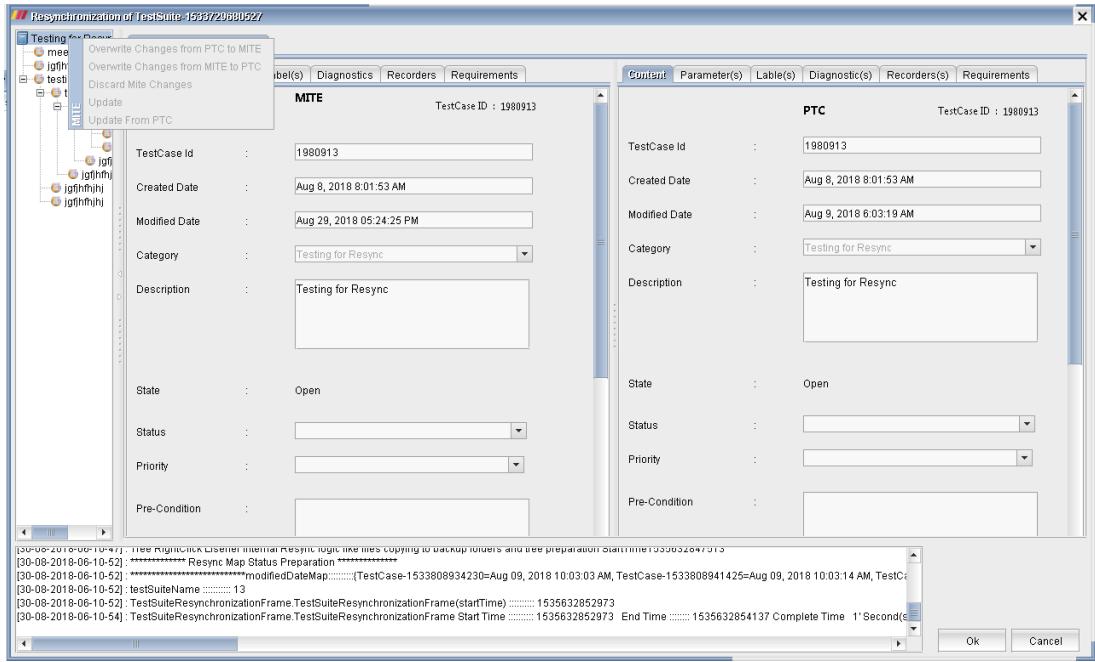


Figure 66: Resynchronization Test Suite Window

7. Select a test case from the tree structure , a comparison window will appear as shown
 - i. It has the details of the data or content present in the selected test case
 - ii. GUI Comparison between PTC and MITE test case
 - iii. All the possible editable content lists

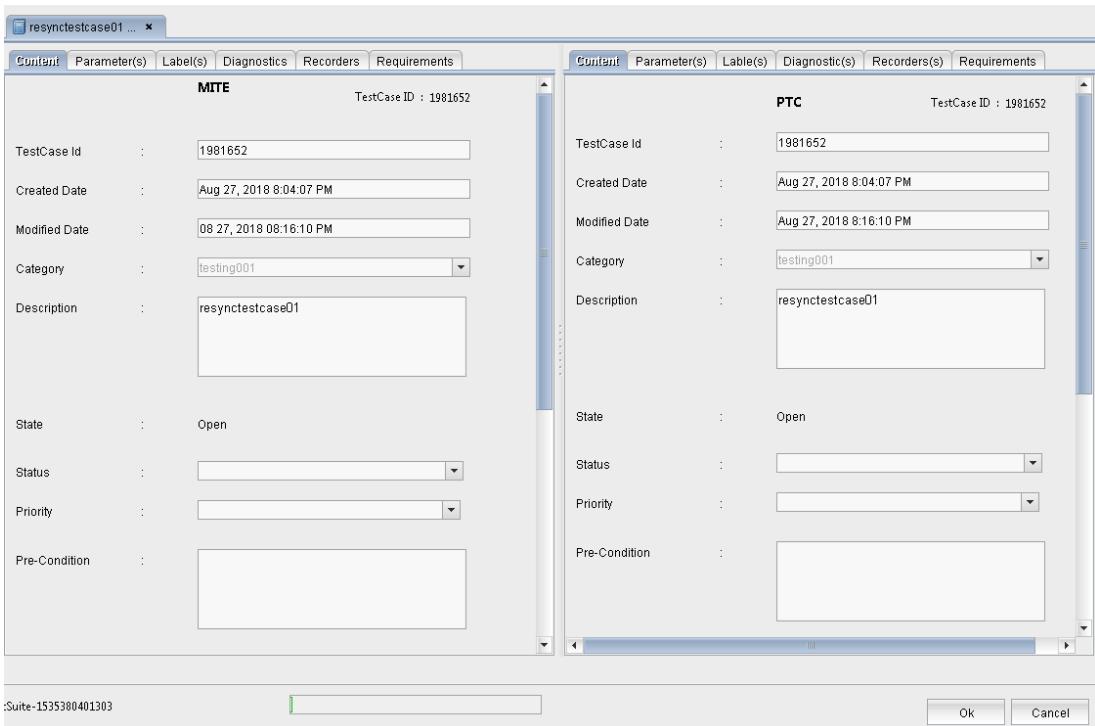


Figure 67: Resync Comparison Window

8. Right click on Test suite node within Comparison window will have different user options as shown in below figure:

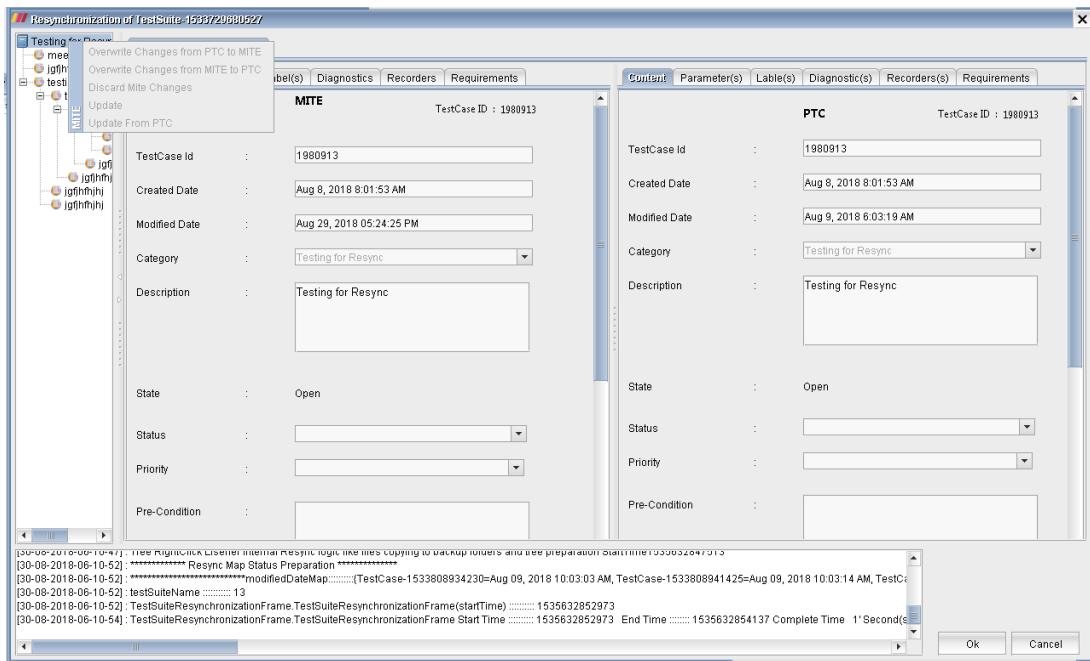


Figure 68: User operations on Resync Comparison Window

- i. Overwrite Changes from PTC to MITE – This will replace the mite content
 - ii. Overwrite Changes from MITE to PTC – This will replace the PTC test case content
 - iii. Discard MITE changes – Recently made changes in mite will be discarded itself
 - iv. Update – Updates the changes from MITE to PTC
 - v. Update From PTC - Updates the changes from PTC to MITE
9. Right click on Test case node within Comparison window will have different user options as shown in below figure:

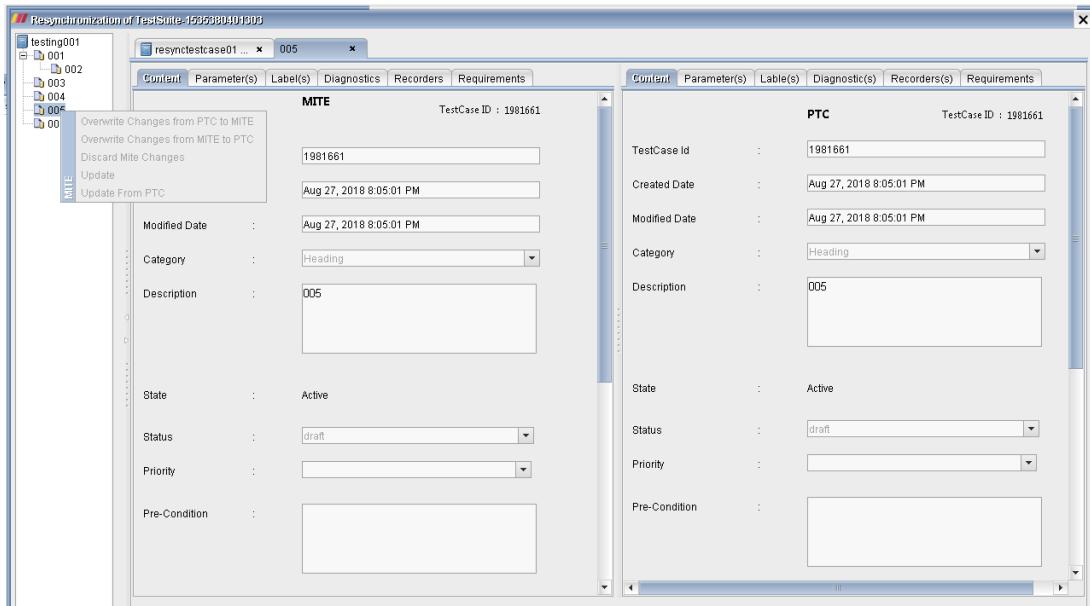


Figure 69: On Test Case User operations on Resync Comparison Window

There will exists two kinds of test cases

- i. *Test Case with changes – These test cases will have the “Update” operations enabled as shown in above figure*
 - ii. *Test Case without changes – These test cases will not have any user operations to perform*
10. Once user operations are performed the comparison window will be closed and changes will be saved too.

11. At this stage user will have the clear picture about the test suite he/she is working on.

Created with the Personal Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

17.1 Parameters Merge in Resync Window

With the current implementation of re synchronization, there is no possibility of merging at Test Suite level. User can get the content either from PTC or from the MITE. There is no possibility of merging one field from PTC and other field from MITE. So, due to this the current value or any field related to parameter will not be updated properly.

With this new implementation, parameter will be updated as follows:

- 1) Overwrite Changes from PTC to MITE ---- The current Values for the parameters which are there in both MITE and PTC will be updated from PTC. Also, if there are any new parameters in the PTC those will be fetched here.
- 2) Overwrite Changes from MITE to PTC ---- The current Values for the parameters which are there in both MITE and PTC will be retained with MITE values. Also, if there are any new parameters in the MITE those will be flushed in the very next flush operation. Also, if there are any new parameters in the PTC they will be fetched in this operation.
- 3) Update From PTC --- The current Values for the parameters which are there in both MITE and PTC will be updated from PTC. Also, if there are any new parameters in the PTC those will be fetched here.

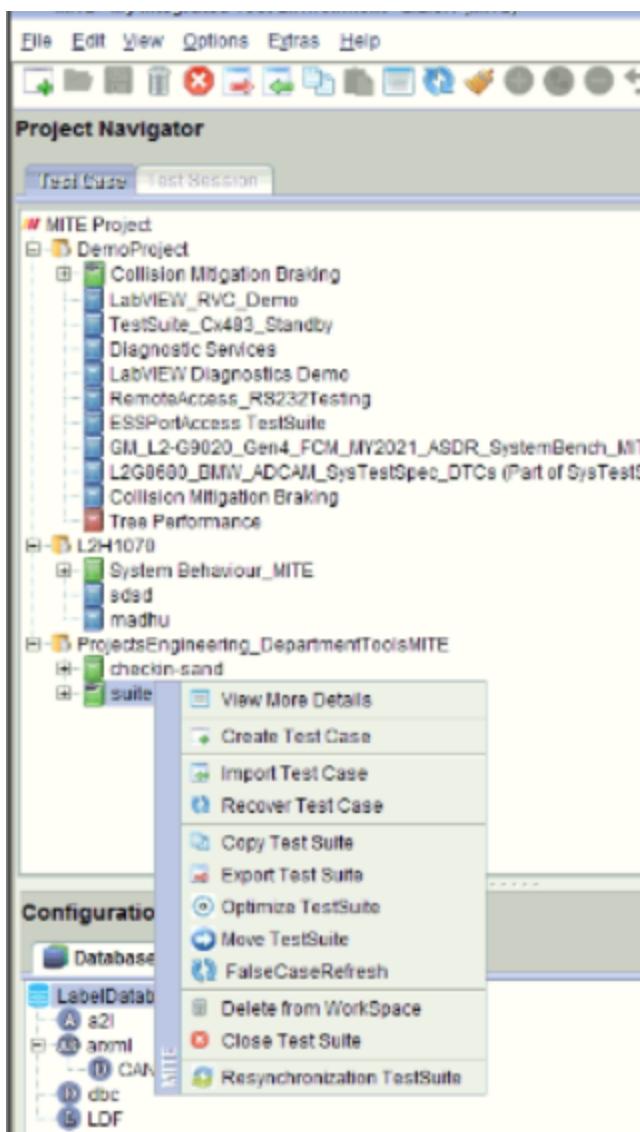
Created with the Personal Edition of HelpNDoc: [Easily create EPub books](#)

17.2 Resynchronization Merge window.

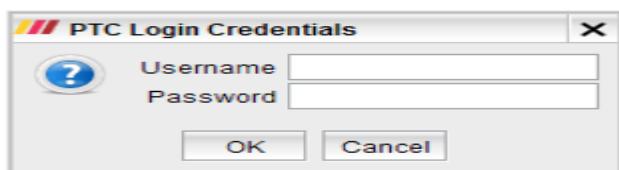
Resynchronization Merge window

Description:- In Resynchronize UI User can compare the MITE Local Test case Content with PTC TestCase content. If User has done any changes in MITE at that time other user has done any modifications in PTC with same test case and first user wants to flush at that time he/she can view red color arrow marks from right to left and that time he/she can use Resync merge window and merge the data from right to left(PTC to MITE).

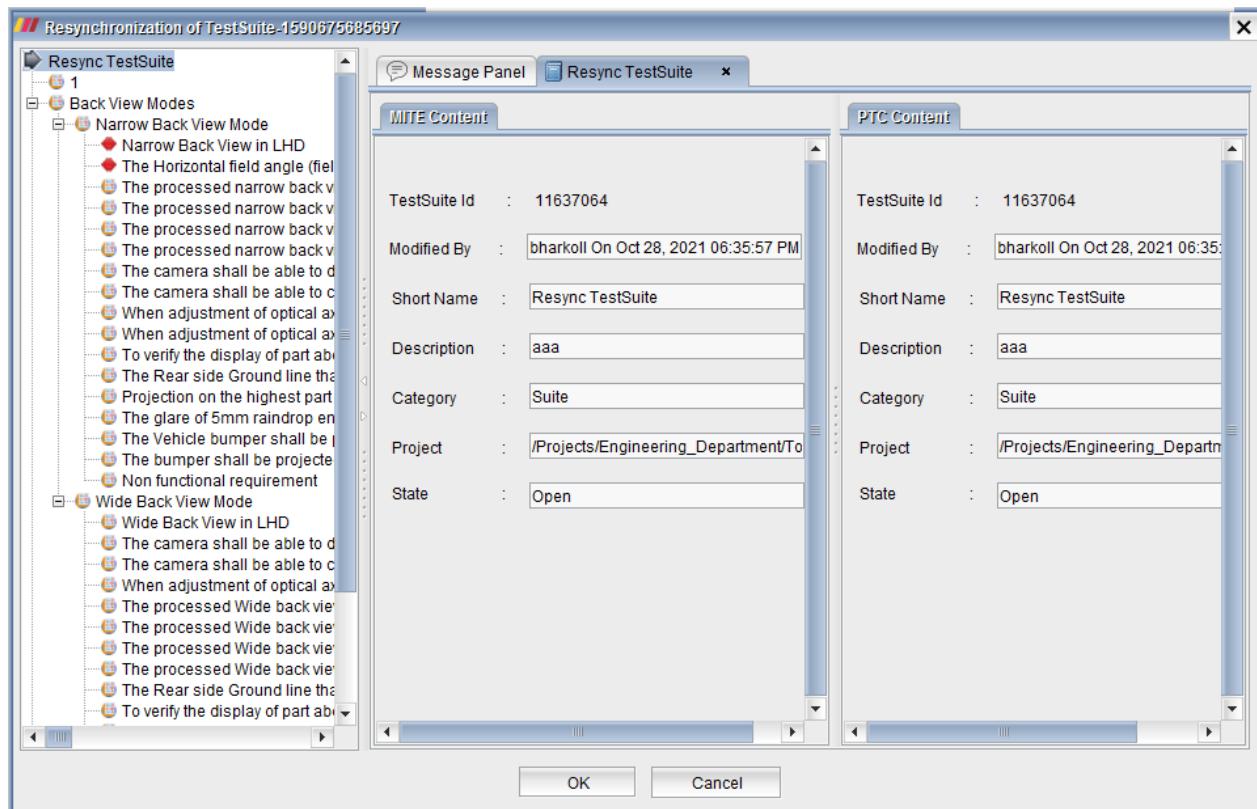
1. In Test Case Level if User select suite and perform right click on it then User will get Resynchronize Test Suite menu.

**Figure 1.1**

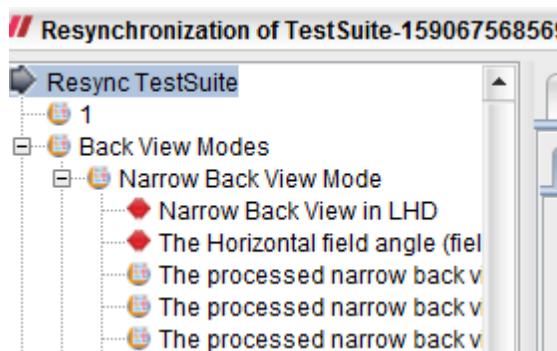
2. If User will click on Resynchronization Test Suite then it will ask for PTC credentials as shown below:-

**Figure 1.2**

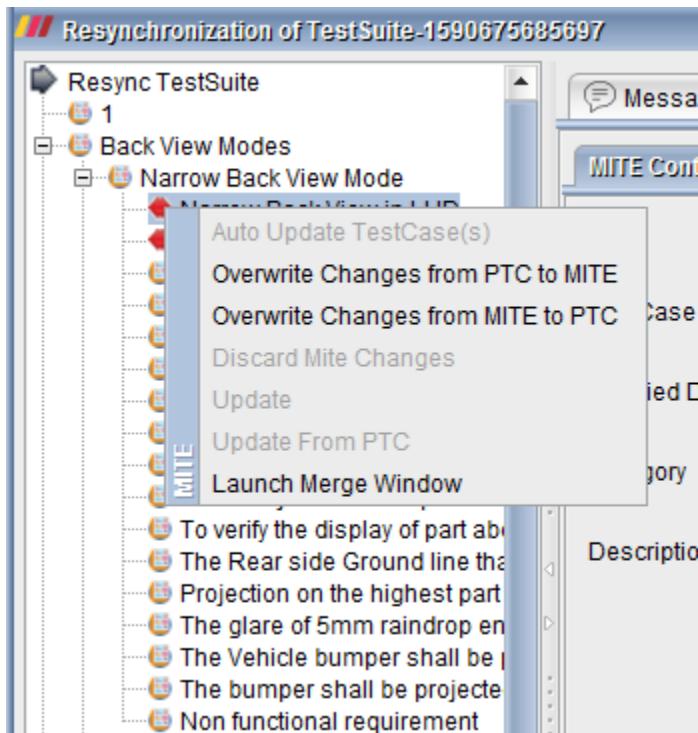
3. After giving the credentials Resynchronization Test Suite window will open.

**Figure 1.3**

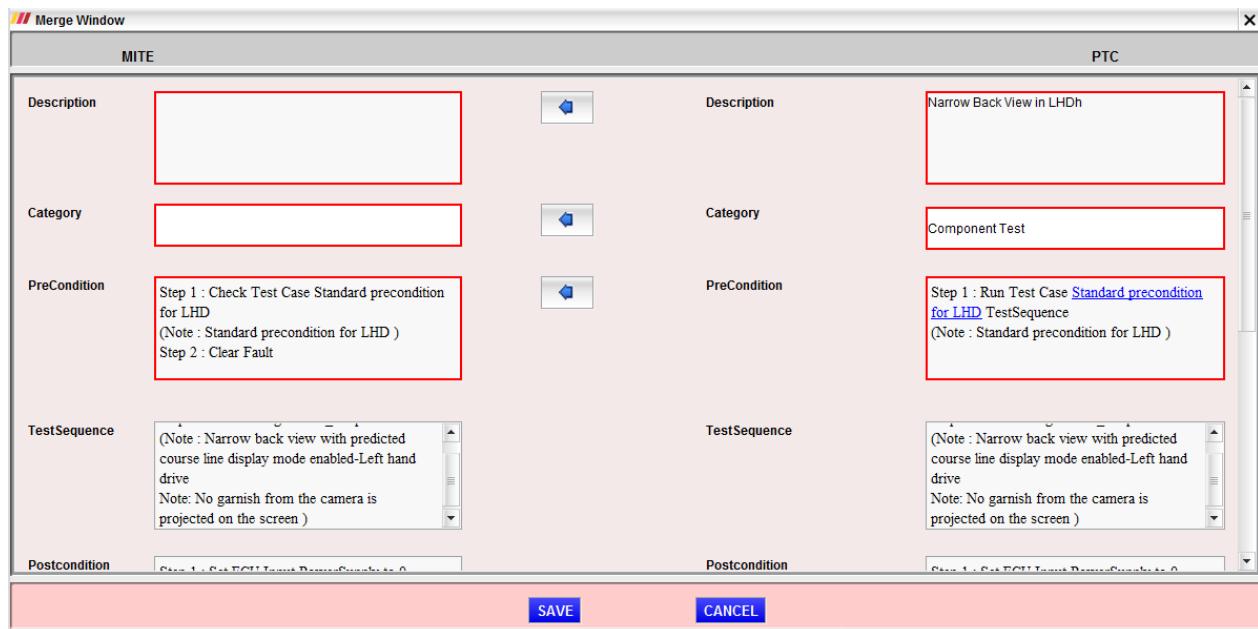
4. In left side of window User will get conflict TestCase (Red color) as shown below:-

**Figure 1.4**

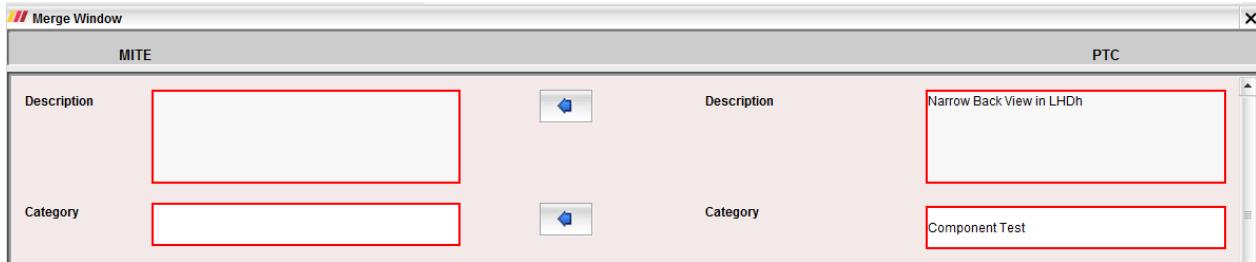
5. If User perform right click on conflict Test Case then User will get Launch Merge window Option as shown below:-

**Figure 1.5**

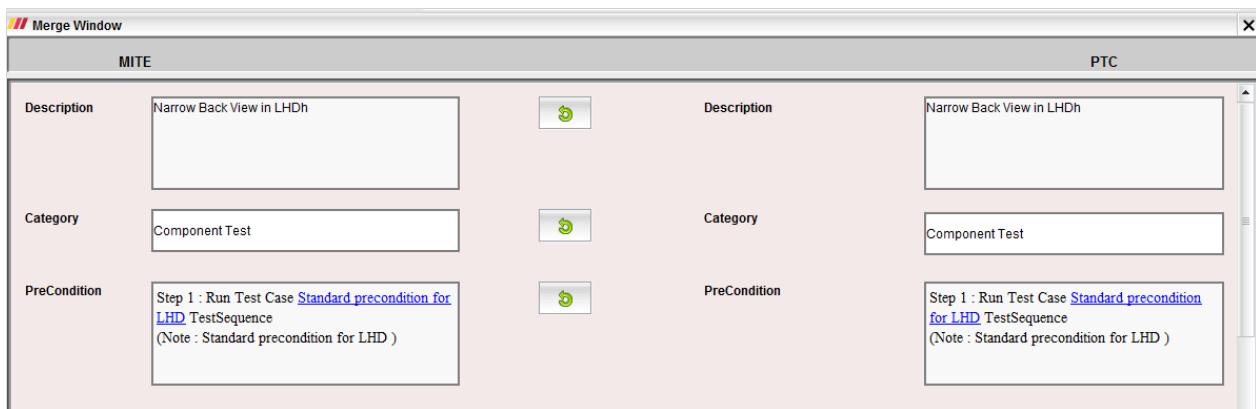
- Upon clicking on Launch Merge Window menu item User will get Launch Merge window.

**Figure 1.6**

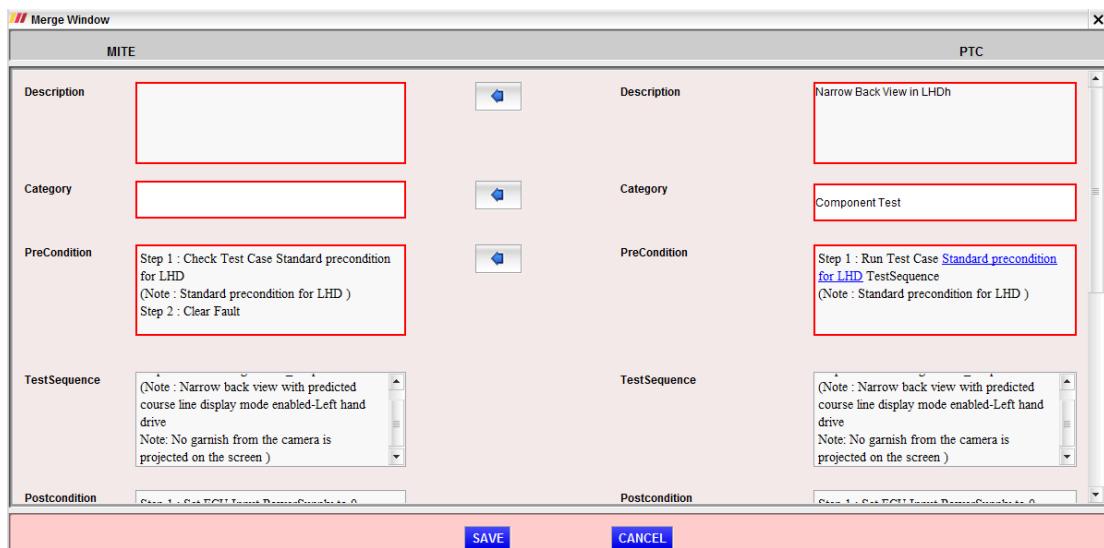
- Here User can view conflict field with red color and if he/she want to take update he/she can click on options (blue arrow mark) in between the field.

**Figure 1.7**

8. After taking update if user think to revert it back before clicking save option and User want previous Mite data again then he/she can click reload option(green color).

**Figure 1.8**

9. After performing operation User has to save his/her data by clicking on save option.

**Figure 1.9**

- 10.** If you don't want to take update you can close the window by pressing cancel button but after pressing cancel button it will ask for you (Do you want to save data).

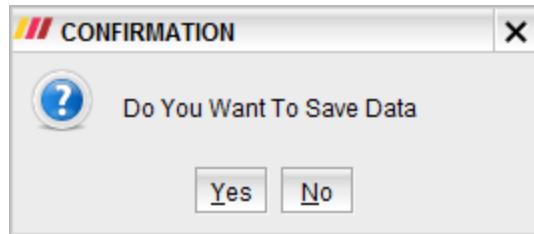


Figure 1.10

- 11.** If you want to save the data you can press yes or if you don't want you can press no.

Created with the Personal Edition of HelpNDoc: [Produce online help for Qt applications](#)

18 Source View Set

Source view set provides the information regarding "MITE" project folder created when PTC flushing occurs. For the first time flushing of Test-Suite the below window will appear;

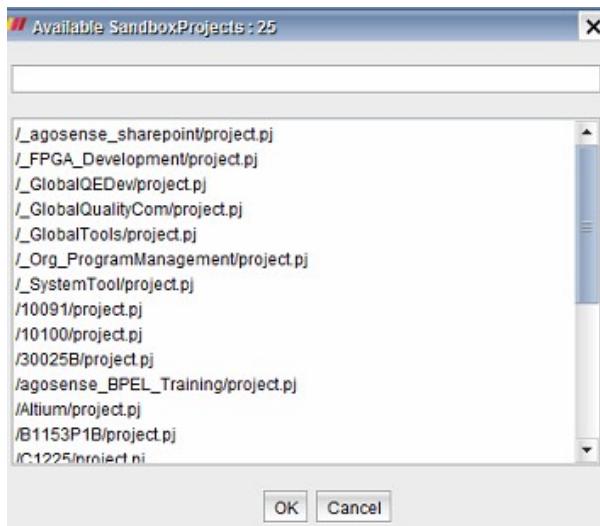


Figure 70: Source View Set selection

Here, MITE will ask the user to provide a path to create MITE folder by selecting a project from the list. This path will be remembered for future flushing of that particular project only. On second time flushing of the same Test-Suite the same path will be used and the "Source View Set selection" window will not appear.

A valid PTC Login credentials must provide to complete the flushing procedure, as shown in below figure:

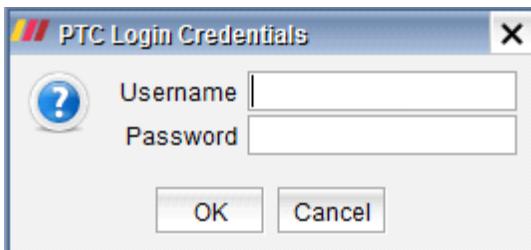


Figure 71: PTC Login credentials

Note: The above authentication is only require for the first time flashing of the project
Once the selected Test-Suite is flushed successfully, MITE project will be created in PTC
(at the selected source view set) as shown below:

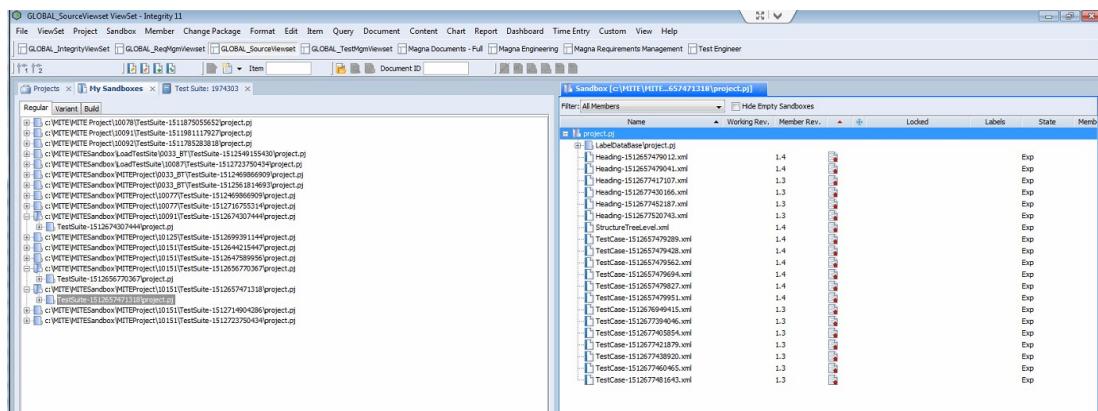


Figure 72: MITE project folder in PTC source view set

No	Recorder Name	Signal Name	Rate(m/s)	Enable/Disable
1	rec1	LINBUS	1230	true
2	rec2			false

Figure 73: MITE Test case in PTC source view set

19 Script generation

To start with Test script generation – User should select “**Test Script**” option from the list.

Test Script generation can be done automatically in MITE and input it takes is MITE format Test cases. MITE UI has three different view sets as shown below.

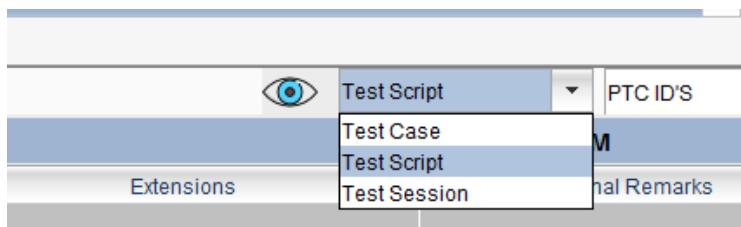


Figure 74: Select Test Script View Set

MITE script generation process mainly depends on the following elements:

1. MITE Test cases
2. Test bench configuration
3. Related library elements and
4. Test data (Parameters, their values).

Created with the Personal Edition of HelpNDoc: [Create cross-platform Qt Help files](#)

19.1 Test Bench Configuration and Label Mapping

To start with Test Bench Configuration and Label Mapping– User should select “**Test Script**” option.

In order to generate the test script which can be executed on a specific test bench, MITE needs to have test bench details. MITE has a separate window to add test bench configuration details.

- Steps to add configuration details:
 1. Select View Set from Eye icon on top right corner
 2. TBC window will activate. Right click on TBC node to add configuration as shown below

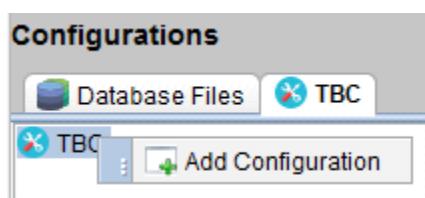
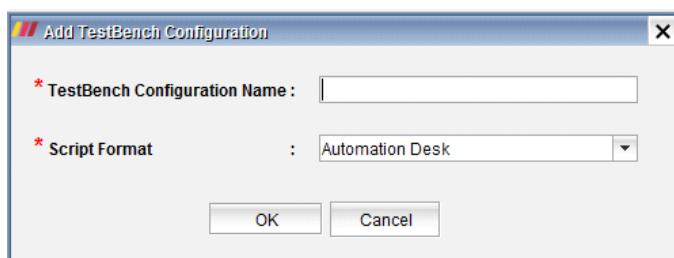


Figure 75: Add TBC Configuration

1. Provide configuration details as bellow



Set Script Format – Currently “Automation Desk” is applicable and Click OK

Figure 76: TBC Configuration Name

2. For selecting Access type (like Model Access or ECU Access) and further details Configuration details by Right click on TBC node and select “Add Configuration Details”

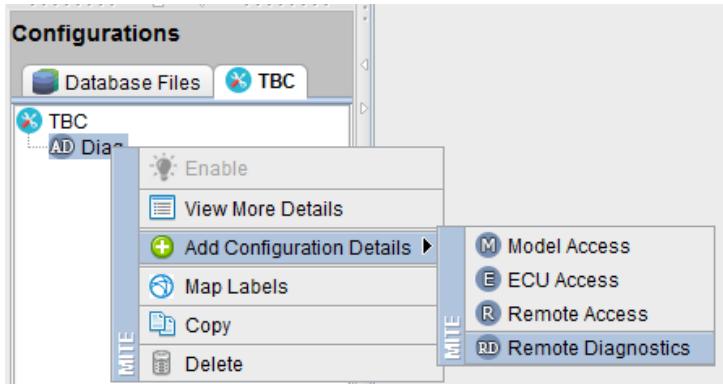


Figure 77: Add Access Details

Model Access:

- Select ECU Access type
- Provide “.sdf” file path to import and fill in the following details
- Click OK

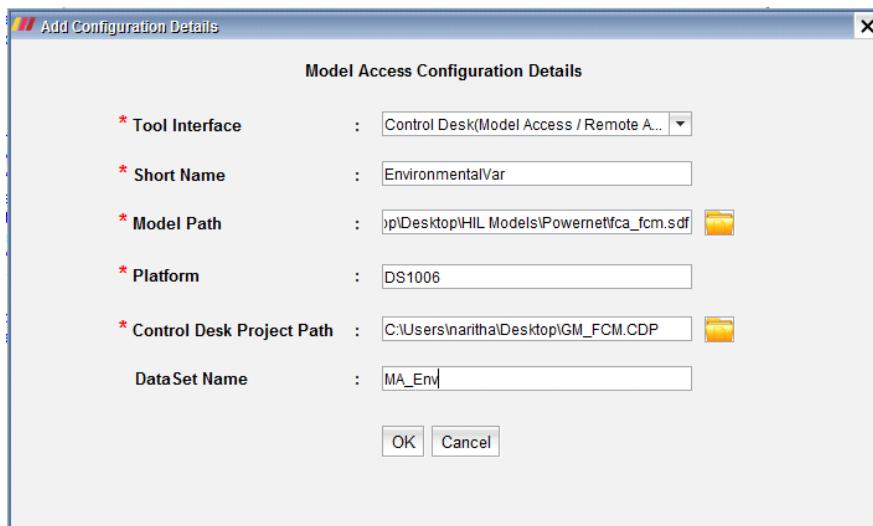


Figure 78 : Add ModelAccess Details

ECU Access:

- Select ECU Access type
- Provide “.A2L” file path to import and fill in the following details
- Click OK

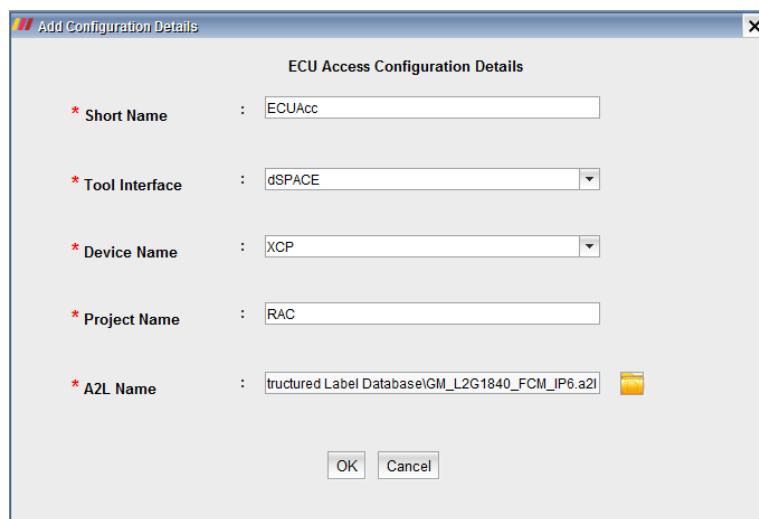


Figure 79 : Add ECUAccess Details

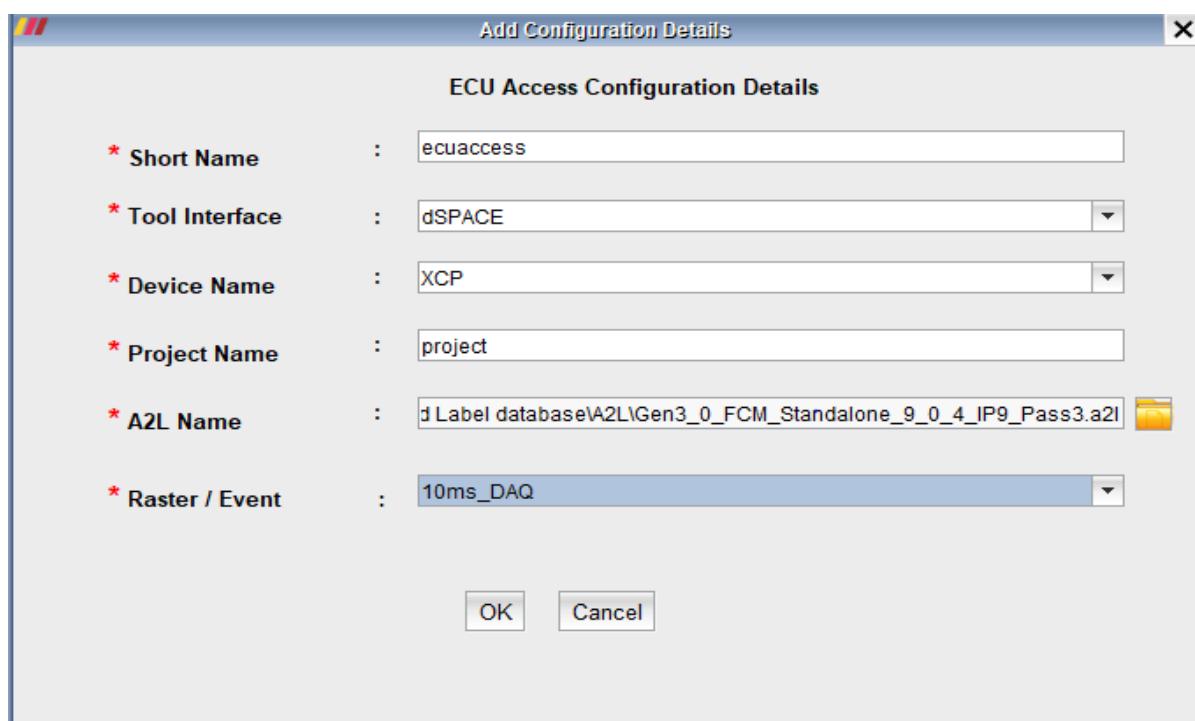
Raster field in ECU Access

Raster can be found in a2l file, User needs to select one of the raster from the extraction of A2L file.

Raster field is given to user in both adding and reload of A2L file.

Scenario 1:

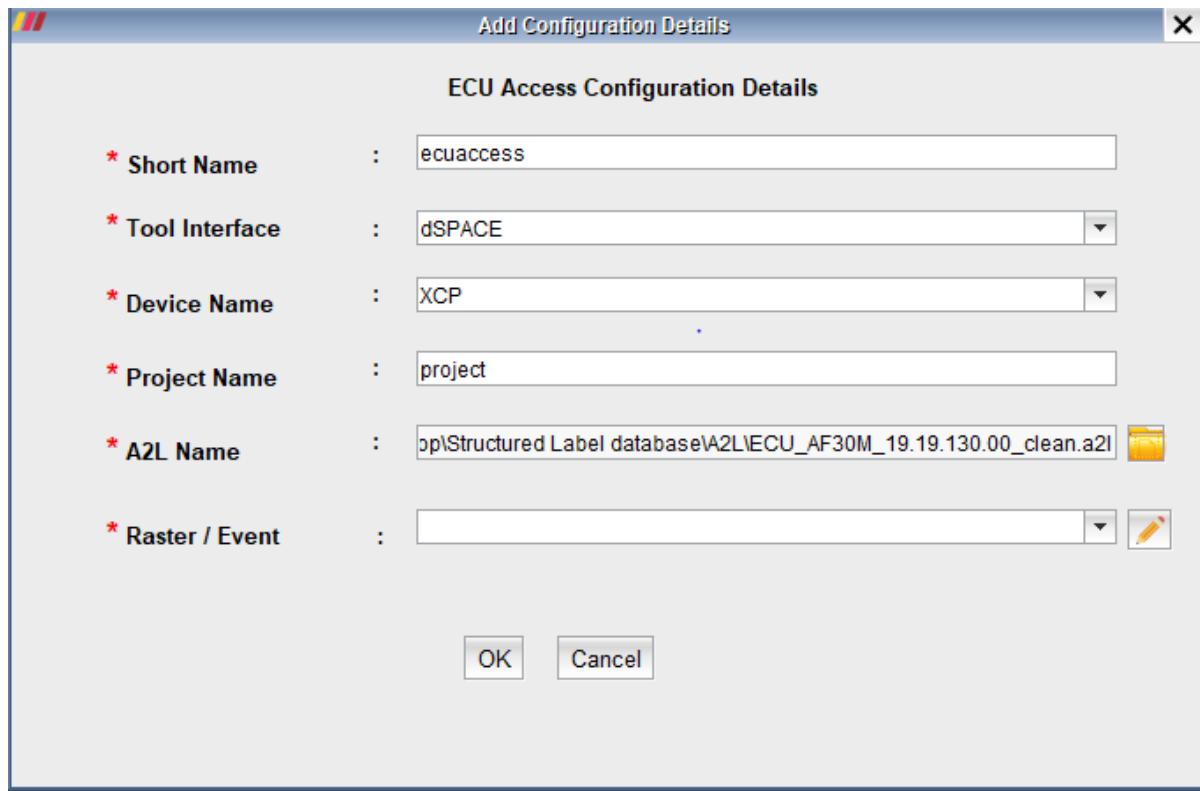
When user add new ECU Access configuration, Click on TBC configuration which user want to add configuration file and select ECU access.



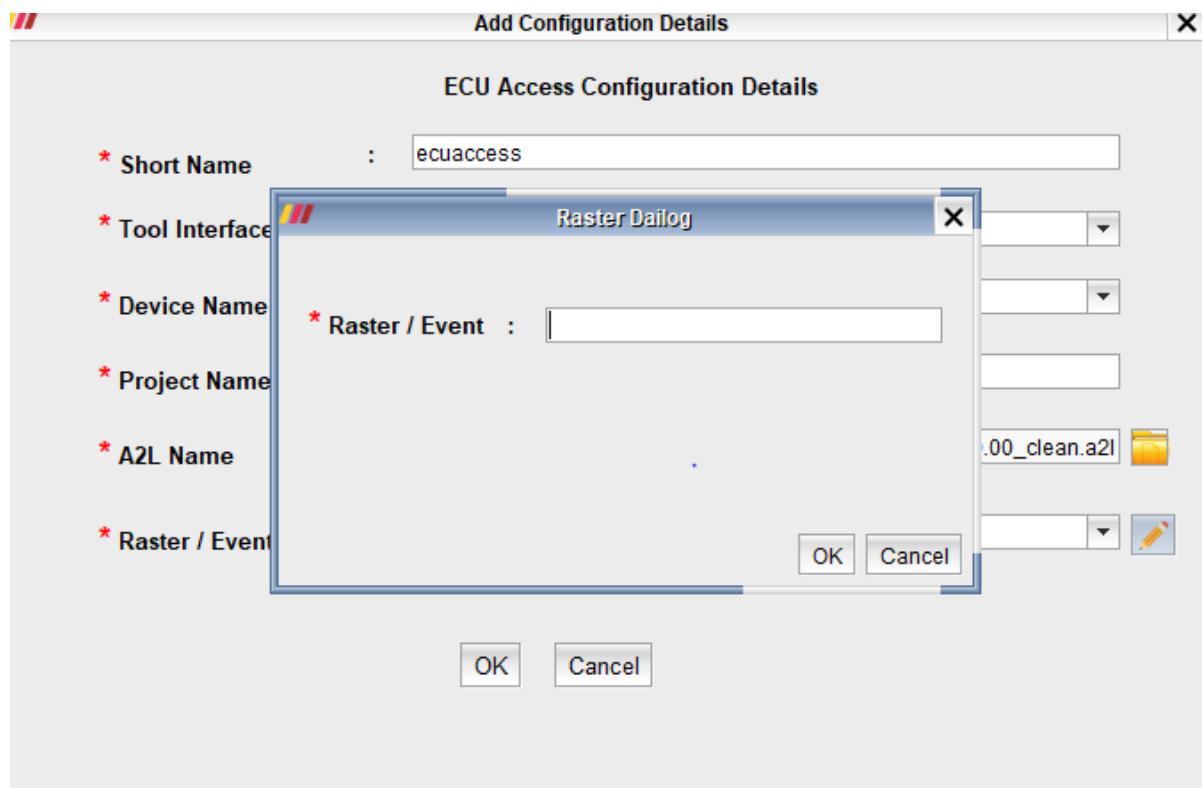
Raster is filled using the A2L file.

Scenario 2:

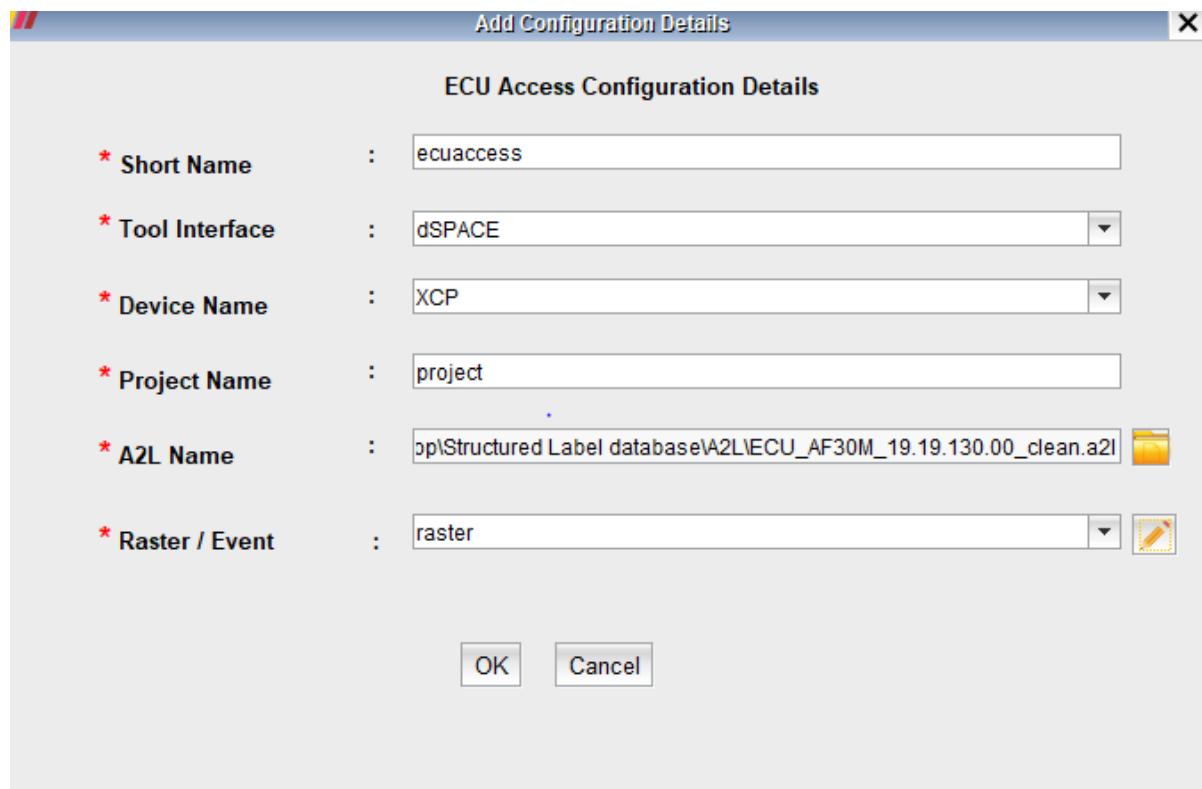
When the Raster/Events are not present then click on the raster button available beside the bar then raster file pop up will occur.

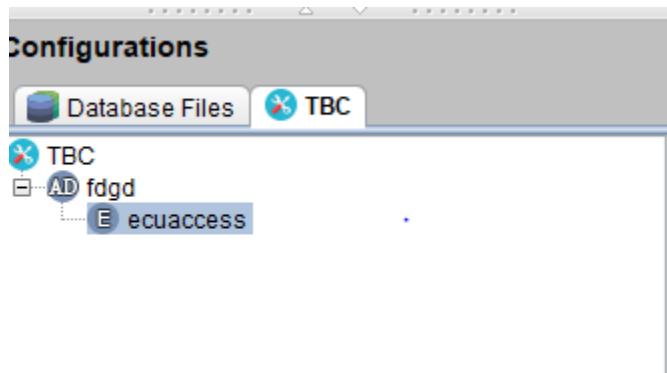


After clicking



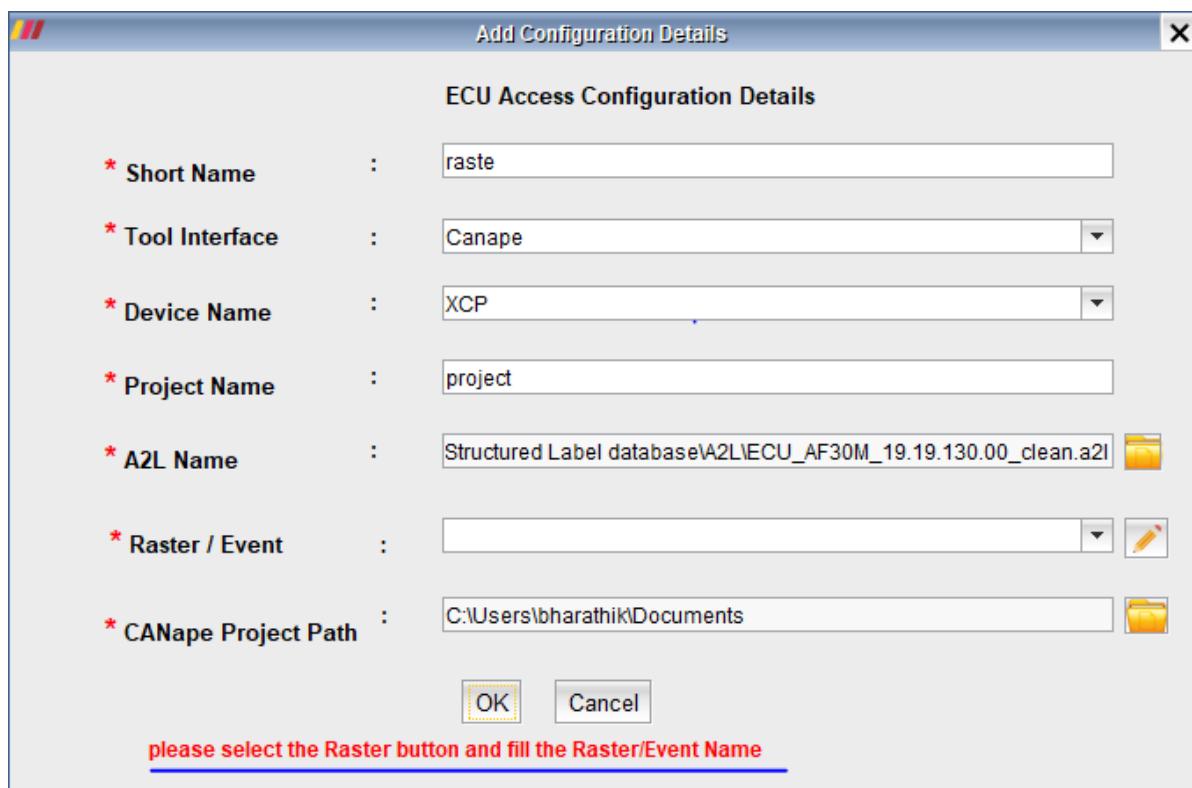
Raster dialog will occur then give the raster name and click on ok.option.





Scenario 3:

Raster filed is a mandatory filed so user need to fill the column.

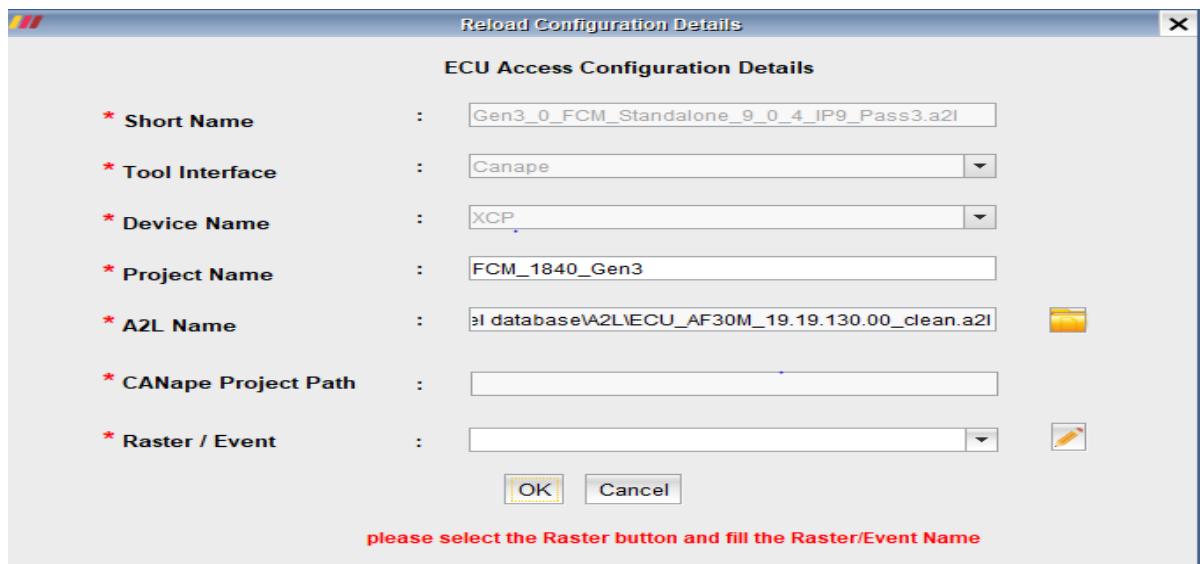


Giving the error that please select the raster button and fill the raster/event name. if user didn't give anything in the mandatory filed.

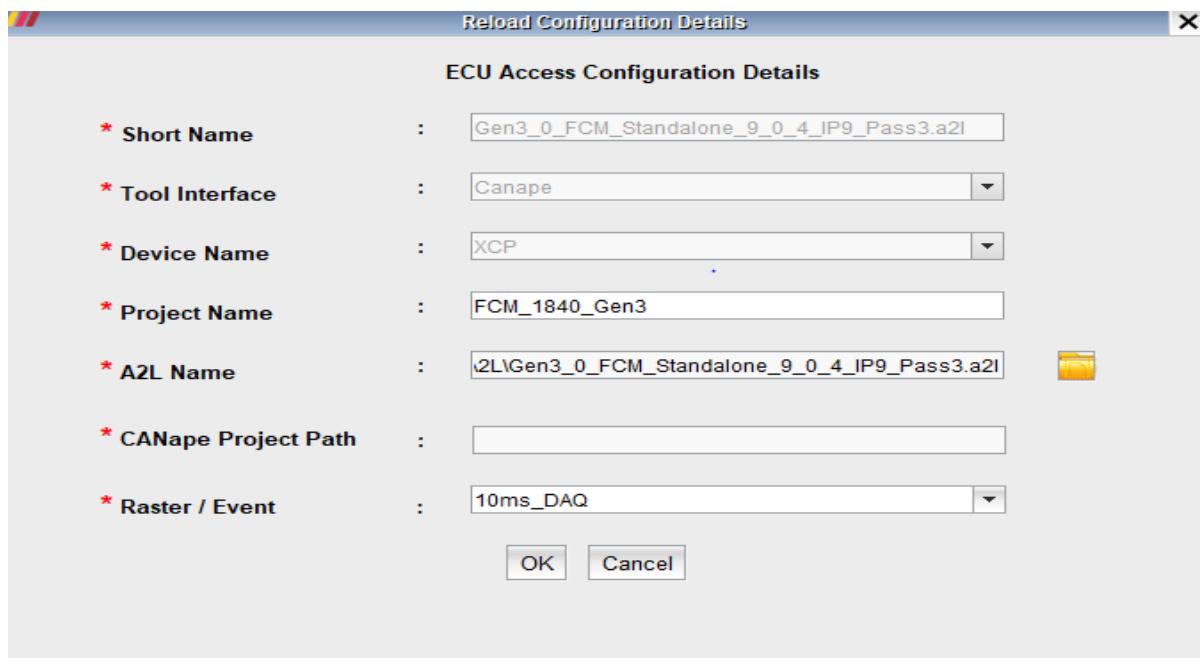
Scenario 4: Using Reload Option:

Step :1 To change the raster name we need to reload again.

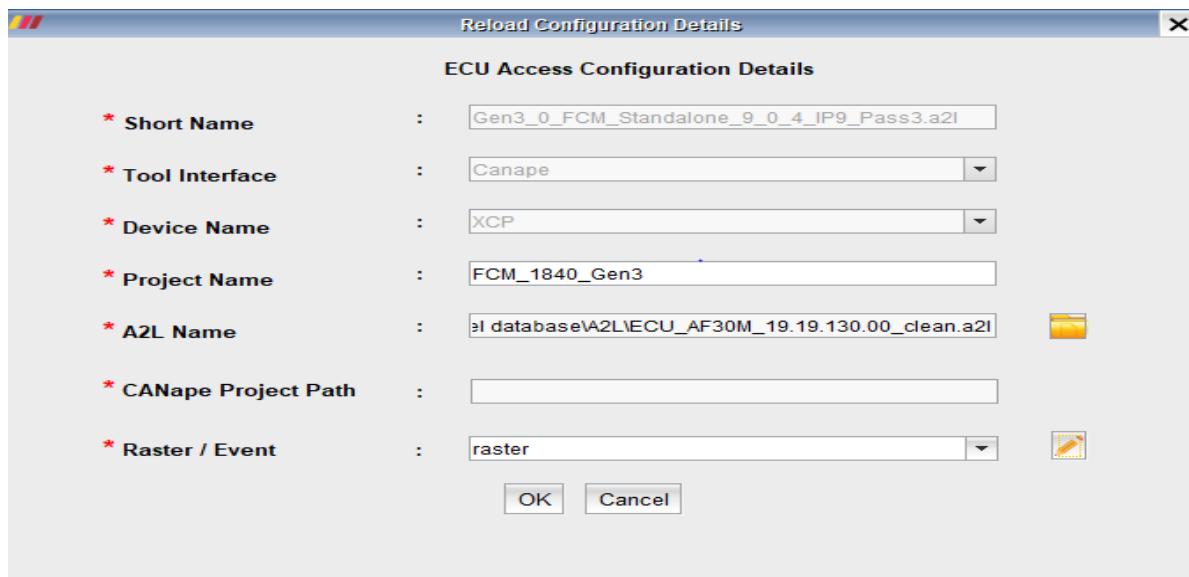
Step :2 Before we have no raster field, now after reloading the file, click on ok and following pop up will occur.

**Scenario 5:**

Reload the used file and change the file based on the requirement and it will extract the raster names.

**Scenario 6:**

If user reload and there is no raster in the file then click on the Icon beside Raster/event



Remote Access:

- Select Remote Access type
- Select Tool interface
- As per tool interface provide the file path to import and fill in the following details
- Click OK

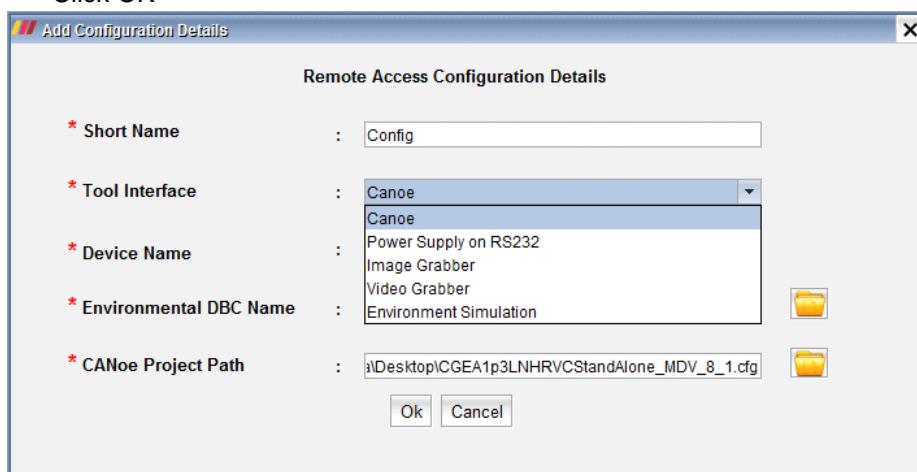


Figure 80 : Remote Access Tool interface(s)

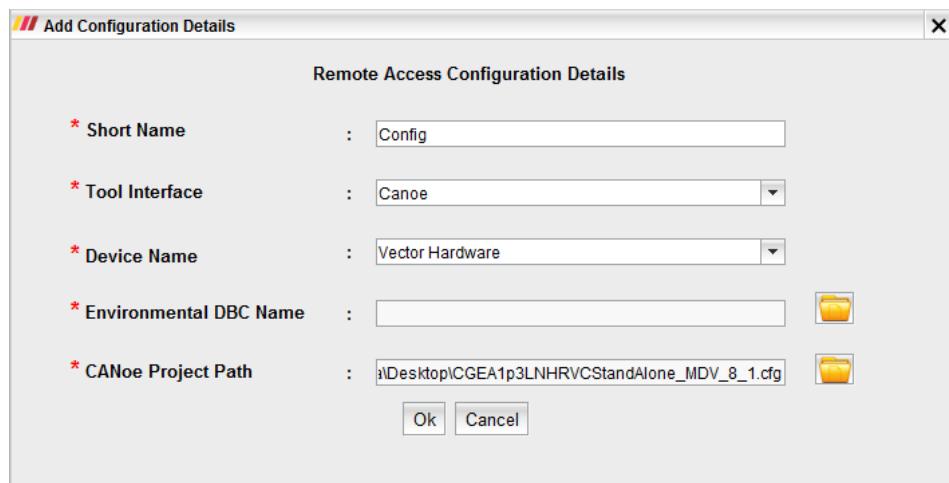


Figure 81 : Remote Access Details

Remote Diagnostics:

- Select Diagnostics Access type
- Provide “.odx or .pdx” file path to import and fill in the following details
- Click OK

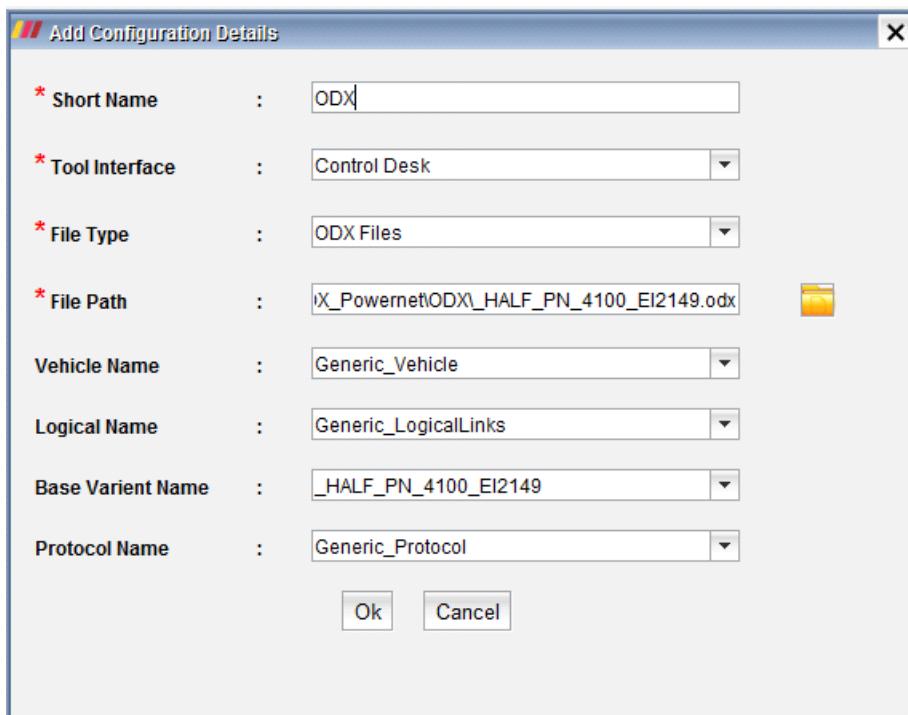
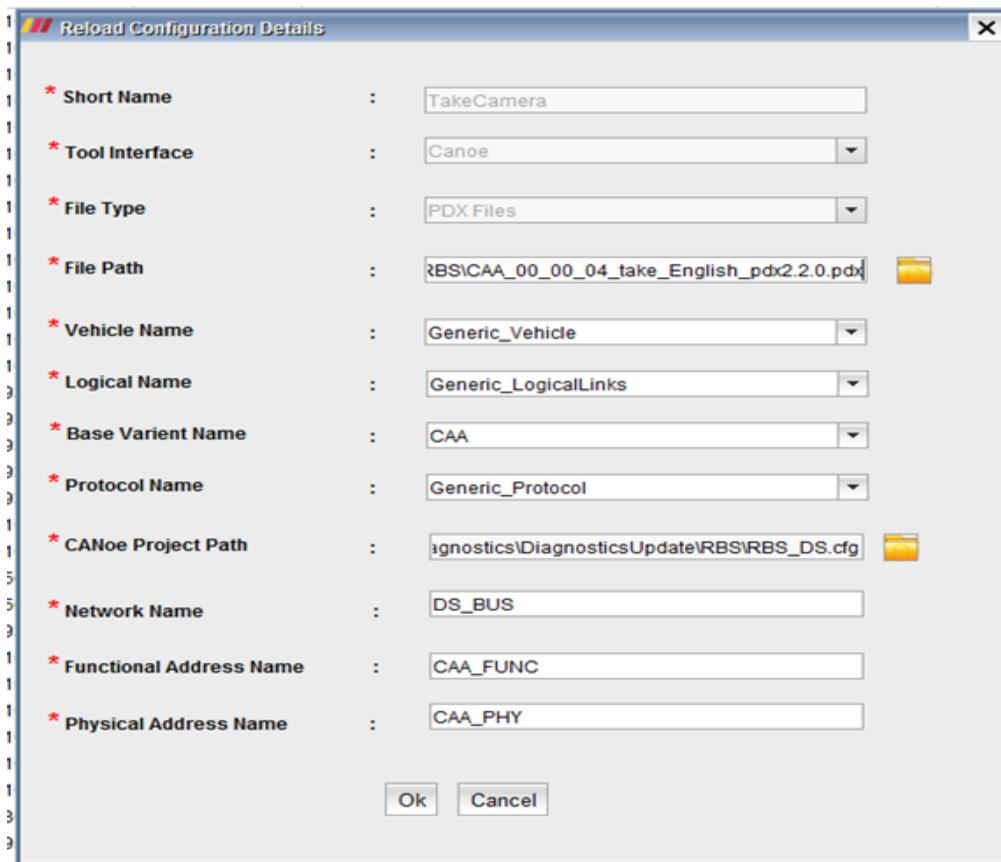


Figure 82 : Remote Diagnosis Details

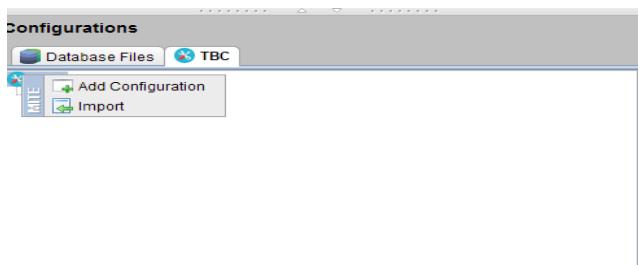
When the Tool interface is Canoe:



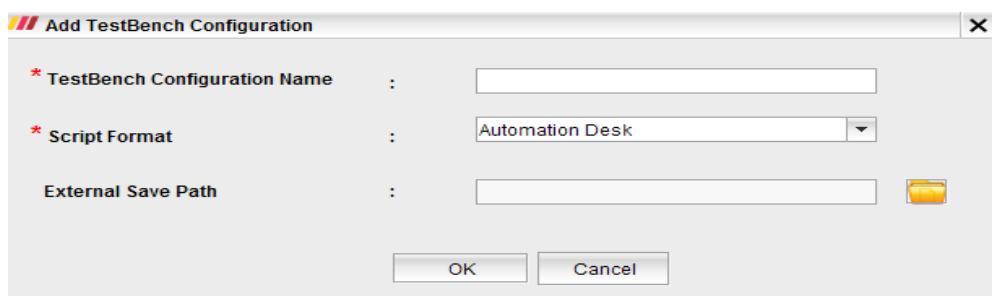
Created with the Personal Edition of HelpNDoc: [Free EPub and documentation generator](#)

19.1.1 Motion Desk TBC

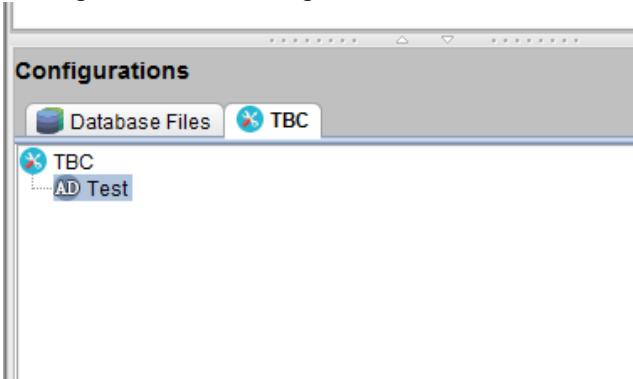
When User Right click on test bench configuration, he/she will get the details like add test bench configuration and import.



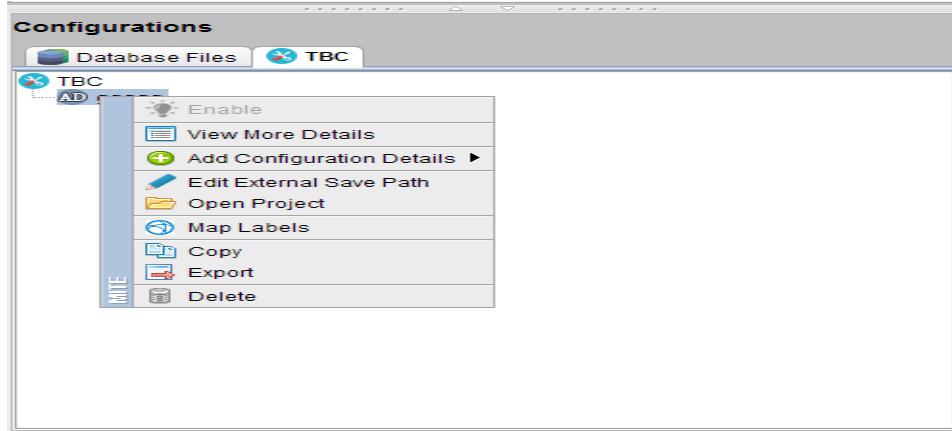
When we click on add configuration we will get pop up as shown in figure,



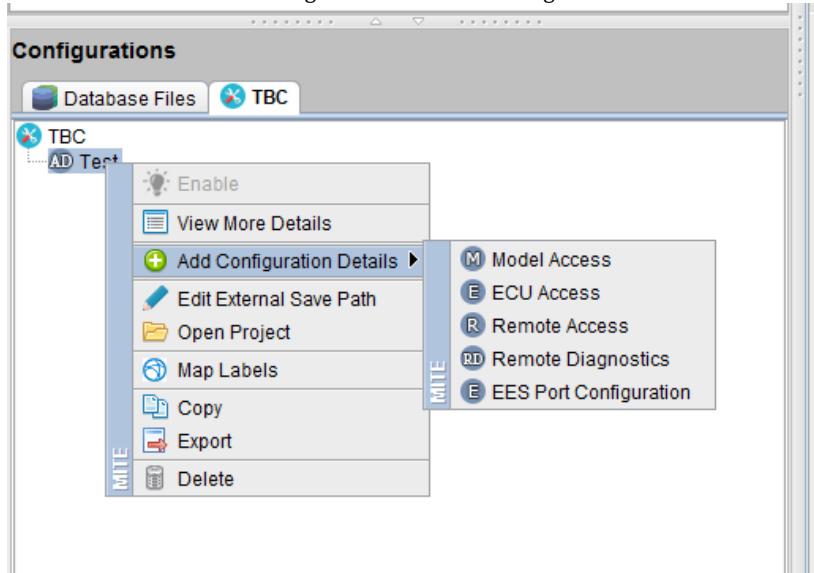
In popup Test bench configuration name, Script format, External save path, all should be filled, this will be added to test bench configuration as shown in figure,



When we Right Click on AD the following details are shown in Figure,



When we click on add configuration details we will get the details as shown in figure,

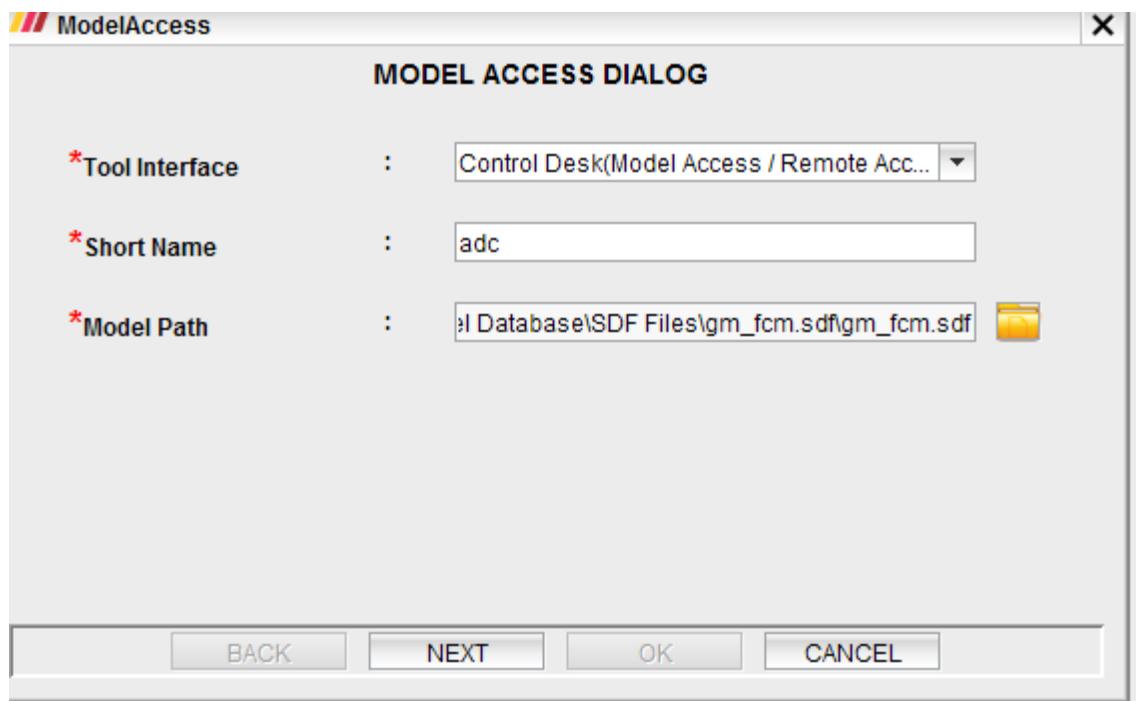


Design and Implementation:-

1) For Model Access Type File:-

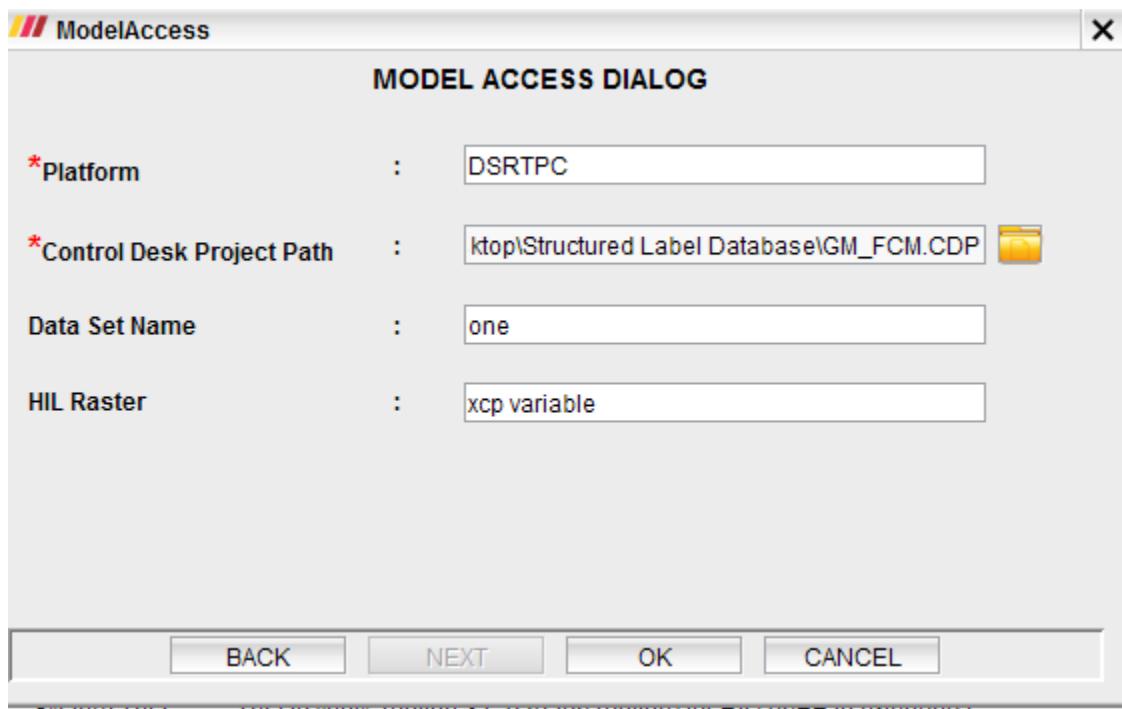
When User wants to add Model Desk Access file to the Configuration then right click on the configuration file click on "Add Configuration Details" click on "Model access" then a window will open as shown in the below.

1.1 Control Desk/Model Access/Remote Access:-



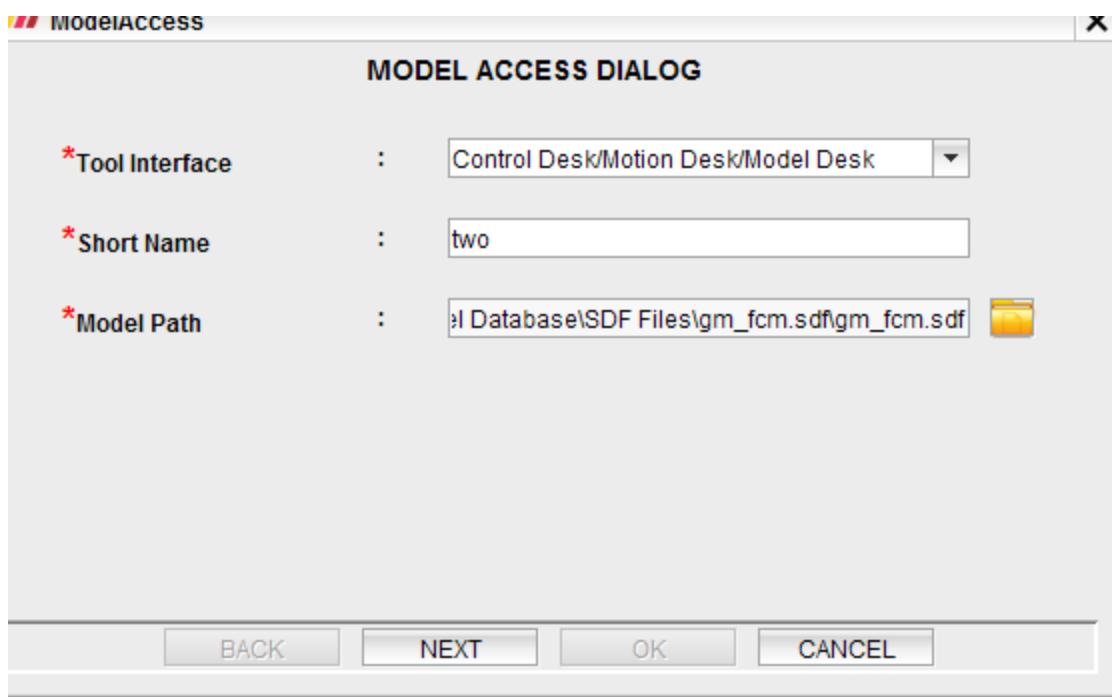
Then User has to fill the following details:-

- Tool Interface: - <Node Name>.
- Short Name:-As per the user
- Model Path:- Import SDF File.



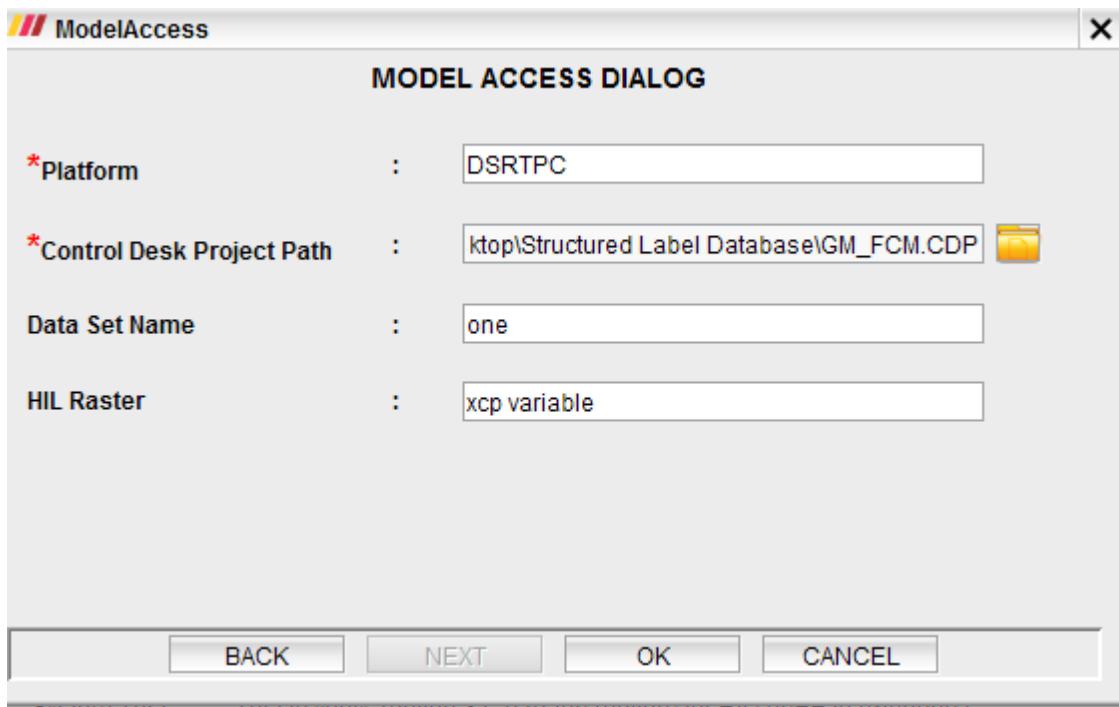
1.2 Control Desk/Motion Desk/Model Desk:-

Select the Tool Interface as "Control Desk/Motion Desk/Model Desk" then window is turn like below.



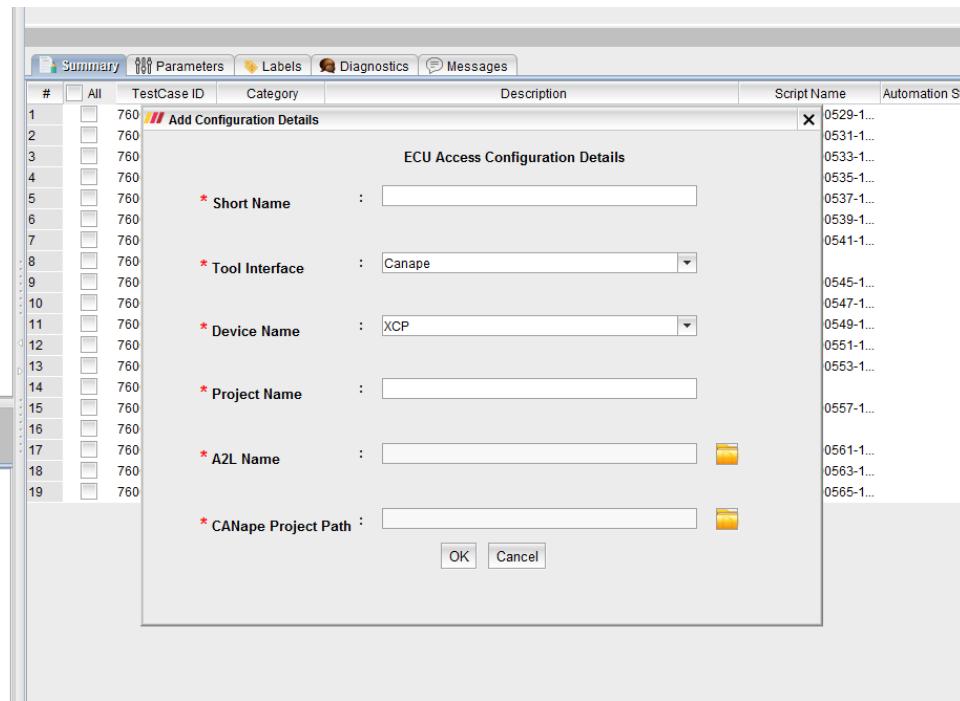
Then User has to fill the following details:-

- Tool Interface: - <Node Name>.
- Short Name:-As per the user
- Model Path: - Import SDF File.
-



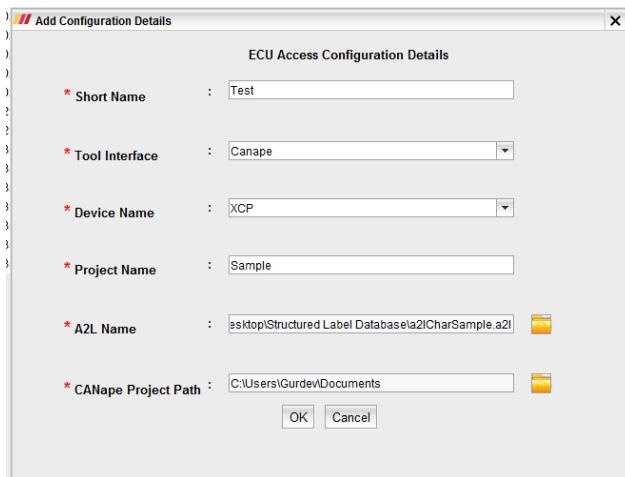
2) For ECU Access type file:-

When the user wants to add ECU Access file to the Configuration then right click on the configuration file click on "Add Configuration Details" click on "ECU Access " one window will open as shown in the below.



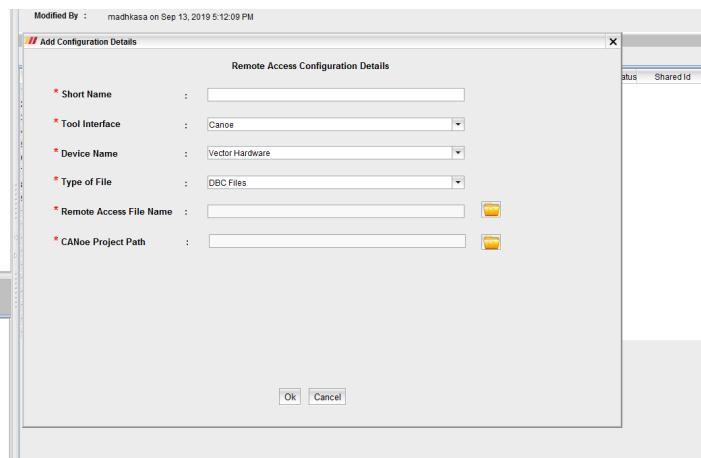
Then User has to fill the following details:-

- Short Name:-As per the user
- Tool Interface: - <Node Name>
- Device Name: - XCP.
- Project Name --> Please fill the Project Name.
- A2L Name --> Import A2L File.
- CANape Project Path --> Please select the folder path containing CANape.ini file.



3) For Remote Access type file:-

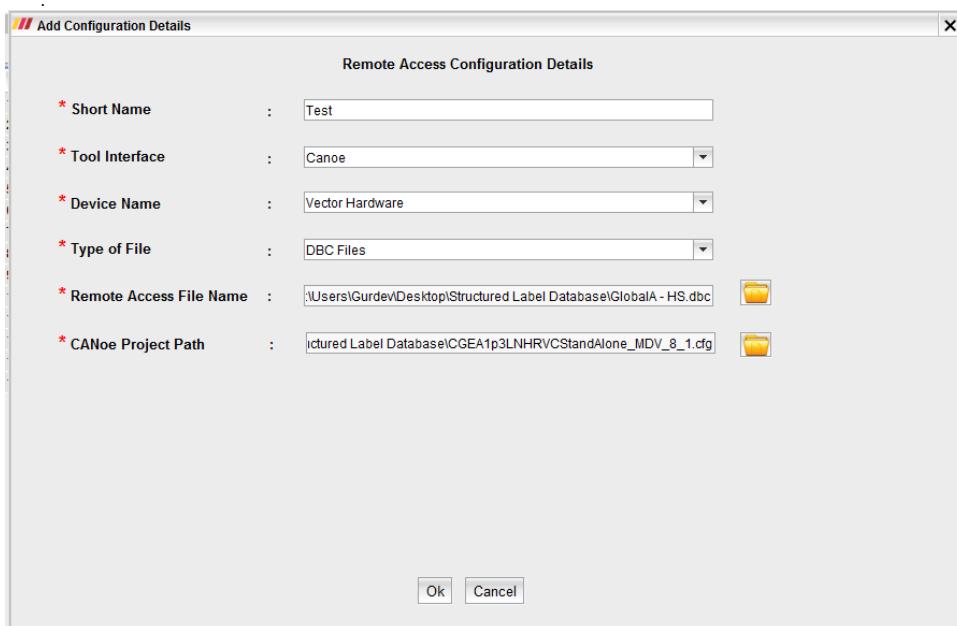
When the user wants to add Remote Access file to the Configuration then right click on the configuration file click on "Add Configuration Details" click on "Remote Access" then a window will open as shown in the below.



3.1 CANoe:-

User has to fill the following details:-

- Short Name --> <Node Name>.
- Device Name --> Vector Hardware.
- Types of File --> Please select the file Type from the dropdown list.
- Remote Access File Name --> Please select the file based on the file Type.
- CANoe Project Path --> Please select the CANoe Configuration file.
- CANoe Bus Name --> Please fill the CANoe Bus Name for the ARXML File Type.
- CANoe Channel --> Please fill the CANoe Channel for the ARXML File Type



3.2 Power Supply on RS232:-

Add Configuration Details

Remote Access Configuration Details

* Short Name :	<input type="text"/>
* Tool Interface :	Power Supply on RS232 <input type="button" value="Edit"/>
* Device Name :	BK Precision 1697 <input type="button" value="Edit"/>
* Type of File :	<input type="text"/>

User has to fill the following details:-

- Short Name --> <Node Name>.
- Device Name --> Please select the relevant Power Supply Model from the dropdown list.

Add Configuration Details

Remote Access Configuration Details

* Short Name :	<input type="text" value="Powersupply"/>
* Tool Interface :	Power Supply on RS232 <input type="button" value="Edit"/>
* Device Name :	BK Precision 1697 <input type="button" value="Edit"/>
* Type of File :	<input type="text"/>

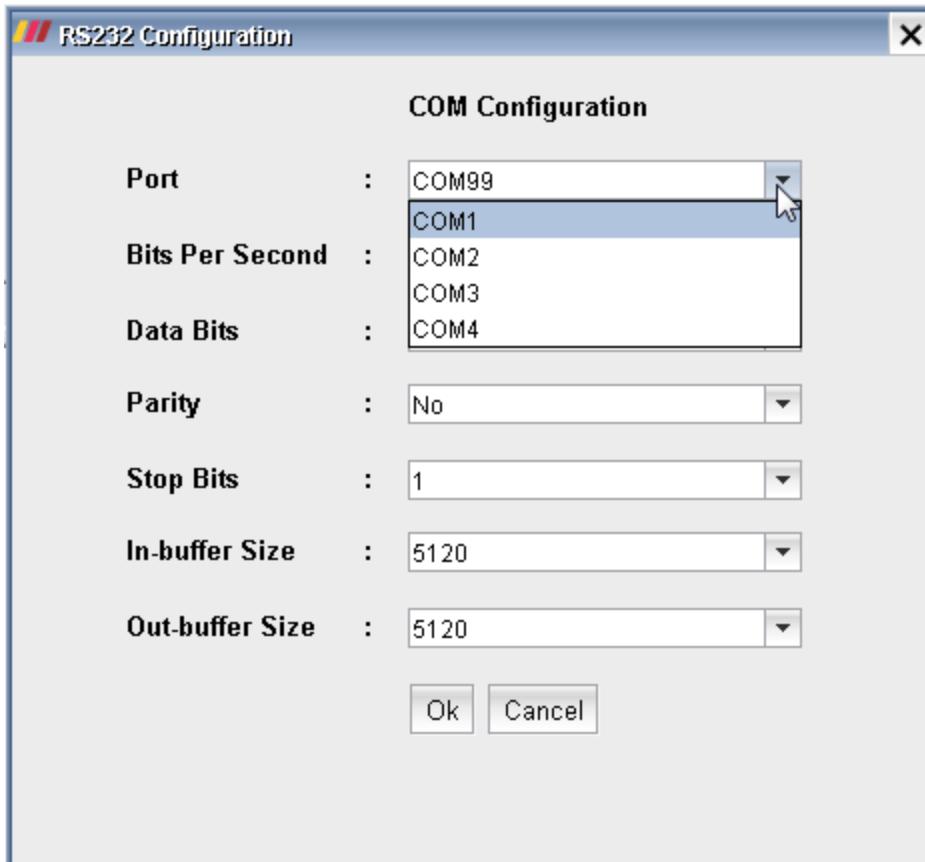
Add Configuration Details

Remote Access Configuration Details

* Short Name :	<input type="text" value="Powersupply"/>
* Tool Interface :	Power Supply on RS232 <input type="button" value="Edit"/>
* Device Name :	BK Precision 1697 <input type="button" value="Edit"/>
* Type of File :	<input type="text" value="BK Precision 1697"/> <input type="text" value="TDK-Lambda-ZSeries"/> <input type="text" value="TDK-Lambda-GENSYS"/>

Additional COM Configuration

- Port: Please select from the dropdown list.
- Bits per Second: Please select from the dropdown list.
- Data Bits: Please select from the dropdown list.
- Parity: Please select from the dropdown list.
- Stop Bits: Please select from the dropdown list.
- In-buffer Size: Please select from the dropdown list.
- Out-buffer Size: Please select from the dropdown list.



Note: The Port Number should be in the form of 'COMn', where n is any number from 1 to 99

3.3 Image Grabber:-

Then User has to fill the following details:-

- Short Name --> <Node Name>.
- Device Name --> Please select the relevant Power Supply Model from the dropdown list.
- Reference Images File Path --> Please select the folder of the Images.

Remote Access Configuration Details

* Short Name : ImageGrabber

* Tool Interface : Image Grabber

* Device Name : Debut

* Type of File : Debut
AmCap

* Reference Images File Path :

Ok Cancel

Remote Access Configuration Details

* Short Name : ImageG

* Tool Interface : Image Grabber

* Device Name : AmCap

* Type of File :

* Reference Images File Path :

Ok Cancel

3.4 Video Grabber:-

User has to fill the following details:-

- Short Name --> <Node Name>.
- Device Name --> Please select the relevant Power Supply Model from the dropdown list.
- Video Recorder Path --> Please select the folder of the Videos.

Remote Access Configuration Details

* Short Name : VideoG

* Tool Interface : Video Grabber

* Device Name : CANoe

* Type of File : CANoe
Debut
AmCap

* Video Record Path : C:\Users\Gurdev\Desktop\Structured Label Database

Ok Cancel

Remote Access Configuration Details

* Short Name : VideoG

* Tool Interface : Video Grabber

* Device Name : Debut

* Type of File :

* Video Record Path : C:\Users\Gurdev\Desktop\Structured Label Database

Ok Cancel

3.5 Environment Simulation:-

Then User has to fill the following details:-

- Short Name --> <Node Name>.
- Device Name --> Please select the relevant Power Supply Model from the dropdown list.
- Reference Videos File Path --> Please select the folder of the Videos.
- Height --> Please fill the height of the screen that Video has to be played.
- Width --> Please fill the width of the screen that Video has to be played.
- X-Coordinate --> Please fill the X-Coordinate offset the screen that Video has to be played.
- Y-Coordinate --> Please fill the Y-Coordinate offset the screen that Video has to be played.

Add Configuration Details

Remote Access Configuration Details

* Short Name :	ENVIRONSIMU
* Tool Interface :	Environment Simulation
* Device Name :	Offline Video
* Type of File :	Offline Video Car Maker
* Reference Videos File Path :	<input type="text"/> 
* Height : <input type="text"/> * Width : <input type="text"/> * X-Coordinate : <input type="text"/> * Y-Coordinate : <input type="text"/>	
<input type="button" value="Ok"/> <input type="button" value="Cancel"/>	

Add Configuration Details

Remote Access Configuration Details

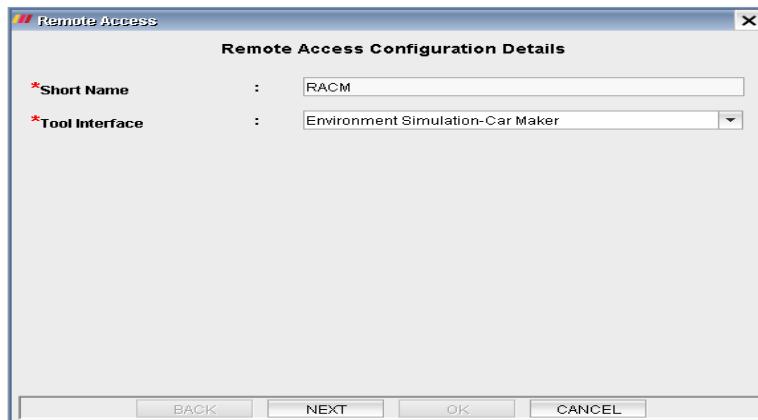
* Short Name :	ENVIRONSIMU
* Tool Interface :	Environment Simulation
* Device Name :	Offline Video
* Type of File :	<input type="text"/>
* Reference Videos File Path :	C:\Users\Gurdev\Documents 
* Height : 8990 * Width : 45656 * X-Coordinate : 6666 * Y-Coordinate : 66	
<input type="button" value="Ok"/> <input type="button" value="Cancel"/>	

Add Configuration Details

Remote Access Configuration Details

* Short Name :	Ewirn
* Tool Interface :	Environment Simulation
* Device Name :	Car Maker
* Type of File :	<input type="text"/>
* Car Maker Path :	C:\Users\Gurdev\Desktop\Structured Label Database 
<input type="button" value="Ok"/> <input type="button" value="Cancel"/>	

3.6 Environment Simulation – Car Maker

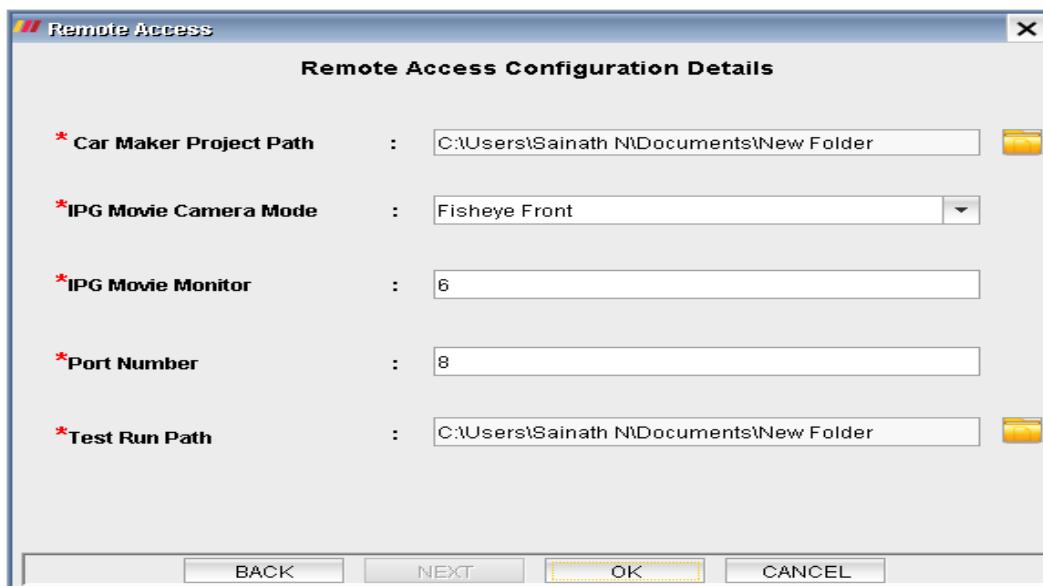


User has to fill the following details:

Short Name -> <Node Name>

Tool Interface -> Environment Simulation-Car Maker

Click Next button.



Car Maker Project Path -> Please select the folder which contains Car Maker Project Path.

IPG Movie Camera Mode -> Please fill the IPG Camera Mode or select it from drop down.

IPG Movie Monitor -> Please fill the IPG Movie Monitor, only integer values are allowed.

Port Number -> Please fill the Port Number, only integer values are allowed.

Test Run Path -> Please select the folder which contains Test Run Path.

Click OK, it gets added into TBC.

4) For Remote Diagnostics Type File:-

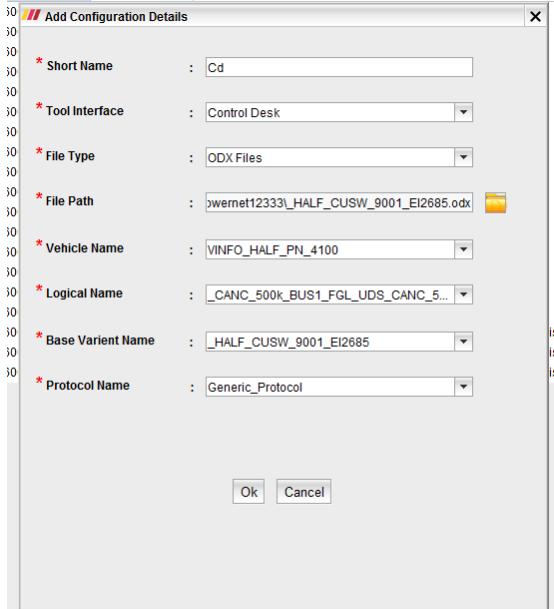
When the user wants to add Remote Diagnostics file to the Configuration then right click on the configuration file click on "Add Configuration Details" click on "Remote Diagnostics " one window will open as shown in the below.

4.1 Control Desk:-

User has to fill the following details:-

- Short Name --> <Node Name>.
- File Type --> Please select from the dropdown list.
- File Path --> Please select the file path based on the File Type.

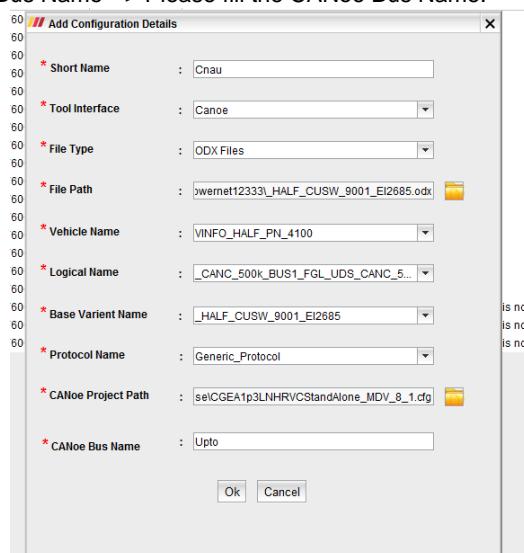
- Vehicle Name --> Please select from the dropdown list.
- Logical Name --> Please select from the dropdown list.
- Base Variant Name --> Please select from the dropdown list.
- Protocol Name --> Please select from the dropdown list.



4.2 CANoe:-

User has to fill the following details:-

- Short Name --> <Node Name>.
- File Type --> Please select from the dropdown list.
- File Path --> Please select the file path based on the File Type.
- Vehicle Name --> Please select from the dropdown list.
- Logical Name --> Please select from the dropdown list.
- Base Variant Name --> Please select from the dropdown list.
- Protocol Name --> Please select from the dropdown list.
- CANoe Project Path --> Please select the CANoe Configuration file.
- CANoe Bus Name --> Please fill the CANoe Bus Name.



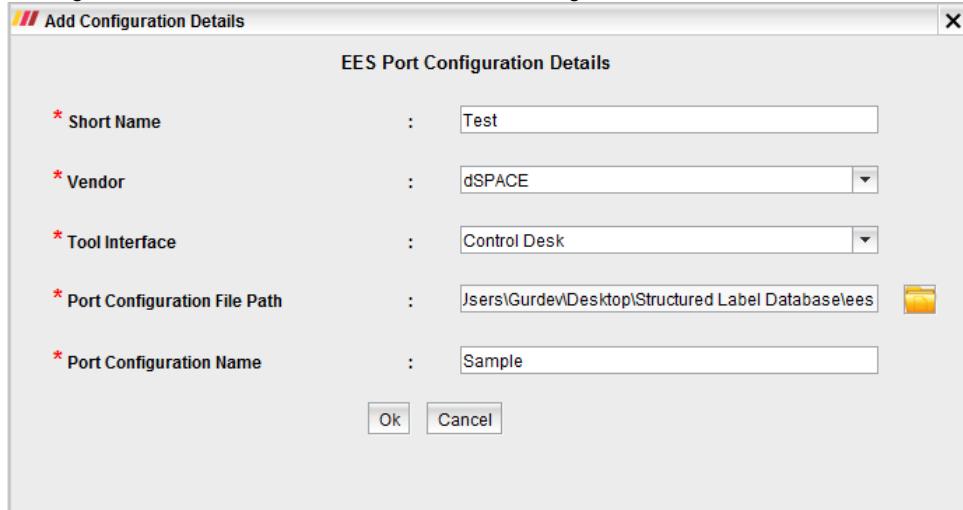
5) For EES Port Configuration Type File:-

When the user wants to add EES Port Configuration file to the Configuration then right click on the

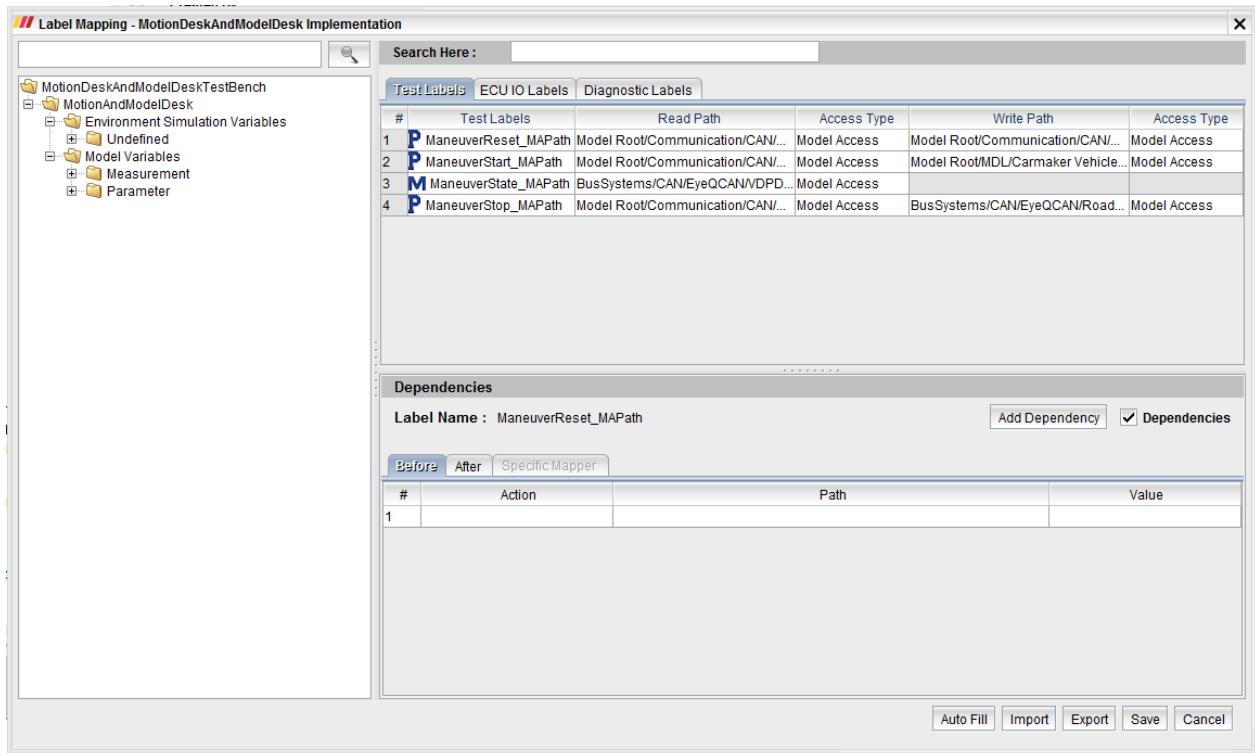
configuration file click on "Add Configuration Details" click on "EES Port Configuration " one window will open as shown in the below.

Then User has to fill the following details:-

- Short Name --> <Node Name>.
- Vendor --> Please select from the dropdown list.
- Project Configuration File Path --> Please select the file for the EES Port Configuration.
- Port Configuration Name --> Please fill the EES Port Configuration Name.



User should Map these Variables for close loop Testing

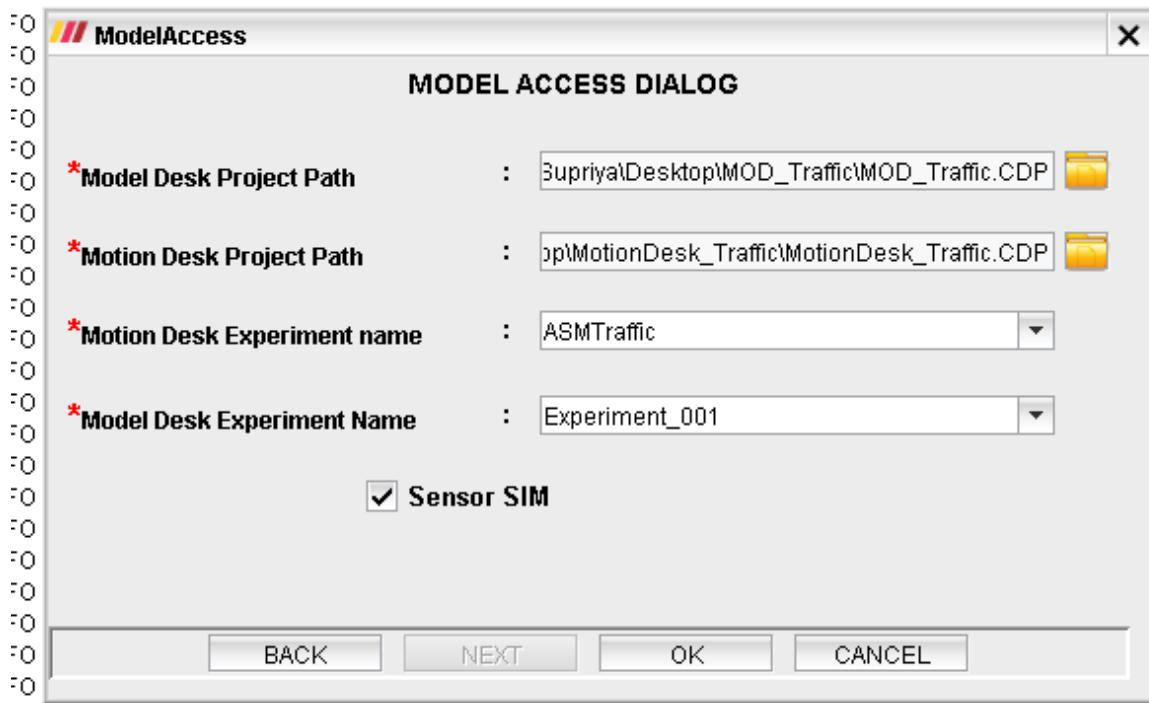


Reference:-

TestBench Configuration	Access Type	Tool Interface	Fields Information
	Model Access	Control Desk	<p>1. Short Name --> <NodeName>.</p> <p>2. Model Path --> Import SDF File.</p> <p>3. Control Desk Project Path --> Select CDP File</p> <p>4. Platform --> Example : SCALAXIO.ds1006.etc...</p> <p>5. Data Set Name --> <Control Desk Data Set Name></p>
			<p>1. Short Name --> <NodeName>.</p> <p>2. Model Path --> Import SDF File.</p> <p>3. Control Desk Project Path --> Select CDP File.</p> <p>4. Platform --> Example : SCALAXIO.ds1006.etc...</p> <p>5. Data Set Name --> <Control Desk Data Set Name>.</p> <p>6. Model Desk Project Path --> Select Model Desk CDP File.</p> <p>7. Motion Desk Project Path --> Select Motion Desk CDP File.</p> <p>8. Motion Desk Experiment Name --> Select from dropdown list.</p> <p>9. Motion Desk Experiment Name --> Select from dropdown list.</p>
	ECU Access	CANApe	<p>1. Short Name --> <NodeName>.</p> <p>2. Device Name --> XCP.</p> <p>3. Project Name --> Please fill the Project Name.</p> <p>4. A2L Name --> Import A2L File.</p> <p>5. CANApe Project Path --> Please select the folder path containing CANApe.ini file.</p>
	Remote Access	Canoe	<p>1. Short Name --> <NodeName>.</p> <p>2. Device Name --> Vector Hardware.</p> <p>3. Type of File --> Please select the fileType from the dropdown list.</p> <p>4. Remote Access File Name --> Please select the file based on the fileType.</p> <p>5. CANoe Project Path --> Please select the CANoe Configuration file.</p> <p>6. CANoe Bus Name --> Please fill the CANoe Bus Name for the ARXML FileType.</p> <p>7. CANoe Channel --> Please fill the CANoe Channel for the ARXML FileType.</p>
		Power Supply on RS232	<p>1. Short Name --> <NodeName>.</p> <p>2. Device Name --> Please select the relevant PowerSupply Model from the dropdown list.</p> <p>Additional COM Configuration</p> <p>1. Port : Please select from the dropdown list.</p> <p>2. Bits Per Second : Please select from the dropdown list.</p> <p>3. Data Bits : Please select from the dropdown list.</p> <p>4. Parity : Please select from the dropdown list.</p> <p>5. Stop Bits : Please select from the dropdown list.</p> <p>6. In-buffer Size : Please select from the dropdown list.</p> <p>7. Out-buffer Size : Please select from the dropdown list.</p>
		Image Grabber	<p>1. Short Name --> <NodeName>.</p> <p>2. Device Name --> Please select the relevant PowerSupply Model from the dropdown list.</p> <p>3. Reference Images File Path --> Please select the folder of the Images.</p>
		Video Grabber	<p>1. Short Name --> <NodeName>.</p> <p>2. Device Name --> Please select the relevant PowerSupply Model from the dropdown list.</p> <p>3. Video Recorder Path --> Please select the folder of the Videos.</p>
		Environmental Simulation	<p>1. Short Name --> <NodeName>.</p> <p>2. Device Name --> Please select the relevant PowerSupply Model from the dropdown list.</p> <p>3. Reference Videos File Path --> Please select the folder of the Videos.</p> <p>4. Height --> Please fill the height of the screen that Video has to be played.</p> <p>5. Width --> Please fill the width of the screen that Video has to be played.</p> <p>6. X-Coordinate --> Please fill the X-Coordinate offset the screen that Video has to be played.</p> <p>7. Y-Coordinate --> Please fill the Y-Coordinate offset the screen that Video has to be played.</p>
	Remote Diagnostics	Control Desk	<p>1. Short Name --> <NodeName>.</p> <p>2. File Type --> Please select from the dropdown list.</p> <p>3. File Path --> Please select the file path based on the fileType.</p> <p>4. Vehicle Name --> Please select from the dropdown list.</p> <p>5. Logical Name --> Please select from the dropdown list.</p> <p>6. Base Variant Name --> Please select from the dropdown list.</p> <p>7. Protocol Name --> Please select from the dropdown list.</p>
		CANoe	<p>1. Short Name --> <NodeName>.</p> <p>2. File Type --> Please select from the dropdown list.</p> <p>3. File Path --> Please select the file path based on the fileType.</p> <p>4. Vehicle Name --> Please select from the dropdown list.</p> <p>5. Logical Name --> Please select from the dropdown list.</p> <p>6. Base Variant Name --> Please select from the dropdown list.</p> <p>7. Protocol Name --> Please select from the dropdown list.</p> <p>8. CANoe Project Path --> Please select the CANoe Configuration file.</p> <p>9. CANoe Bus Name --> Please fill the CANoe Bus Name.</p>
	EES Port Configuration	Control Desk	<p>1. Short Name --> <NodeName>.</p> <p>2. Vendor --> Please select from the dropdown list.</p> <p>3. Project Configuration File Path --> Please select the file for the EES Port Configuration.</p> <p>4. Port Configuration Name --> Please fill the EES Port Configuration Name.</p>

19.1.2 Sensor sim

If user want to start scene download in motion desk they need to check the sensor sim box



stCase ID	Category	Description	Script Name	Automation Status	Shared Id	Shared
1948	Heading	Pre&Post_Condition				
1950	System Test	LCA/I KA/I CW/I Precondition				
1948						
1984						
1952						
1953						
1954						
1955						
1956						
1957						
1958						
1959						
1960						
1961						
1962						
1963						
1964						
1965						
1966						
1967						
1968						
1969						
1970						
1971						
1972						
1973						
1974						
1975						
1976						
1977						
1978						
1979						
1980						
1981						
1982						
1983						
1984						
1985						
1986						
1987						
1988						
1989						
1990						
1991						
1992						
1993						
1994						
1995						
1996						
1997						
1998						
1999						
2000						
2001						
2002						
2003						
2004						
2005						
2006						
2007						
2008						
2009						
2010						
2011						
2012						
2013						
2014						
2015						
2016						
2017						
2018						
2019						
2020						
2021						
2022						
2023						
2024						
2025						
2026						
2027						
2028						
2029						
2030						
2031						
2032						
2033						
2034						
2035						
2036						
2037						
2038						
2039						
2040						
2041						
2042						
2043						
2044						
2045						
2046						
2047						
2048						
2049						
2050						
2051						
2052						
2053						
2054						
2055						
2056						
2057						
2058						
2059						
2060						
2061						
2062						
2063						
2064						
2065						
2066						
2067						
2068						
2069						
2070						
2071						
2072						
2073						
2074						
2075						
2076						
2077						
2078						
2079						
2080						
2081						
2082						
2083						
2084						
2085						
2086						
2087						
2088						
2089						
2090						
2091						
2092						
2093						
2094						
2095						
2096						
2097						
2098						
2099						
2100						
2101						
2102						
2103						
2104						
2105						
2106						
2107						
2108						
2109						
2110						
2111						
2112						
2113						
2114						
2115						
2116						
2117						
2118						
2119						
2120						
2121						
2122						
2123						
2124						
2125						
2126						
2127						
2128						
2129						
2130						
2131						
2132						
2133						
2134						
2135						
2136						
2137						
2138						
2139						
2140						
2141						
2142						
2143						
2144						
2145						
2146						
2147						
2148						
2149						
2150						
2151						
2152						
2153						
2154						
2155						
2156						
2157						
2158						
2159						
2160						
2161						
2162						
2163						
2164						
2165						
2166						
2167						
2168						
2169						
2170						
2171						
2172						
2173						
2174						
2175						
2176						
2177						
2178						
2179						
2180						
2181						
2182						
2183						
2184						
2185						
2186						
2187						
2188						
2189						
2190						
2191						
2192						
2193						
2194						
2195						
2196						
2197						
2198						
2199						
2200						
2201						
2202						
2203						
2204						
2205						
2206						
2207						
2208						
2209						
2210						
2211						
2212						
2213						
2214						
2215						
2216						
2217						
2218						
2219						
2220						
2221						
2222						
2223						
2224						
2225						
2226						
2227						
2228						
2229						
2230						
2231						
2232						
2233	</					

ModelAccess

MODEL ACCESS DIALOG

***Model Desk Project Path** : C:\Supriya\Desktop\MOD_Traffic\MOD_Traffic.CDP

***Motion Desk Project Path** : C:\opMotionDesk_Traffic\MotionDesk_Traffic.CDP

***Motion Desk Experiment name** : ASMTraffic

***Model Desk Experiment Name** : Experiment_001

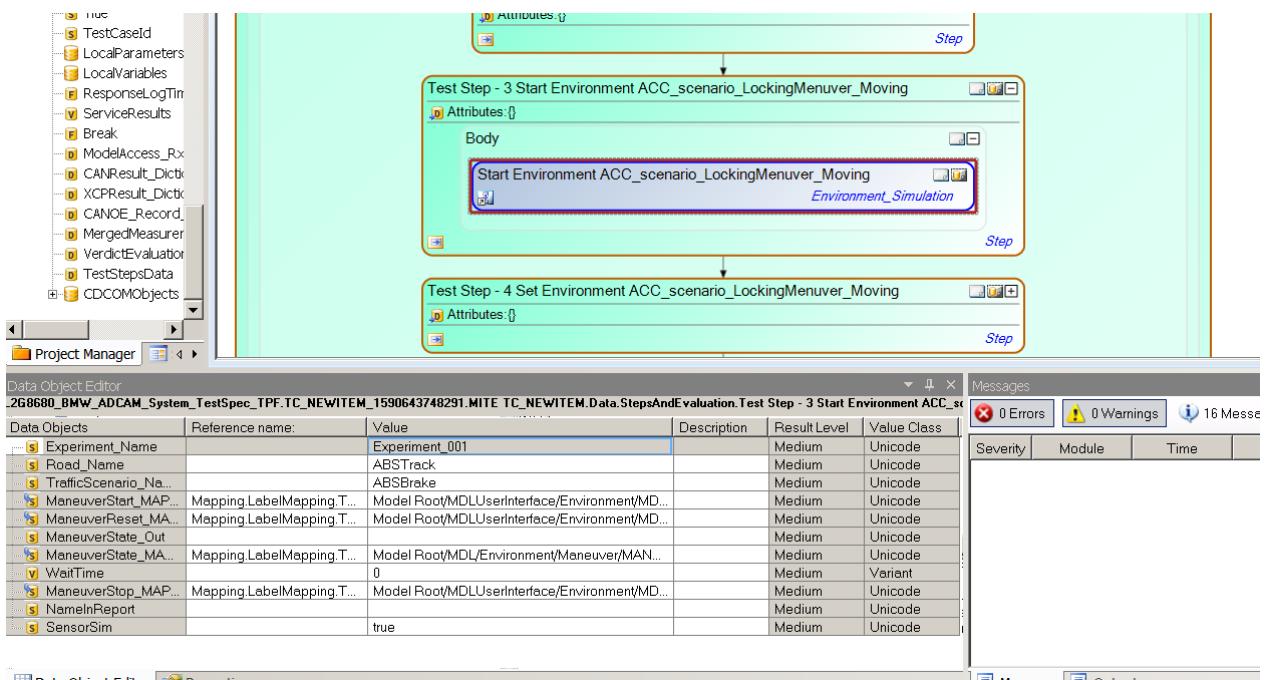
Sensor SIM

BACK **NEXT** **OK** **CANCEL**

Labels Messages Comments Traceability Recorder Diagnostics

The screenshot shows the ModelAccess application interface. At the top is the 'Model Access Dialog' window with fields for project paths and experiment names. Below it is the main workspace showing a 'Test Step - 3 Start Environment ACC_scenario_LockingMenuever_Moving' step. To the left is the 'Project Manager' tree view, and at the bottom is the 'Data Object Editor' table. A 'Messages' panel is visible on the right.

Data Objects	Reference name:	Value	Description	Result Level	Value Class
\$Experiment_Name	Experiment_001	ABSTrack		Medium	Unicode
\$Road_Name		ABSBrake		Medium	Unicode
\$TrafficScenario_Na...		Model Root/MDLUserInterface/Environment/MD...		Medium	Unicode
\$ManeuverStart_MAP...	Mapping_LabelMapping_T...	Model Root/MDLUserInterface/Environment/MD...		Medium	Unicode
\$ManeuverReset_MA...	Mapping_LabelMapping_T...	Model Root/MDLUserInterface/Environment/MD...		Medium	Unicode
\$ManeuverState_Out...		Model Root/MDLUserInterface/Environment/MD...		Medium	Unicode
\$ManeuverState_MA...	Mapping_LabelMapping_T...	Model Root/MDLUserInterface/Environment/MD...		Medium	Unicode
\$WaitTime	0	0		Medium	Variant
\$ManeuverStop_MAP...	Mapping_LabelMapping_T...	Model Root/MDLUserInterface/Environment/MD...		Medium	Unicode
\$NameInReport				Medium	Unicode
\$SensorSim	false			Medium	Unicode



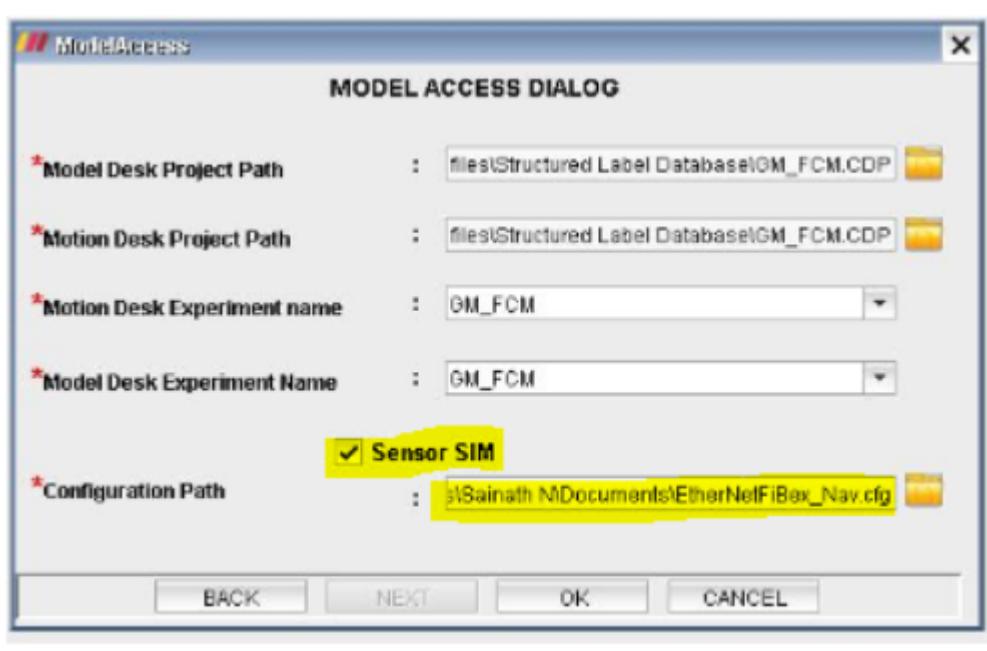
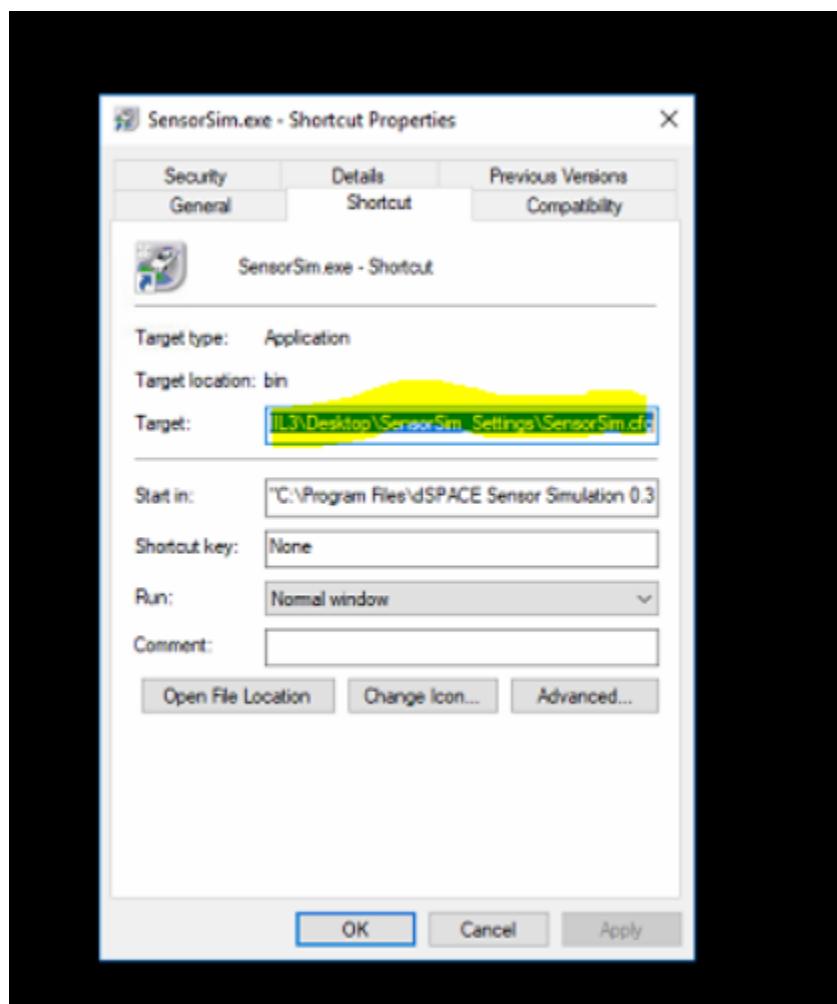
SENSOR SIM :

1.PREREQUISITES :

User need add sensor sim exe path to environmental variables with variable name as **SensorSim**.

And value will path of sensor sim exe.

In target you will get sensor sim exe path and cfg path you need to fill cfg path in control desk /motion desk/model desk configuration after making sensor sim option enable.



If sensor sim is enable it will initialize sensor sim and help to download scene and close sensor sim.

Created with the Personal Edition of HelpNDoc: [Free help authoring environment](#)

19.1.3 Innoviz XMB Reprocessing Test Bench Configuration

Step 1:-

User should select “**Test Script**” option from the list.

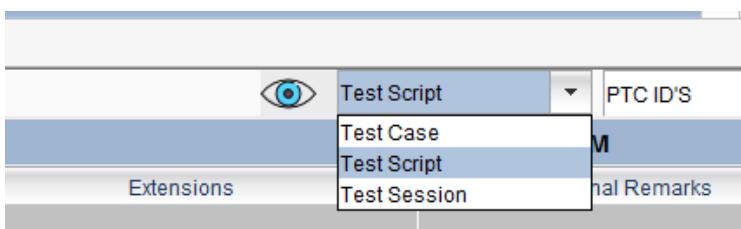
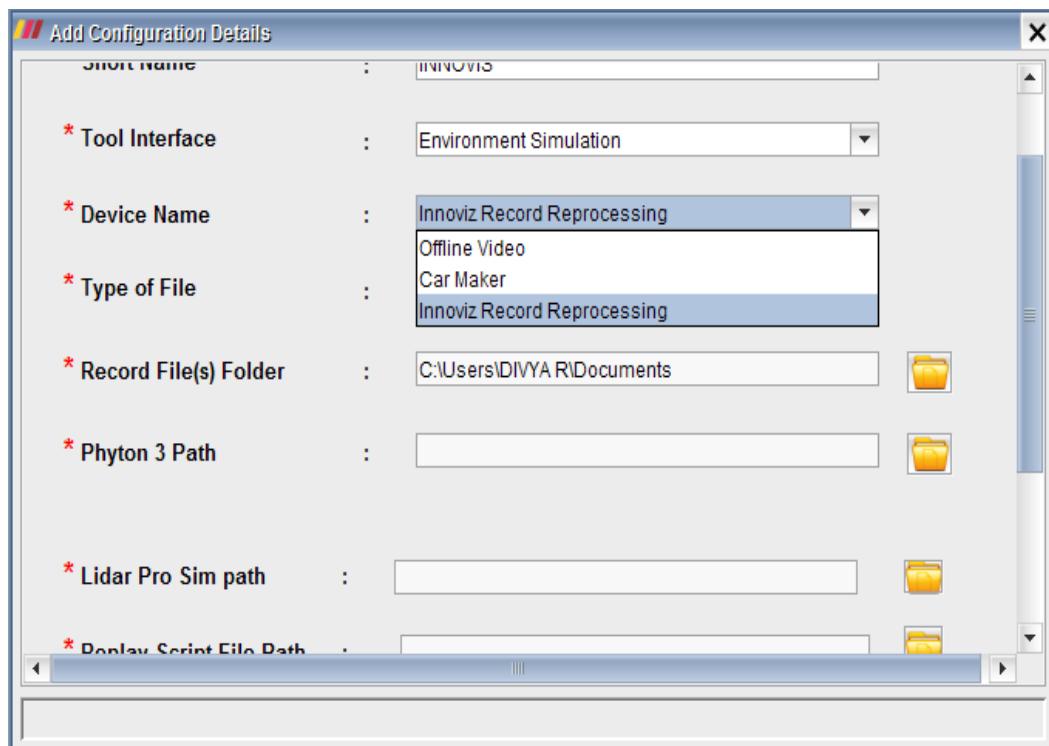


Figure: Select Test Script View Set

Step 2:-

User has to select add configuration details and from there user has to select Remote Access and environment Simulation as a Tool Interface.

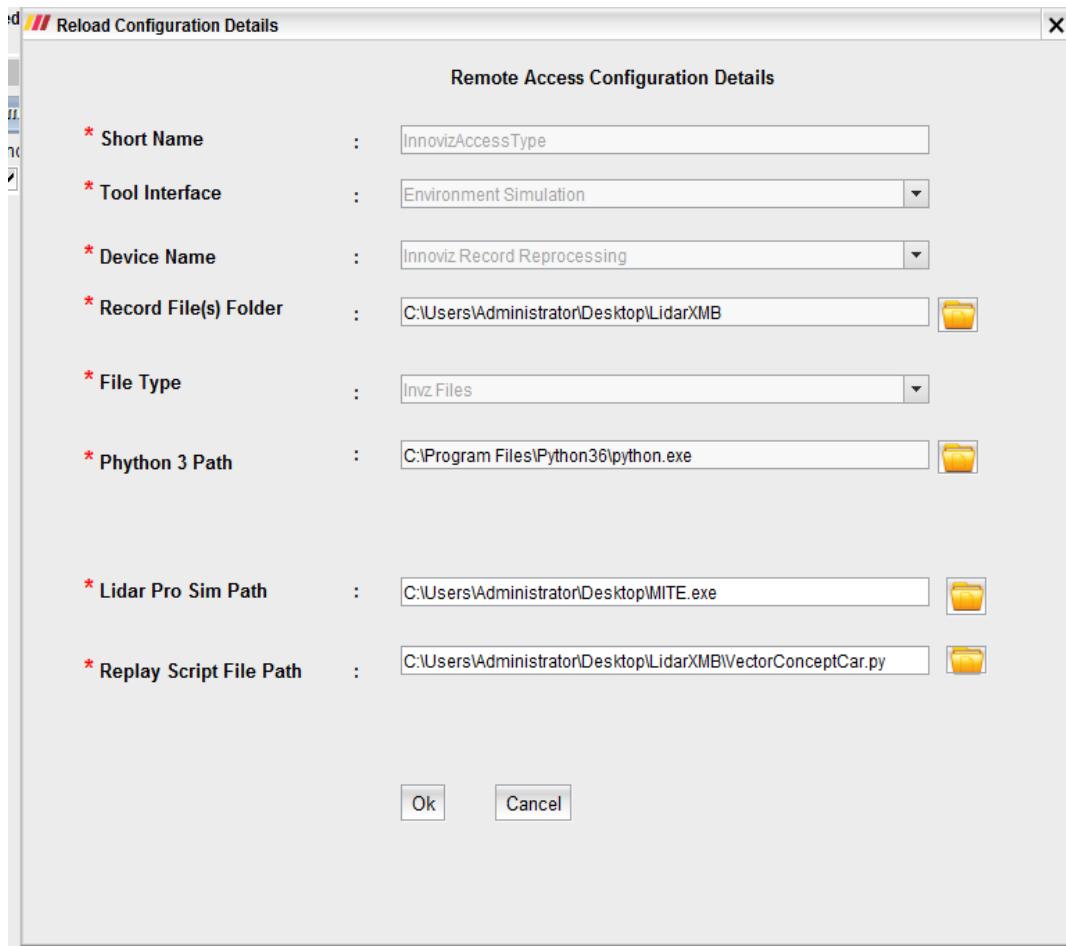


After Filling Required Fields user get the following popup:

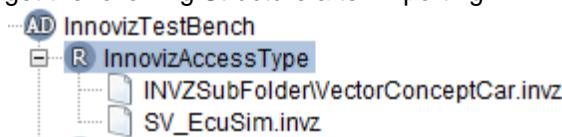
For Python 3 path:-.exe

For Lidar Pro Sim path:-.exe

For Replay Script File Path:-.py

**Step 3:-**

User will get the following Structure after importing.



Created with the Personal Edition of HelpNDoc: [Easily create CHM Help documents](#)

19.1.4 Magna Image Grabber

Step 1:-

User should select “**Test Script**” option from the list.

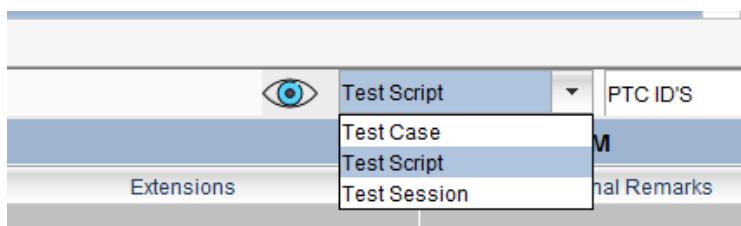
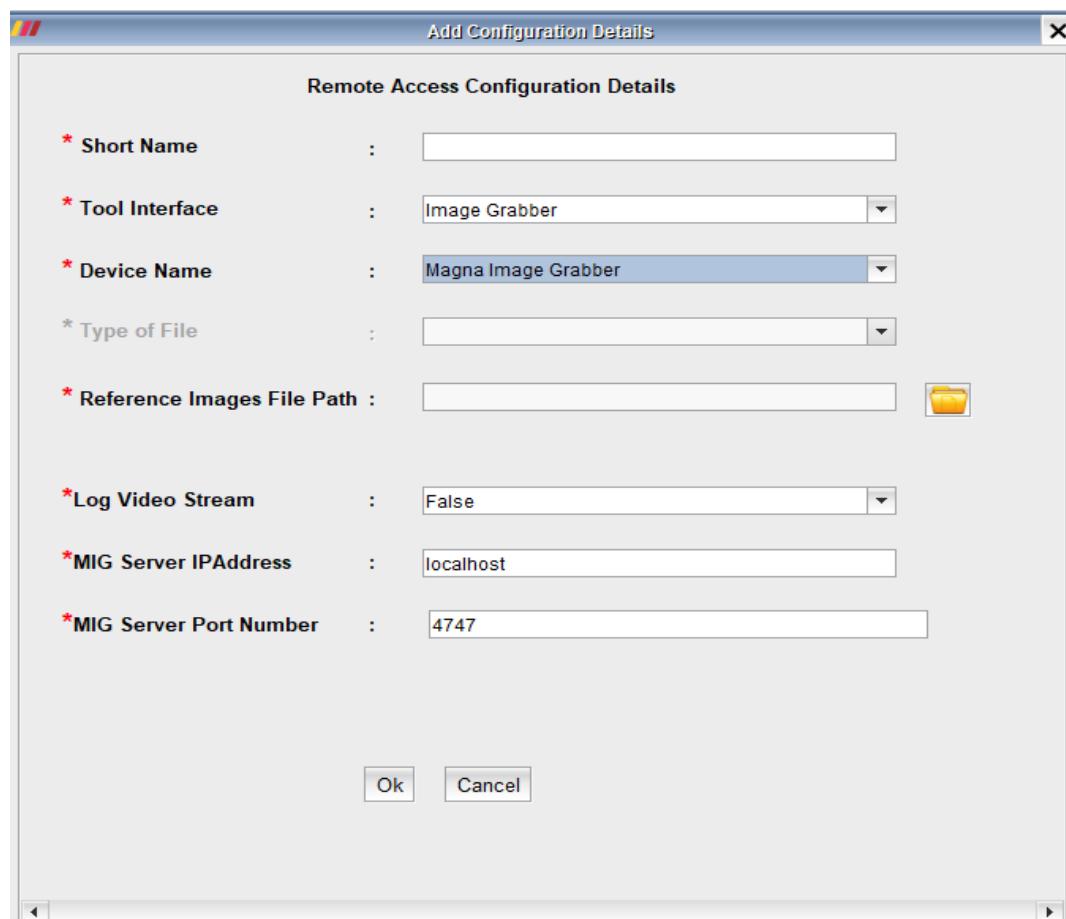
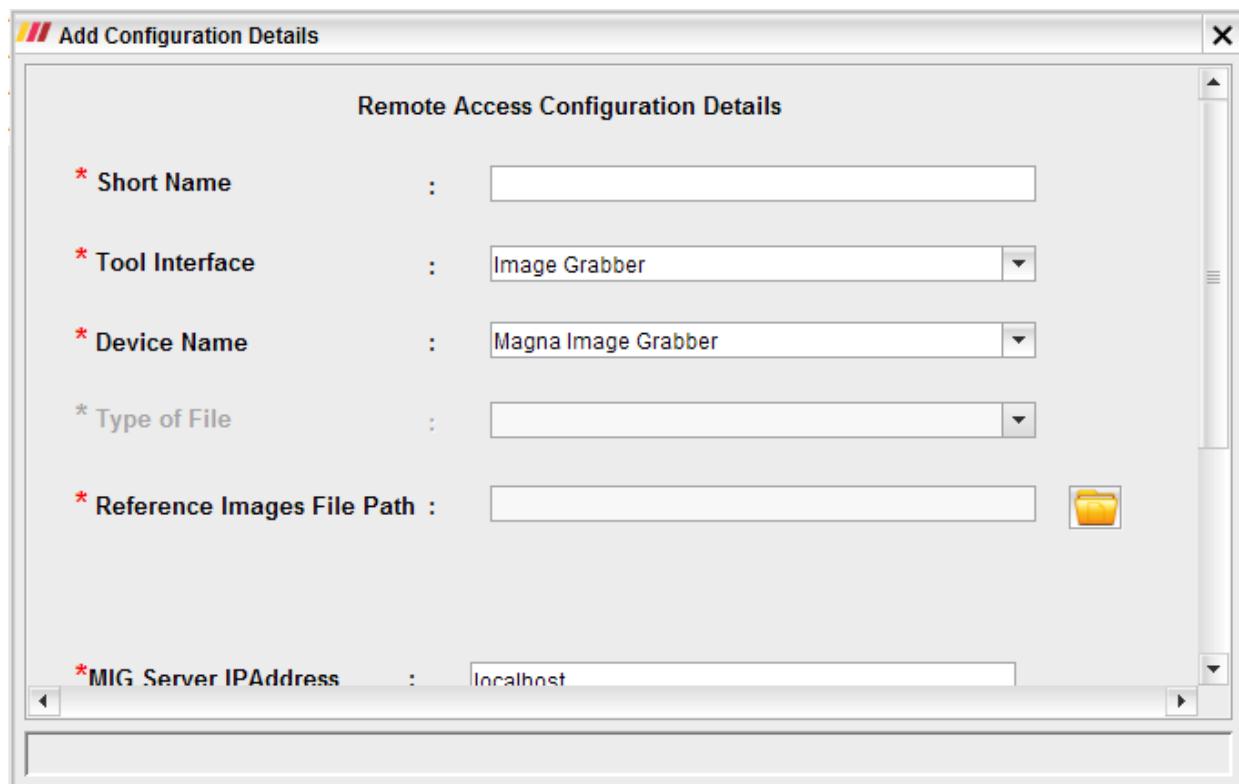


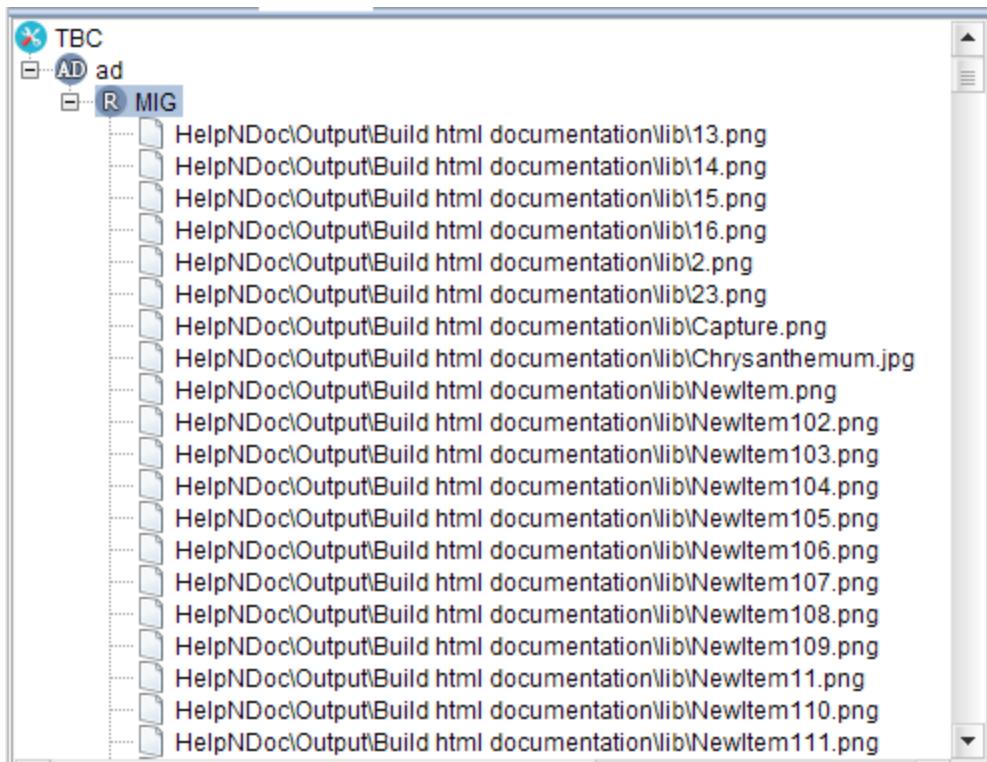
Figure: Select Test Script View Set

Step 2:-

User has to select add configuration details and from there user has to select Remote Access and Image grabber as a Tool Interface and Device name is Magna Image Grabber.

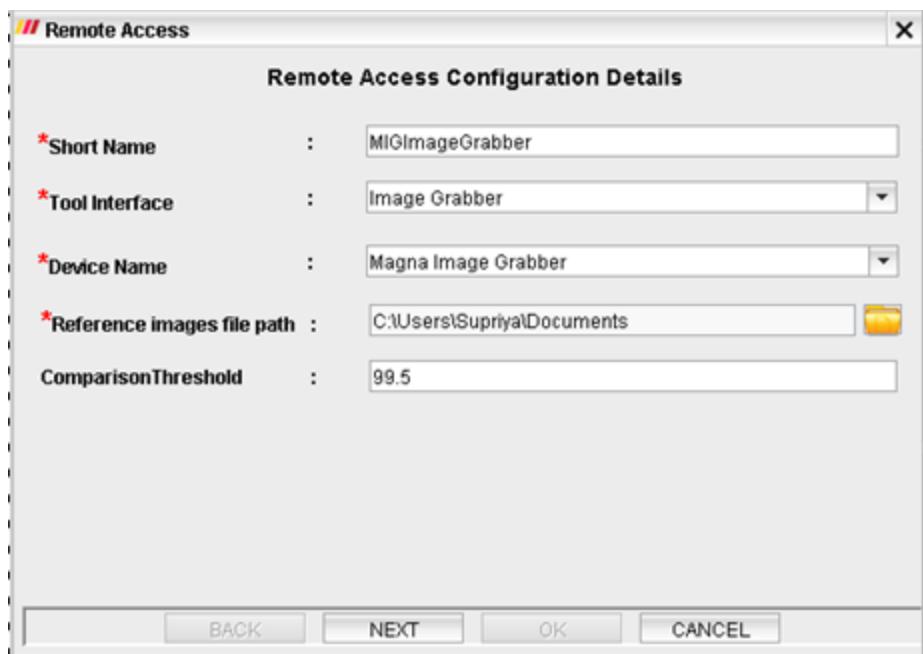
**Step 3:-**

After filling all the requirements user will get the Following Structure.

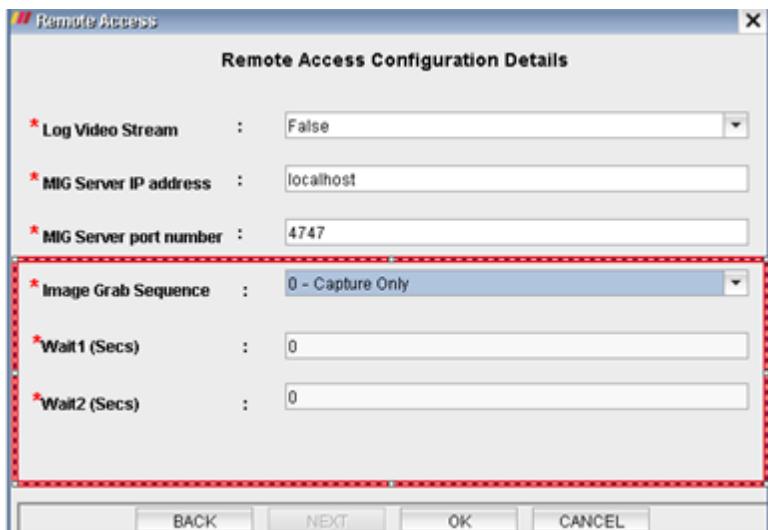


MIG - GRABBER

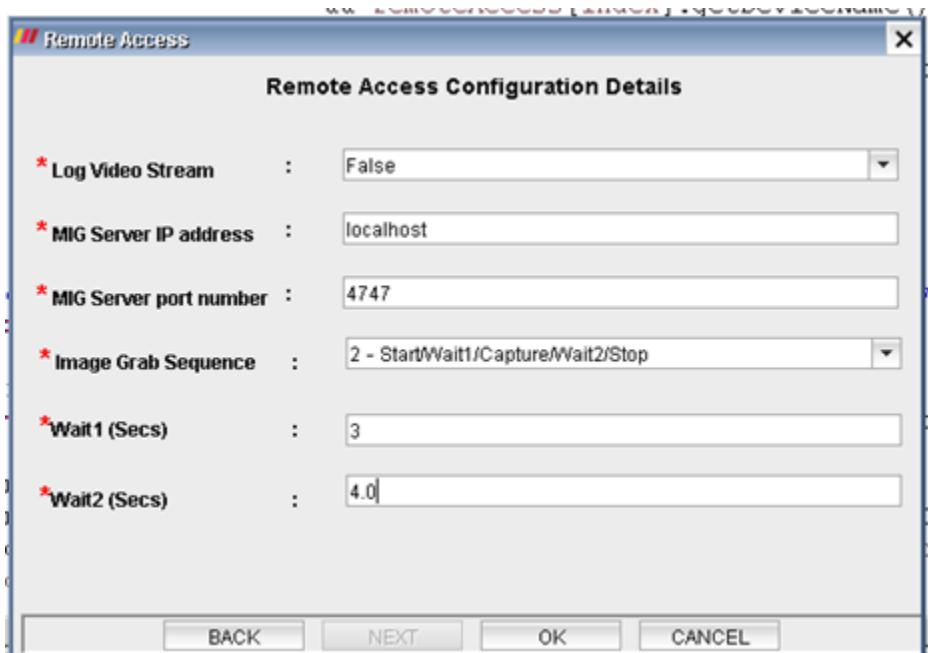
User will be able to choose capture sequence with three choices which will be provided in TBC. This will affect the way image is captured during testing. User should choose the option accordingly. When User will use Remote Access configuration with tool interface as Image Grabber and device name as Magna image grabber the following pop up will be seen:-



Here User can able to view new field as :Image Grab Sequence ,Wait1(secs),Wait2(secs)



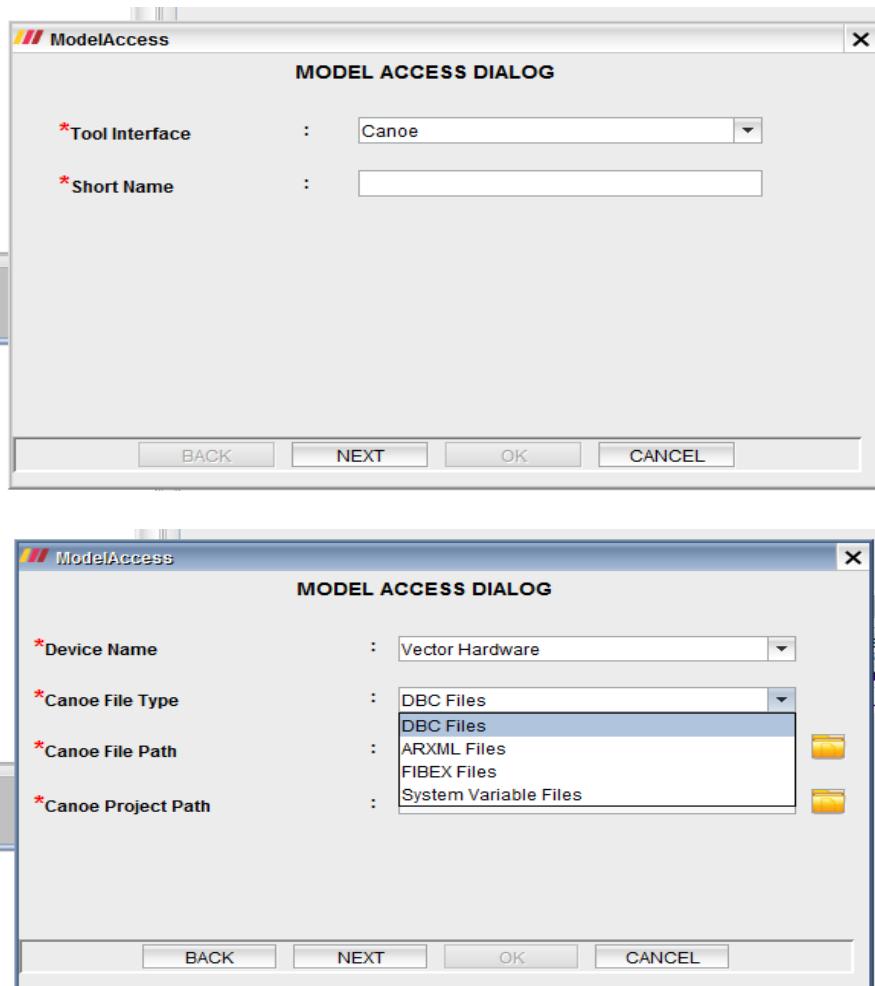
When Image Grab Sequence is 2- Start/Wait1/Capture/Wait2/Stop



Created with the Personal Edition of HelpNDoc: [Free PDF documentation generator](#)

19.1.5 CANoe XIL Interface

From MITE Version, MITE v2.2.0.0 it is possible to record the data at the required rate using the Canoe XIL integration. The Canoe XIL can be configured at TBC level in the Model Access Dialog. To configure the Canoe XIL, the tool interface “Canoe” has to be selected and user can import any XIL related file from the shown option List.



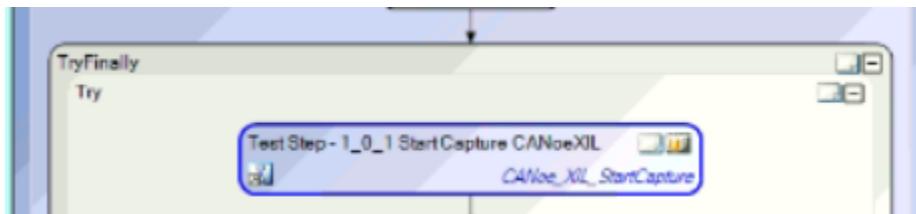
After configuring the TBC with the required Canoe XIL file, user has to do the Label Mapping in the Label Mapping Frame.

While doing the Label Mapping, to avoid the confusion to the user between the Canoe Signals from Remote Access and the Canoe Signals from the Model Access, the prefix "XIL:" is shown to the Canoe Paths that are derived from Model Access Canoe Files.

Test Labels		ECU IO Labels	Diagnostic Labels	
#	IO Labels	Read Path		Access Type
1	EngineSpeed	NoNameSpace_Int		Model Access
2	HeadLight	XIL:\FCnt		Model Access
3	NS_StructMem	XIL:\NoNameSpace_Float		Model Access
4	NS_StructMem_Check	XIL:\NoNameSpace_Int		Model Access
5	NS_Sysvar	XIL:\TestNS::StructInstance.Float		Model Access
6	NS_Sysvar_Check	XIL:\TestNS::StructInstance.IntMe		Model Access
7	NoNameSpaceSysvar	XIL:\TestNS::TestFloat		Model Access
8	NoNameSpaceSysvar_...	XIL:\TestNS::TestInt		Model Access
9	enwar	FCnt		Model Access

Once the user save the data and generate the scripts, FDX file will be generated for all the Canoe XIL related Read Paths and Write Paths. This FDX file will be save inside the TBC in the below shown location:
 FDX File Location: "C:\MITE\MITE Project\<Project>\<TestSuite>\AutomationProject\<TBC>\TestScripts\AutomationDesk Project\Reference Files\<TBC Data.xml>"

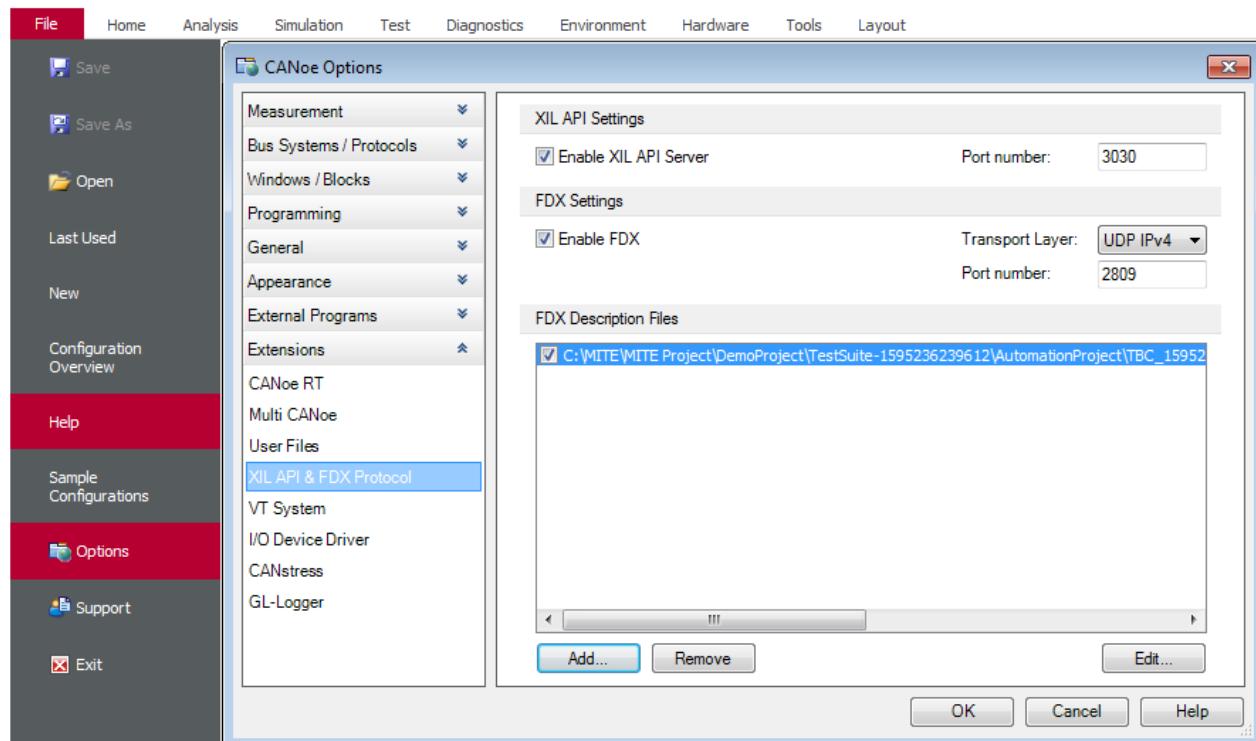
After the Test Script Generation, the corresponding XIL related attributes will be filled in the Automation Desk.



The list of signals recorded will be shown as below:

Item	Type	Value
0	List	['NoNameSpace_Int', 'TestNS::TestInt', 'TestNS::StructInstance.FloatMem']
0	str	'NoNameSpace_Int'
1	str	'TestNS::TestInt'
2	str	'TestNS::StructInstance.FloatMem'

To configure the generated FDX file in the Canoe, Go to File → Options → Extensions → Select “XIL API & FDX Protocol” → Click on Add and give the FDX path and click on OK.



XIL API :-

```
CANoe APPLICATION INITIALIZATION - Start
Exception Occured during opening CANoe configuration : com_error: 0x80004005: Unspecified error
Retrying to open configuration...
Exception Occured during opening CANoe configuration : com_error: 0x80004005: Unspecified error
Retrying to open configuration...
Exception Occured during opening CANoe configuration : com_error: 0x80004005: Unspecified error
Couldn't Load configuration, Check if the Configuration needs to be saved.
\TC_NEWITEM_1595859893358_CANoeLog_20200728_175129.asc
CANoe Application Initialization - End

Execution failed!
Exception: Incorrect FDX Configuration
Exception in <user input of Main Library>.Exec "ValidateFDXFilePath" in line 41: 'raise Exception("Incorrect FDX Configuration")'
Exception Traceback (most recent block last):
  in project element "Toyota_Canoe_XIL_Testing.TC_NEWITEM_1595859893358.For.MITE_TC_NEWITEM.Data.Initialization.CANoe_XIL_Init"
    → automationdesk:Toyota_Canoe%20XIL%20Testing.adl#TC_NEWITEM_1595859893358.For.MITE%20TC_NEWITEM.Data.Initialization.CANoe_XIL
  in linked library "Mite_Library" in element "CANoe_XIL_Init.ValidateFDXFilePath"
    → automationdesk:Mite_Library.adl#CANoe_XIL_Init.ValidateFDXFilePath
  in command of "ValidateFDXFilePath"
    → automationdesk:Mite_Library.adl?Line=41;Show=Command#CANoe_XIL_Init.ValidateFDXFilePath
```

*) FDX file was not configured in Canoe.

When this error occur follow below steps:-

1. Open Canoe
2. Click on File
3. Select Options
4. In extensions tab Select XIL API & FDX Protocol
5. Load the FDX file created by MITE from the following location

Ex:- Reference Files\FDXData_TBC_xxxxxxxxxxxxxxx.xml

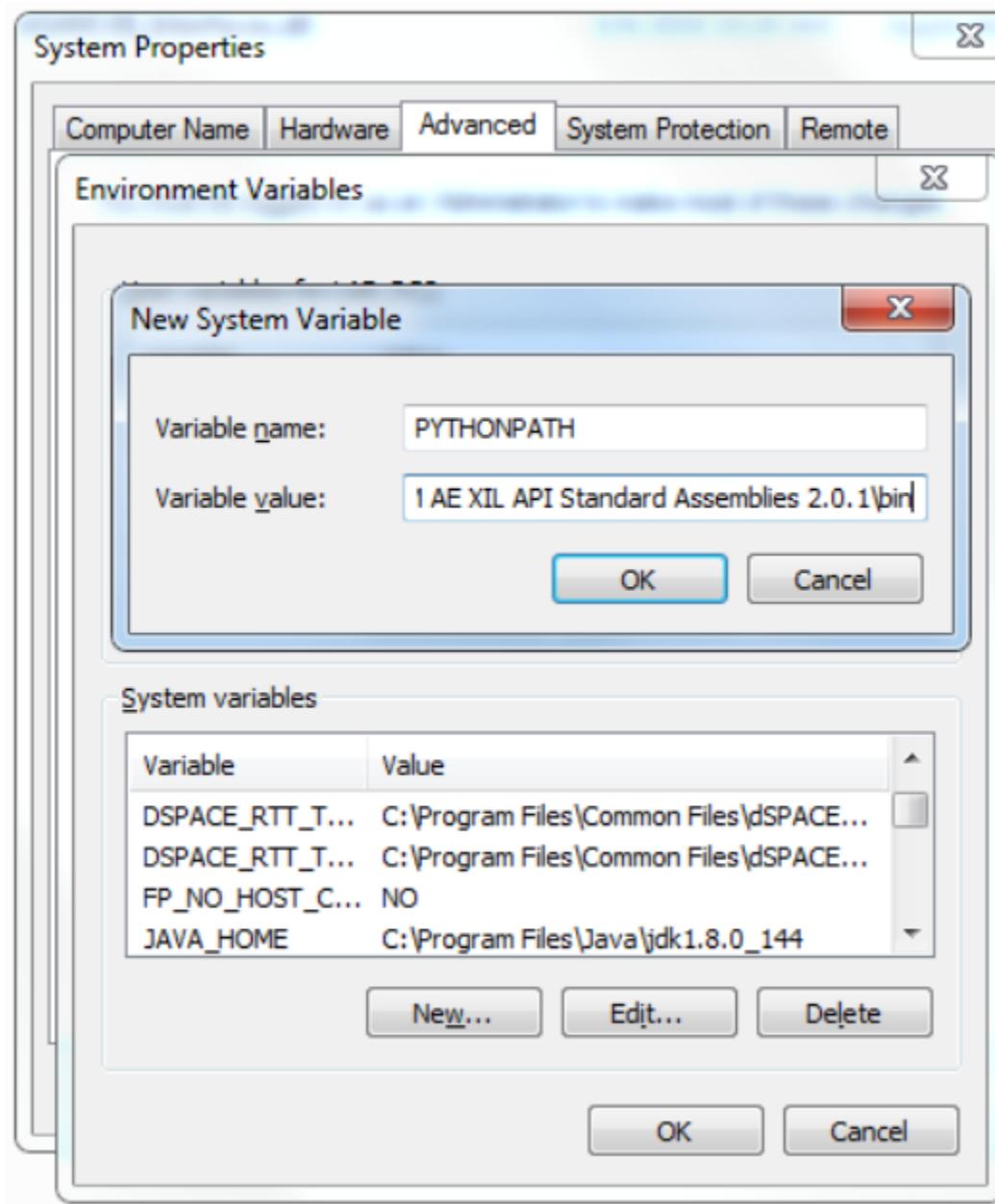
```
1595944578.77
CANoe Application Initialization - Start
Exception Occured during opening CANoe configuration : com_error: 0x80004005: Unspecified error
Retrying to open configuration...
Exception Occured during opening CANoe configuration : com_error: 0x80004005: Unspecified error
Retrying to open configuration...
Exception Occured during opening CANoe configuration : com_error: 0x80004005: Unspecified error
Couldn't Load configuration, Check if the Configuration needs to be saved.
C:\Users\LAB_PC2\Documents\TBC_1595931866005\TestScripts\AutomationDesk Project\Toyota_Canoe XIL Testing\{6B754E7C-F1DC-41BC-B939-CANoe Application Initialization - End
Error during loading of the requested Testbench library occurred. [An assembly of Vector (CANoe64), version 2.0.1 could not be found at ASAM.XIL.Implementation.TestbenchFactory.Testbench.TestbenchFactory.CreateVendorSpecificTestbench(String vendorName, String
Error during loading of the requested Testbench library occurred. [An assembly of Vector (CANoe32), version 2.0.1 could not be found at ASAM.XIL.Implementation.TestbenchFactory.Testbench.TestbenchFactory.CreateVendorSpecificTestbench(String vendorName, String
Error during loading of the requested Testbench library occurred. [An assembly of Vector (CANoe64), version 2.1.0 could not be found at ASAM.XIL.Implementation.TestbenchFactory.Testbench.TestbenchFactory.CreateVendorSpecificTestbench(String vendorName, String
Error during loading of the requested Testbench library occurred. [An assembly of Vector (CANoe32), version 2.1.0 could not be found at ASAM.XIL.Implementation.TestbenchFactory.Testbench.TestbenchFactory.CreateVendorSpecificTestbench(String vendorName, String
Unable to create testbench

Exception: Unable to create testbench
in element "Toyota_Canoe XIL Testing.TC_NEWITEM_1595859893358.For.MITE_TC_NEWITEM.Data.Initialization.CANoe_XIL_Init.@Mite_Library"
  → automationdesk:///C:/Users/LAB_PC2/Documents/TBC_1595931866005/TestScripts/AutomationDesk%20Project/Toyota_Canoe%20XIL%20Te
```

*) In case of above error please follow below procedure

1. We need to config the ASAM XIL API dll files location in Environmental variables, Dll file location path given below

C:\Program Files (x86)\ASAM e.\ASAM AE XIL API Standard Assemblies 2.0.1\bin



2. After configuring the file path in System variables path need to restart the Automation Desk.
3. Still in case of same error install Vector CANoe XIL API.exe mentioned in the below location.
C:\Program Files\Vector CANoe 11.0\Installer Additional Components\XILAPI\Vector CANoe XIL API.exe

Created with the Personal Edition of HelpNDoc: [Full-featured Help generator](#)

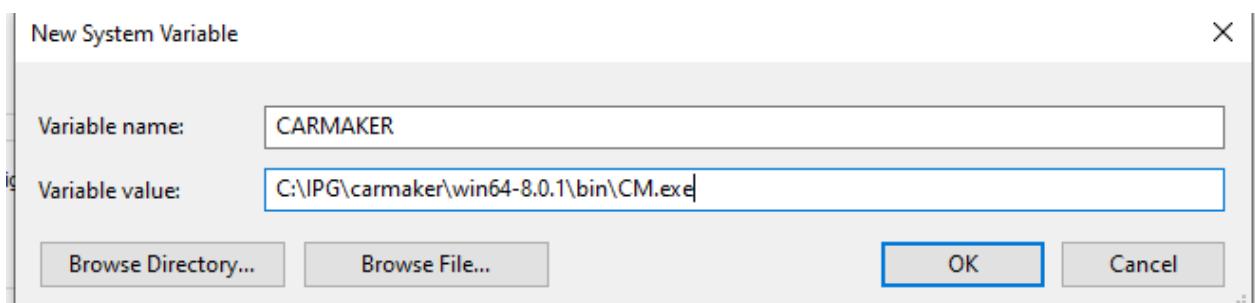
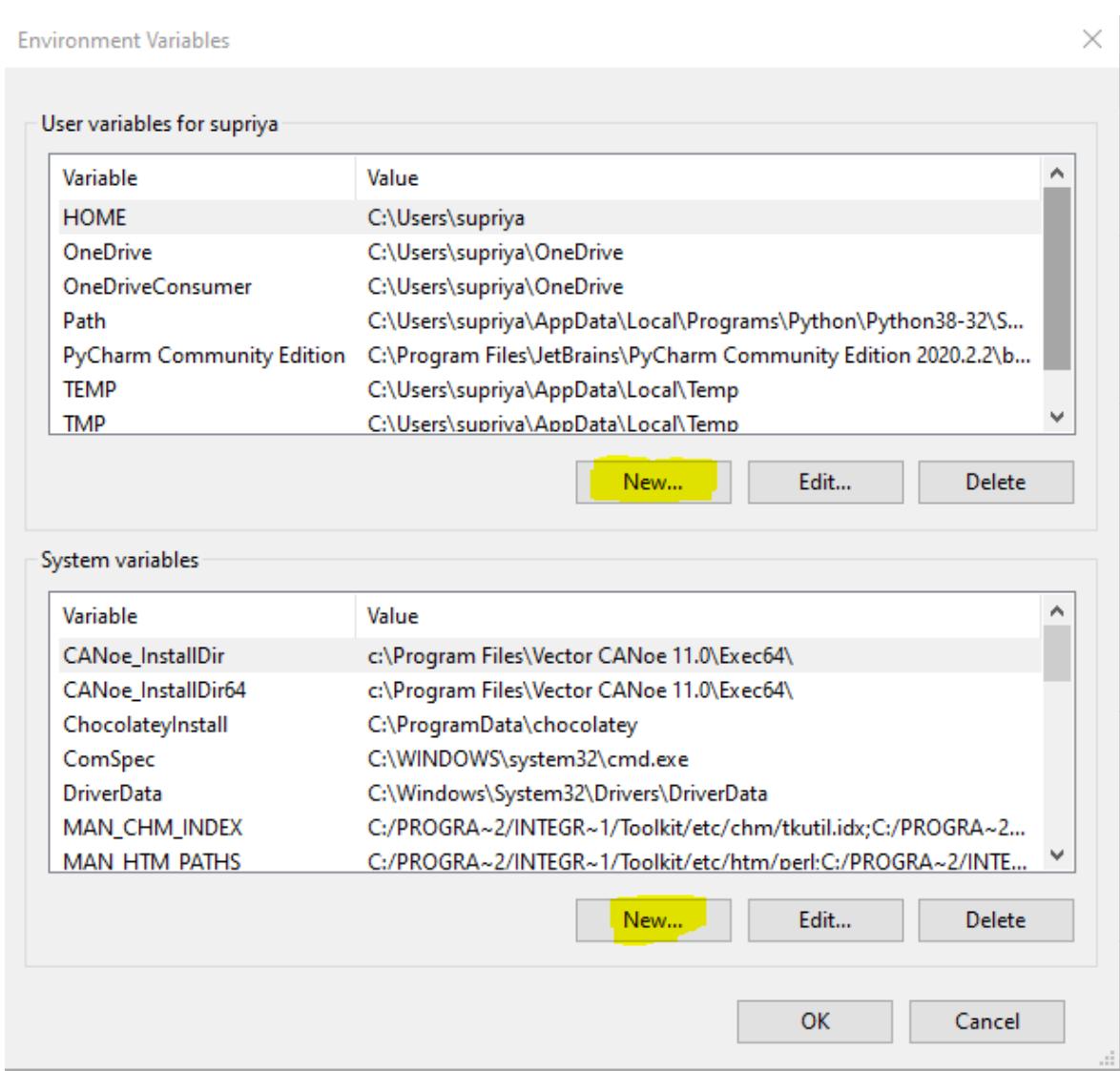
19.1.6 CAR MAKER

CAR MAKER

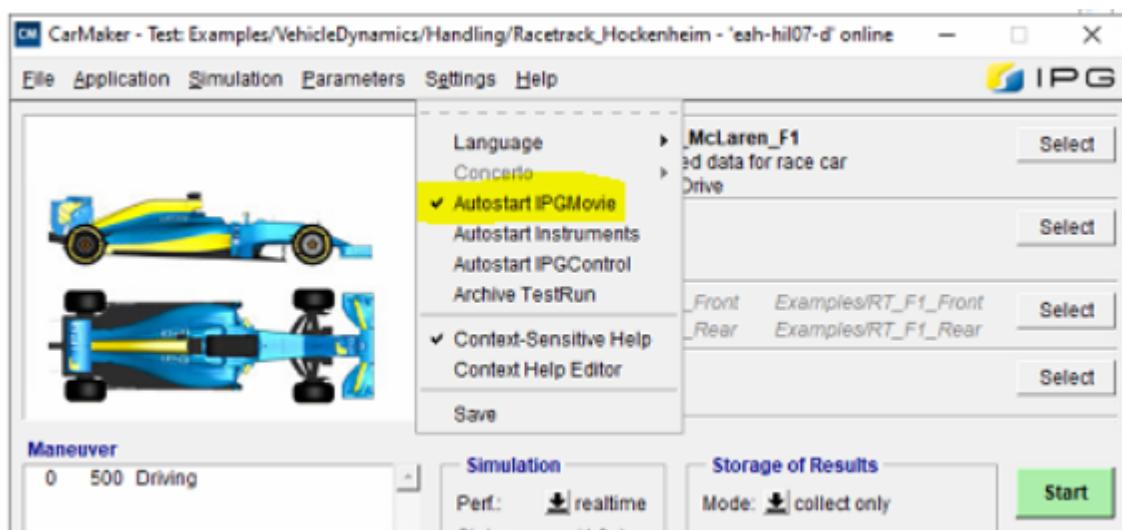
1.PREREQUISITES :

User need add car maker path to environmental variables.

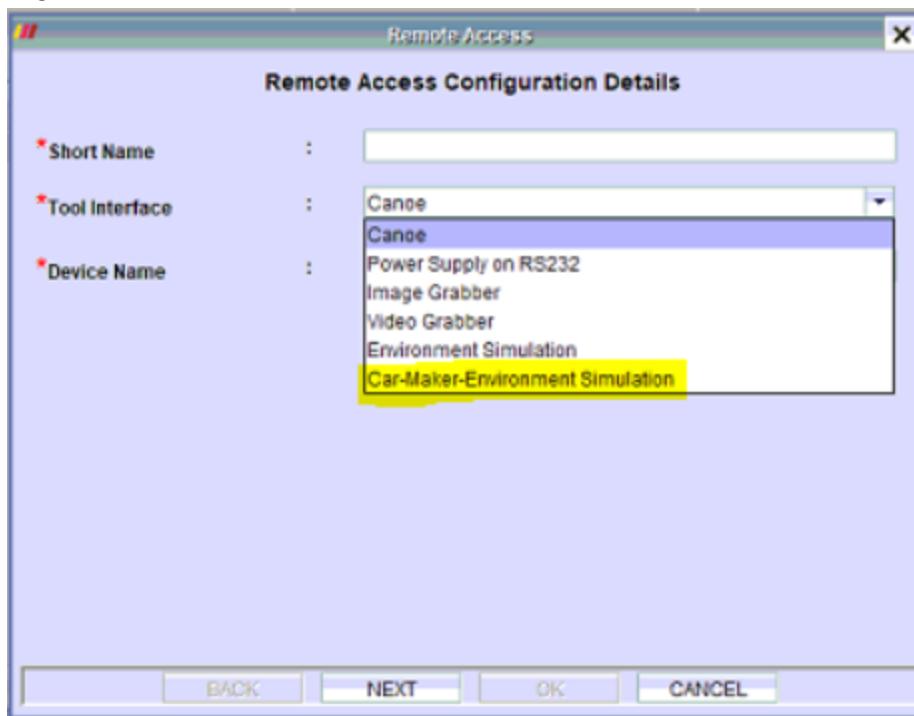
Environment Variables

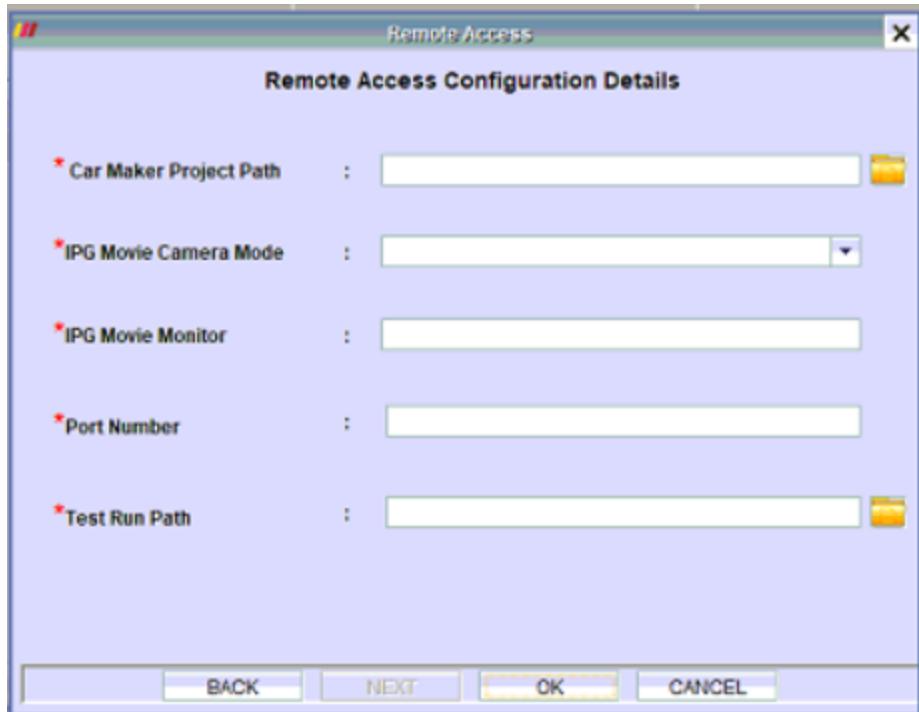


- IPGMovie should be auto start mode.
Open carmakerGUI → Settings → enable AutoStart IPGMovie → close Carmaker Gui /IPGMovie(if open)

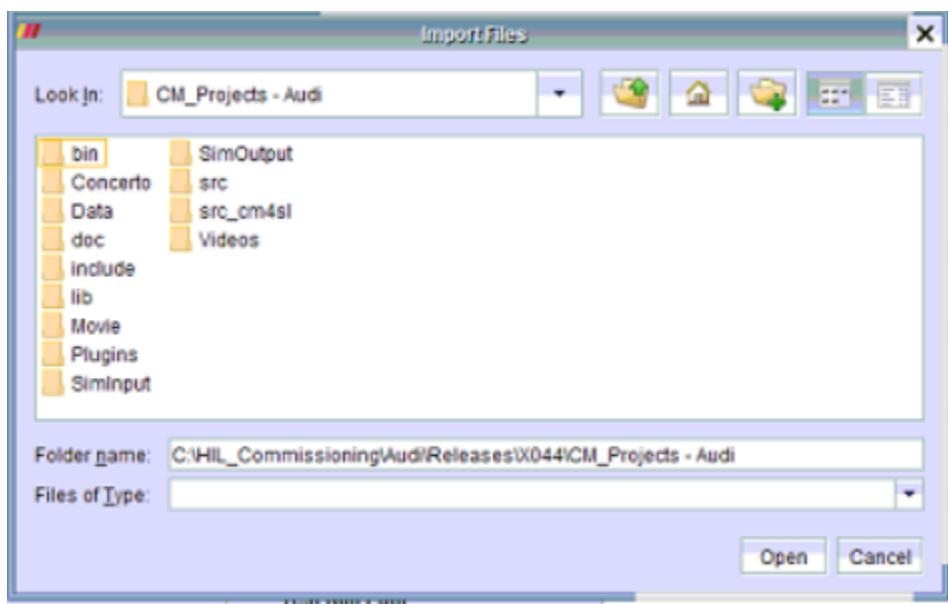


Action : Start,Stop,Run
 Service Type : Environment,Video
 TBC :

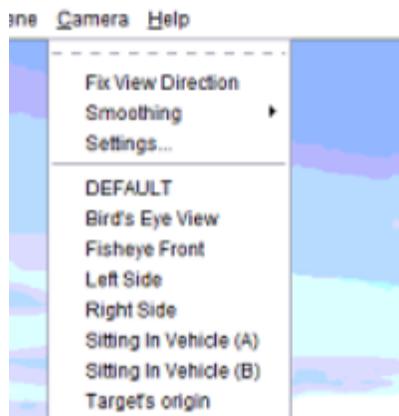




1.Car Maker Project Path : It should be upto project path inside which bin concerto all will reside.



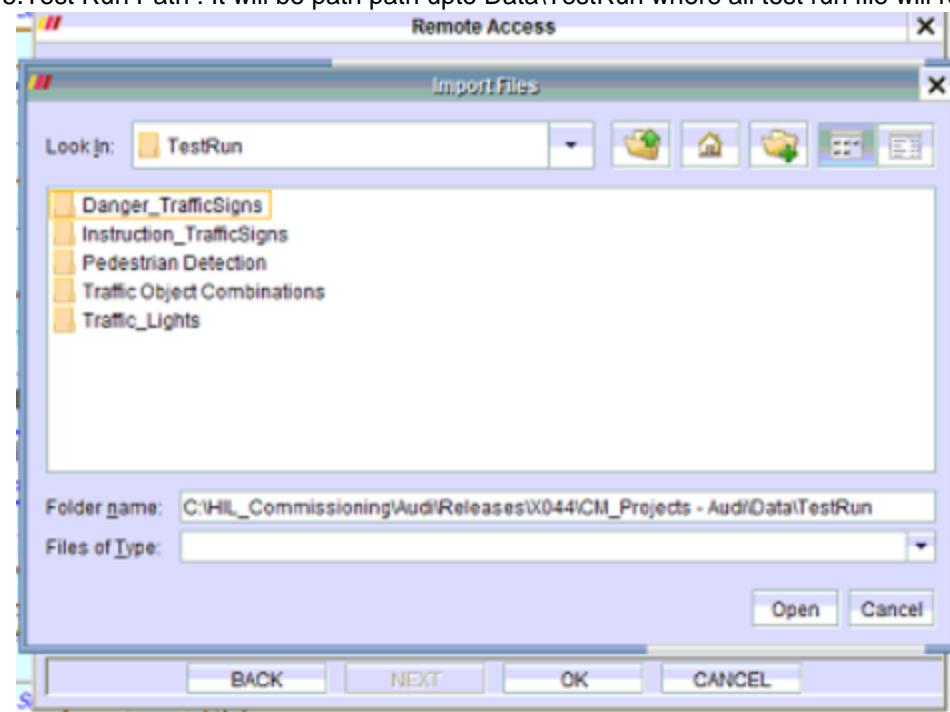
2. IPG Movie Camera Mode : Which camera mode of IPG Movie required .



3.Movie Monitor : In which monitor user want simulation to run e.g 3

4 Port Number : It will be auto read. If during execution exception like port number already in use User can change port number

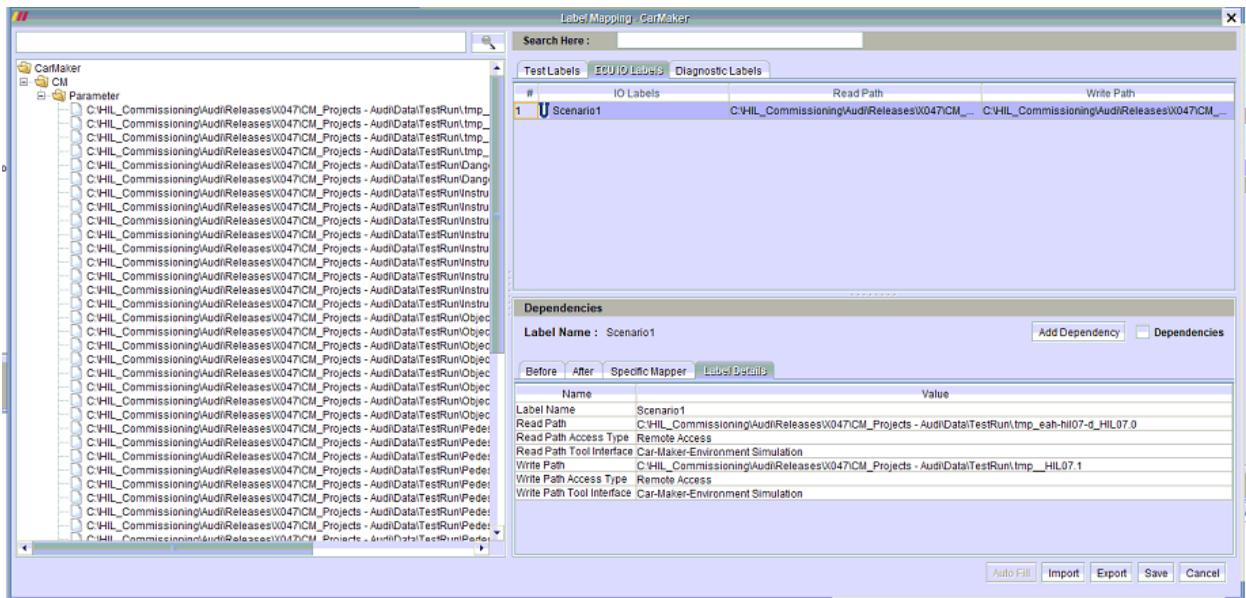
5.Test Run Path : It will be path upto Data\TestRun where all test run file will reside.



6.LABEL MAPPING :

User will get all the testrun file path in label mapping frame .

#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value
1	Start	Video	Scenario1	5
2	Wait	Time		5

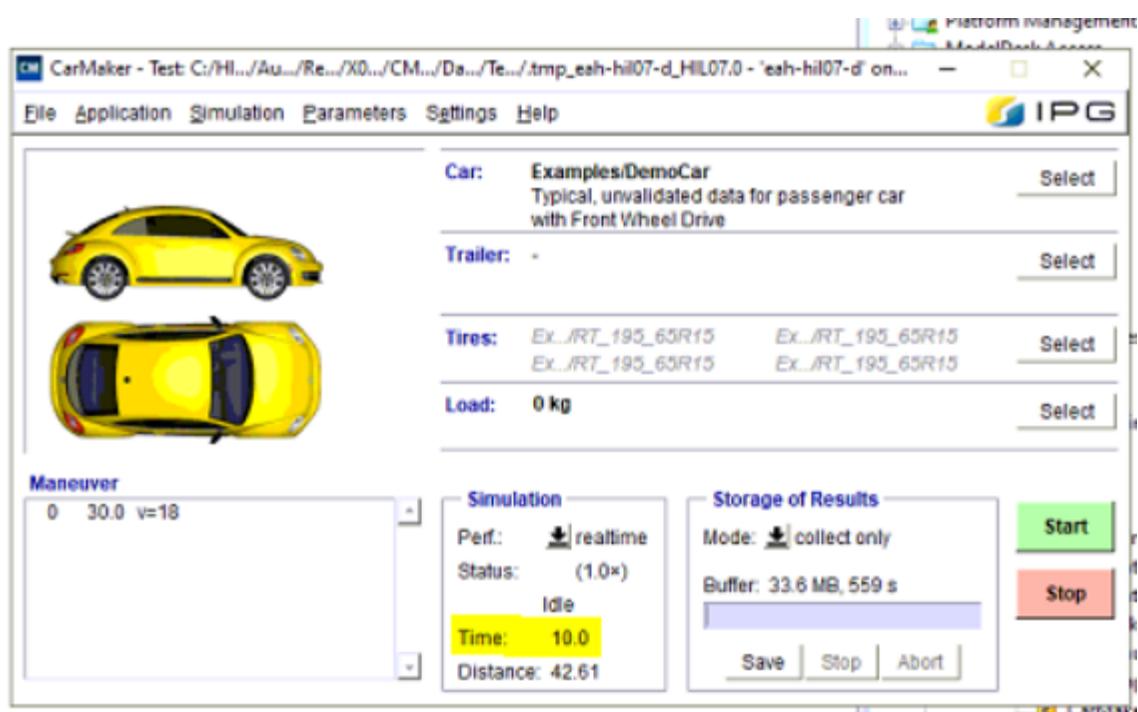


7. Start Video/Start Environment : It will load the testrun which user will map in label mapping frame and it will start the simulation.

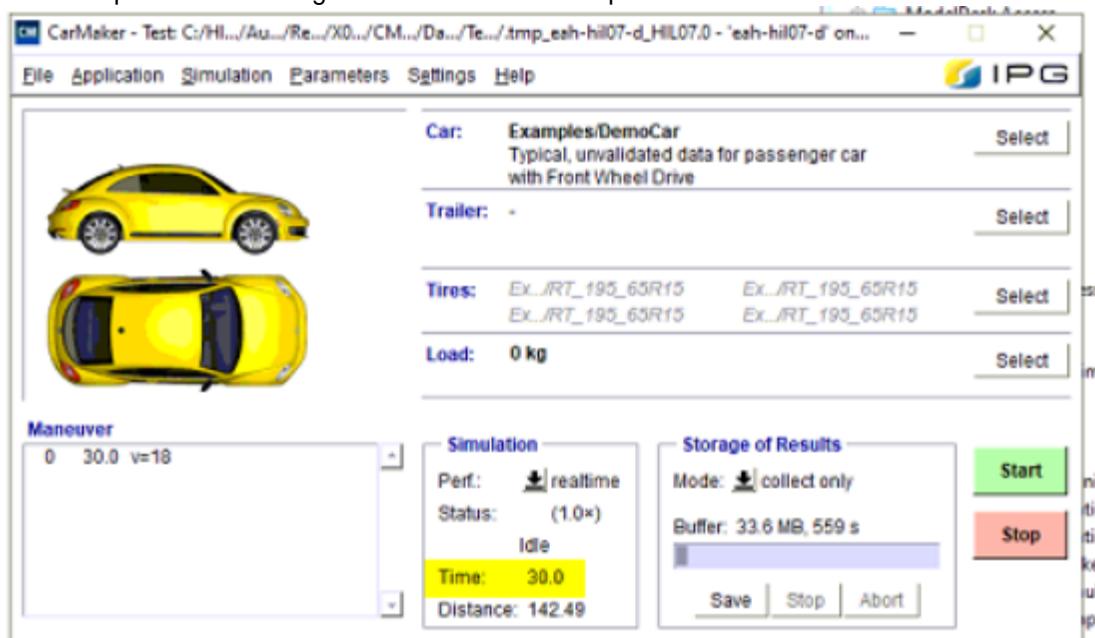
- Stop Video/Stop Environment : It will stop the simulation running in car maker.
- Run Video /Run Environment: It will load the simulation and start simulation .

#	Test Sequence					
1	Run	Video	Scenario1	10		
2	Wait	Time		5		
3	Run	Video	Scenario1			
#	Post Conditions					
6	Wait	Time		5		
7	Stop	Environment	Scenario1			
#	Test Sequence					
1	Run	Video	Scenario1		for 100secs	
2	Wait	Time		5		
3	Run	Video	Scenario1			
4	Wait	Time		5		
5	Run	Video	Scenario2	30		
#	Post Conditions					

If desired expected value/extension for duration is given the simulation will stop at corresponding time.
NOTE : simulation stop time in for run test case should be less than or equal to entire simulation time.

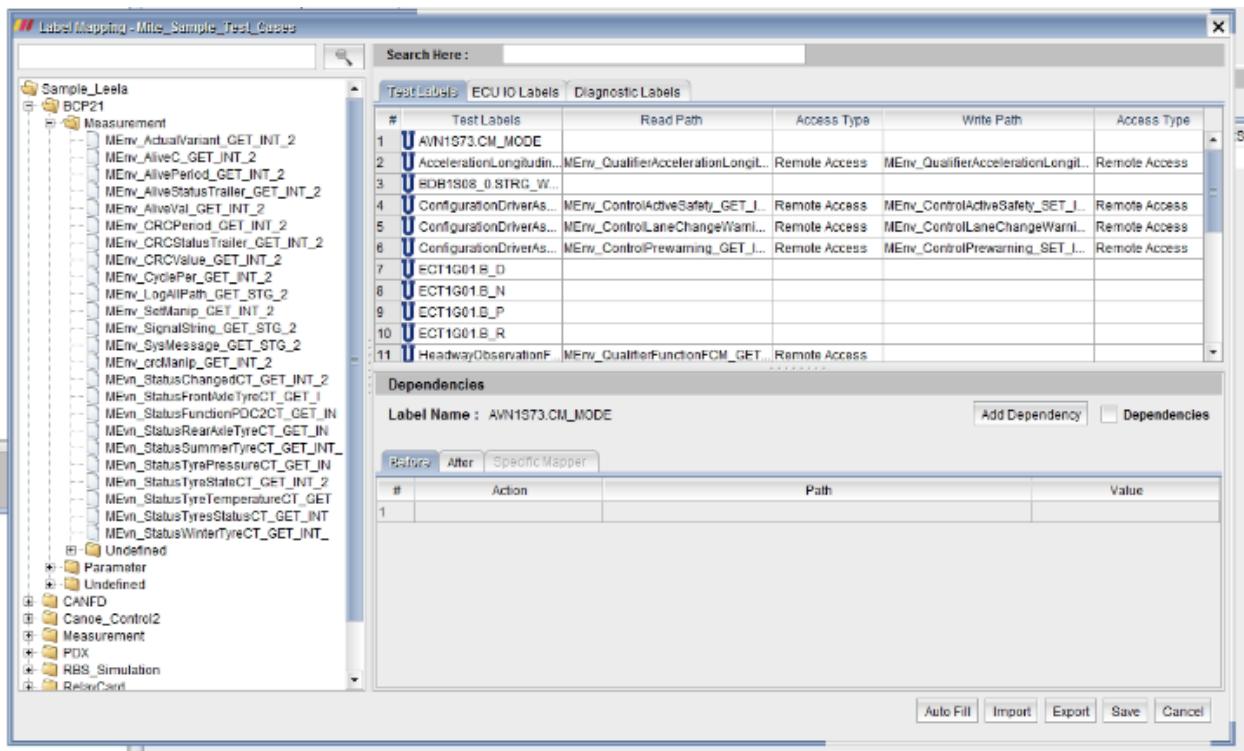


If desired expected value not given simulation will run upto status is idle



LABEL MAPPING DETAILS REPORT .

1.After test script generation project creation ,the label mapping details will shown in report.



Label Mapping Details:					
Test Labels Mapping Info:					
SNo.	Label Name	Read Path	Access Type ToolInterface	Write Path	Access Type ToolInterface
1	HeadwayObservationAE0::eHeadwayObservation.ForecastSafetyBrakeFCM	MEnv_ForecastSafetyBrakeFCM_GET_INT_1	Remote Access_Canoe		
2	ConfigurationDriverAssistanceFunctionsLegacy::controlLaneChangeWarning	MEnv_ControlLaneChangeWarning_SET_INT_1	Remote Access_Canoe	MEnv_ControlLaneChangeWarning_SET_INT_1	Remote Access_Canoe
3	ConfigurationDriverAssistanceFunctionsLegacy::controlPrewarning	MEnv_ControlPrewarning_GET_INT_1	Remote Access_Canoe	MEnv_ControlPrewarning_SET_INT_1	Remote Access_Canoe
4	HeadwayObservationAE1::eHeadwayObservation.reductionThresholdValueBrakeAssistFCM	MEnv_ReductionThresholdValBrakeAssFCM_GET_INT_1	Remote Access_Canoe		
5	HeadwayObservationFCM::headwayObservationFCM.qualifierFunctionFCM	MEnv_QualifierFunctionFCM_GET_INT_1	Remote Access_Canoe		
6	AccelerationLongitudinalLateralProvision::accelerationLongitudinalFCM	MEnv_QualifierAccelerationLongitudinalFCM_GET_INT_1	Remote Access_Canoe	MEnv_QualifierAccelerationLongitudinalFCM_SET_INT_1	Remote Access_Canoe
7	HeadwayObservationFCM::headwayObservationFCM.warningFCM	MEnv_WarningFCM_GET_INT_1	Remote Access_Canoe		
8	ConfigurationDriverAssistanceFunctionsLegacy::controlActiveSafety	MEnv_ControlActiveSafety_GET_INT_1	Remote Access_Canoe	MEnv_ControlActiveSafety_SET_INT_1	Remote Access_Canoe
Diagnostics Labels Mapping Info:					
SNo.	Label Name	Request Path	Access Type Tool Interface	ResponsePath	Access Type ToolInterface
1	Req_5101-STEUERN_ROUTINE/StartRoutine	\$5101-STEUERN_ROUTINE/StartRoutine	Remote Diagnostics_Canoe		
2	Req_57101-STEUERN_ROUTINE/StartRoutine			\$7101-STEUERN_ROUTINE/StartRoutine	Remote Diagnostics_Canoe
3	Req_662-STATUS_LESEN			\$662-STATUS_LESEN	Remote Diagnostics_Canoe
4	Req_514-FS_LOESCHEN	\$14-FS_LOESCHEN	Remote Diagnostics_Canoe		
5	Req_5624119-STATUS_LESEN/_EVEQ_STATE_STAT_EVEQ_STA_TE			\$624119-STATUS_LESEN/_EVEQ_STATE_STAT_EVEQ_STATE	Remote Diagnostics_Canoe
6	Req_52e4048-STEUERN/_BRAKE_AUSLOSUNG_FUNKTIONSVERIFUEGRABERKETT_FCM	\$2e4048-STEUERN/_BRAKE_AUSLOSUNG_FUNKTIONSVERIFUEGRABERKETT_FCM	Remote Diagnostics_Canoe		
7	Req_51091-DEFAULT_SESSION	\$1091-DEFAULT_SESSION	Remote Diagnostics_Canoe		
8	Req_554-FS_LOESCHEN			\$54-FS_LOESCHEN	Remote Diagnostics_Canoe
9	Req_52e4048-STEUERN/_BRAKE_AUSLOSUNG_DAUER_IN_MS_FCM	\$2e4048-STEUERN/_BRAKE_AUSLOSUNG_DAUER_IN_MS_FCM	Remote Diagnostics_Canoe		
10	Req_522-STATUS_LESEN	\$22-STATUS_LESEN	Remote Diagnostics_Canoe		
11	Req_52e4048-STEUERN/_BRAKE_AUSLOSUNG_WARNING_FCM_CRITICALITY	\$2e4048-STEUERN/_BRAKE_AUSLOSUNG_WARNING_FCM_CRITICALITY	Remote Diagnostics_Canoe		
12	Req_52e4048-STEUERN/_BRAKE_AUSLOSUNG_FORECAST_FCM	\$2e4048-STEUERN/_BRAKE_AUSLOSUNG_FORECAST_FCM	Remote Diagnostics_Canoe		

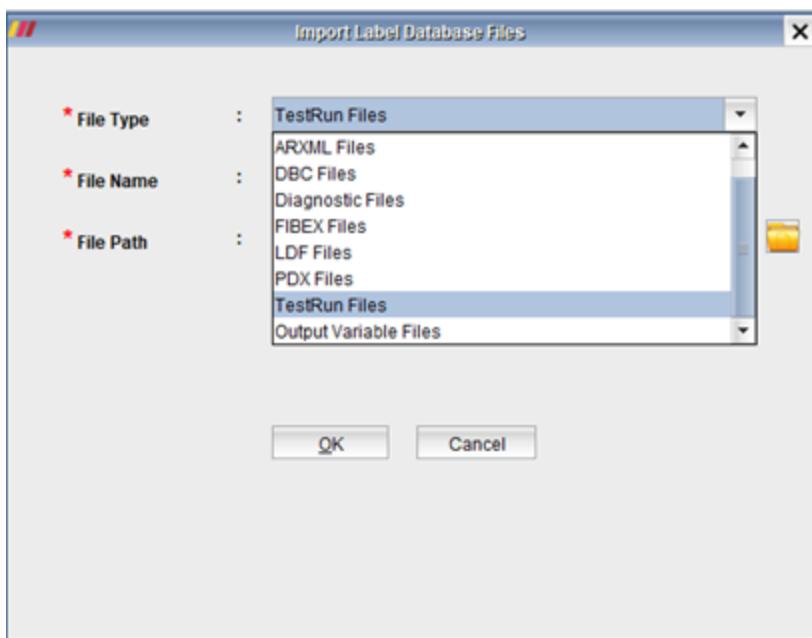
File: MITE Logfile/MITELogFile.30-ile 30-09-2020-10-43-36; i-2020-10-45-36.log - Notepad++

CarMaker User Manual

I. Test Case Role:

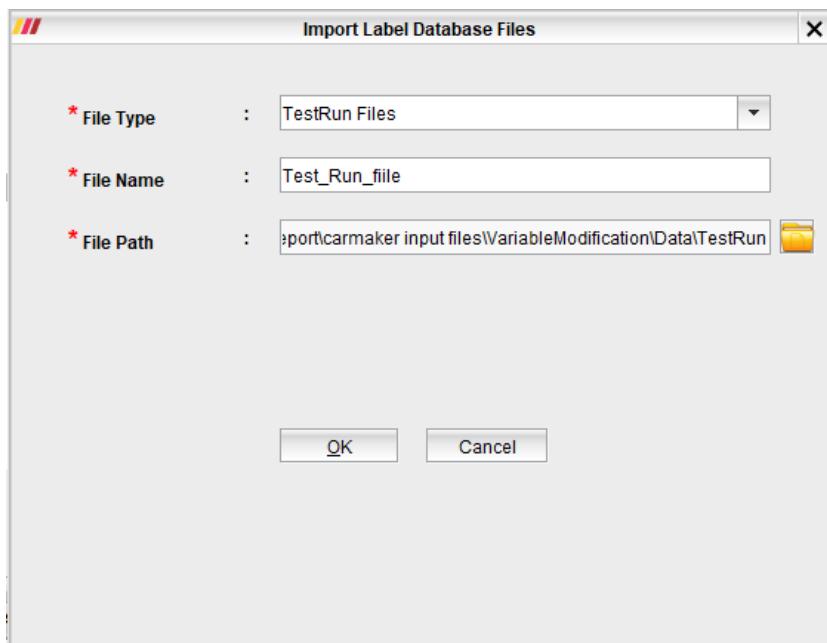
- ***Importing TestRun Files:***

Select the role of a test case. Open the project's navigation window. Choose a test suite. Select the label database file in the settings box. Right-click on the label database option and select Import Label Database. A frame will appear; pick File Type, then TestRun Files, then fill in the remaining information such as file name and file path (The testrun file where you have saved in your local system).



Import TestRun Files

- a) File Type: Select the file type as TestRun Files
- b) File Name: Give a file name
- c) File Path: Provide the testrun file path which is saved in your local system



Imported TestRun Files

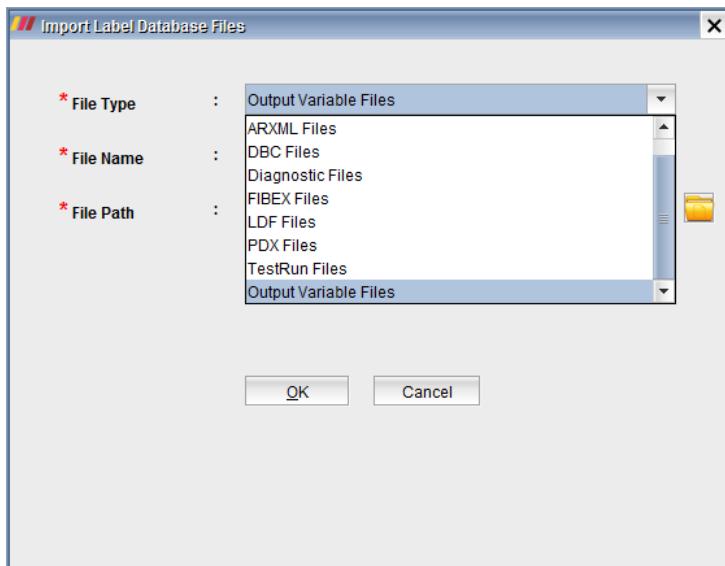
Note:

1. The label database file's path can be specified up to a folder or file.
2. If a folder is selected, all of the label database files are displayed in the dropdown; if a single file is selected, only that file is displayed. The label database file can be Import, Export, Copy/Paste, Reload, Delete etc. upon right click on it.

- ***Importing Output Variable Files:***

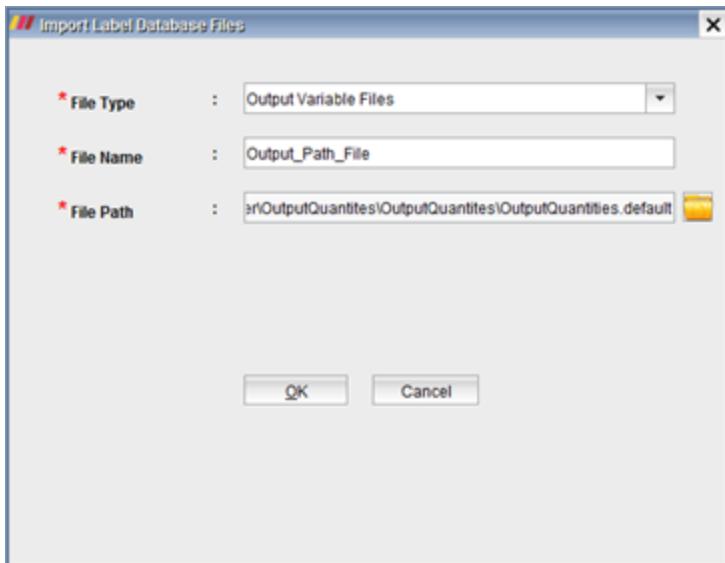
Select the role of a test case. Open the project's navigation window. Choose a test suite. Select the label database file in the settings box. Right-click on the label database option and select Import Label Database. A frame will appear; pick File Type, then Output Variable Files, then fill in the remaining information such as file name and file path (The Output Variable file where you have saved in your local system).

Note: For the output variable file location, it will available inside your project folder → ... → SimOutput folder → ... → output quantities file



Import Output Variable Files

- d) File Type: Select the file type as Output Variable Files
- e) File Name: Give a file name
- f) File Path: Provide the Output Variable file path which is saved in your local system



Imported Output Variable Files

TEST CASE AUTHORIZING:

- In the test case editor, write the test step as follows:
Here is an example test case that shows how to write test steps.

Performance Test - TestCase0...					PTC ID : NEW ITEM	
#	Action	Parameter/Service	Parameter/Description	Desired Value/Expected	Extensions	Additional Rem...
#	Pre Conditions					
#	Test Sequence					
1	Set	Simulation Input	SampleFile.Longitudinal dynamics			
2	While	Test Step (s)			Test Sequence Step 1	
2.1	Check	Simulation Variable Car.v		18		
2.2	Check	Simulation Variable Car.vFL		45		
2.3	Check	Simulation Variable Car.tx		67		
3	Repeat	Test Step (s)			Test Sequence Step 1	
3.1	Set	Simulation Input	SampleFile.Longitudinal dynamics/DrivMan.1.LongDyn\$_spd2	60,60,60,70,80,90,100		
3.2	Set	Simulation Input	SampleFile.Longitudinal dynamics/DrivMan.3.Brake_Stime	1,2,3,1,2,3,4		
3.3	Check	Simulation Variable Car.v		40,50,60,80,90,100,120		
3.4	Check	Simulation Variable Car.vFL		40,50,60,80,90,100,120		
#	Post Conditions					

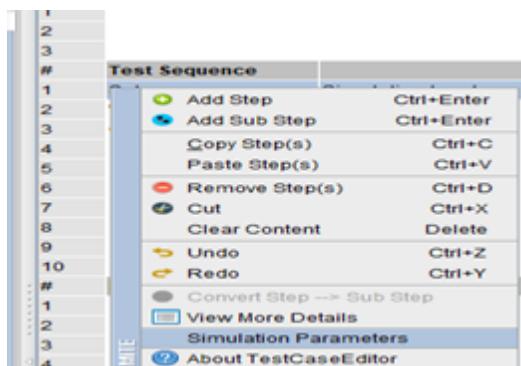
Fig1: Test Case Authoring**Step 1:**

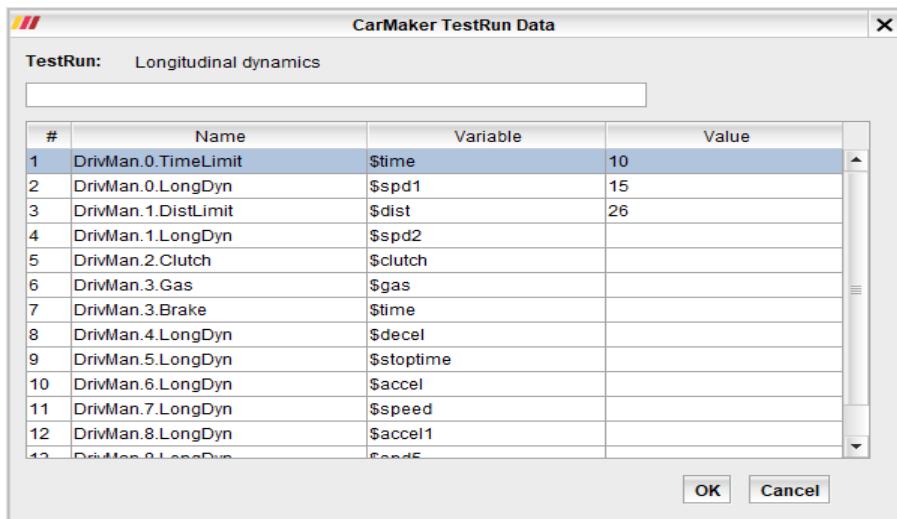
Action	Parameter/ServiceType	Parameter/Description
Set	Simulation Input	Testrun input file

- To obtain the test run file labels in the drop down (Parameter/Description column), the user must import the test run input file in the label database option while writing the testcase.

Assigning value to \$Variable:

- For a test step with simulation input, in order to add the value to their corresponding \$variables. Right-click on the step node and choose Simulation parameters. A frame will be opened with a name, variable, and value column. We must set values to the \$ variables.

**Simulation Parameters**



Assigning value to \$ variable

Step 2 (Check Simulation Variable):

Following that, using the While function writing the check statements (refer step 2 in Fig1)

Step 2.1, 2.2, 2.3:

- Following that, defining the test run output variables (refer step 2.1, 2.2, 2.3 in Fig1)
- To obtain the test run output file labels in the drop down (Parameter/Description column), the user must include the test run output file in the label database option.

Step 3 (Repeat functionality):

Following that, using the Repeat function user can write repeat for \$variables and assign multiple values and can check the respective simulation variables (refer step 3 in Fig1)

Step 3.1, 3.2:

- The user can enter multiple (range of values) in the expected value column, separated by commas (see the test step 3.1 /3.2 in Fig1).
- The test run file name will be displayed in the parameter/description column together with the \$ variables which defined in the repeat step in extensions (Range of Steps).
- Similarly follow the same procedure for test steps 3.3, 3.4 as shown in above Fig 1.

Note: While function is for single value case, Repeat is for multiple value case (Range of values).

II.Test Script Role:

- **CarMaker Tool Interface:**

Choose a test suite. Change the role to the test script role. Go to the configuration window and choose TBC. Select Add configuration details, then Model access from the context menu when you right-click the TBC file. A frame will appear with the CarMaker Matlab tool interface. Then enter a brief name and click next. Give the carmaker's project route (The carmaker output location, carmaker output path, carmaker testrun path where you have saved in your local system). The carmaker tbc file will be added to the tree when you click OK.

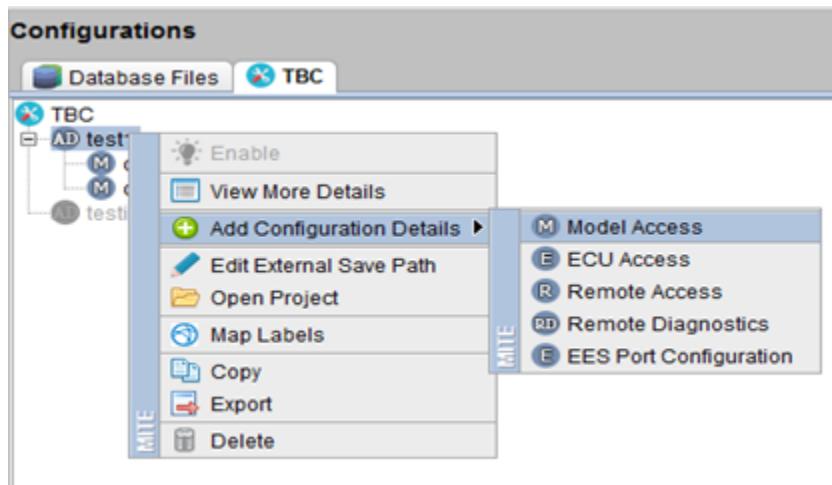


Image 1

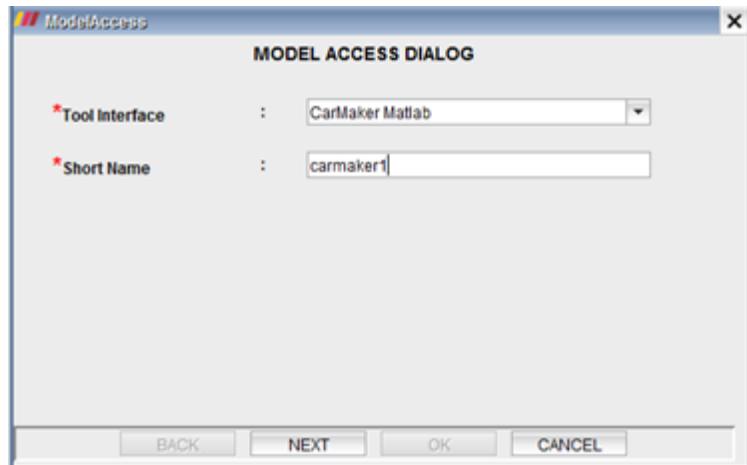


Image 2

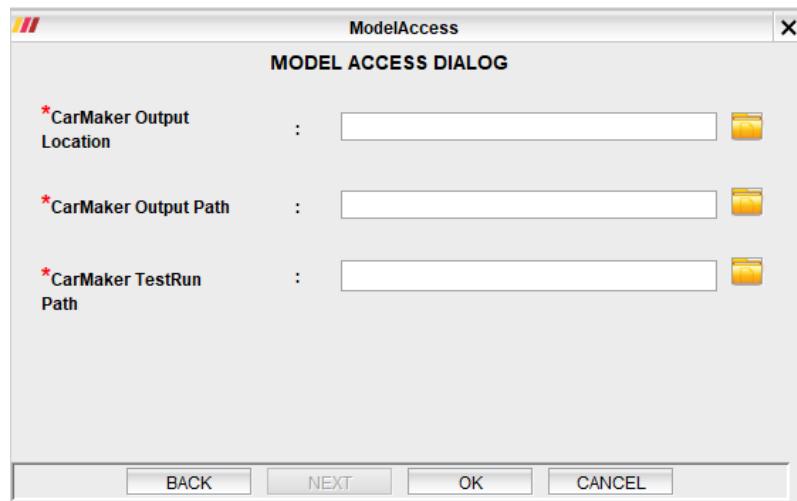
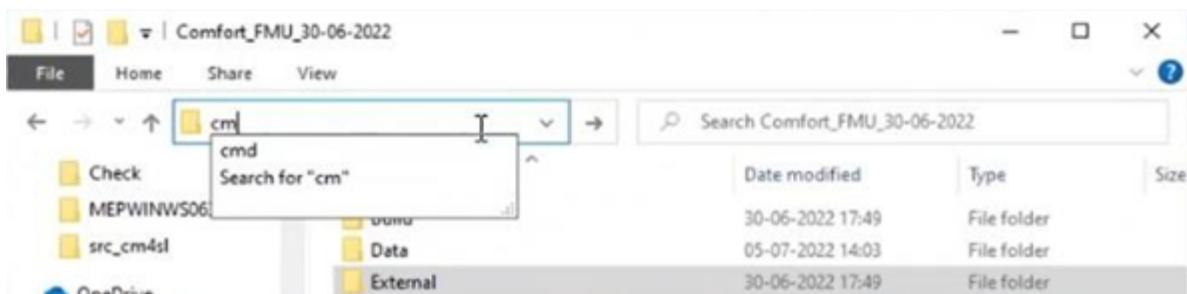


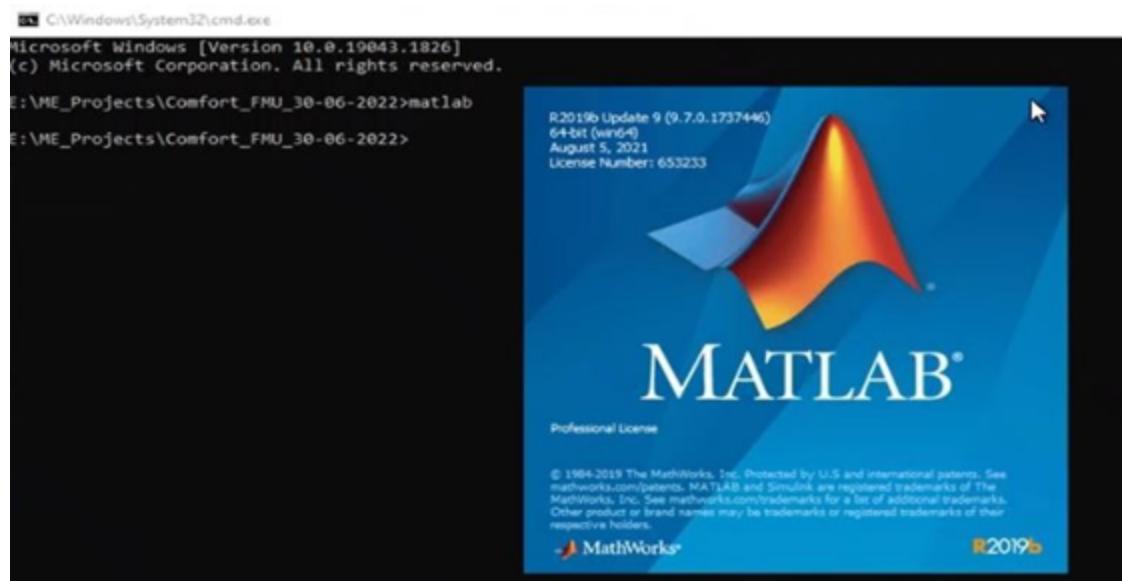
Image 3

CarMaker Output Location: Follow the steps below to specify the carmaker output location.

1. Go to the path of the carmaker output file and then open cmd, where the user can see the location and type "matlab" and press enter. Matlab will launch.



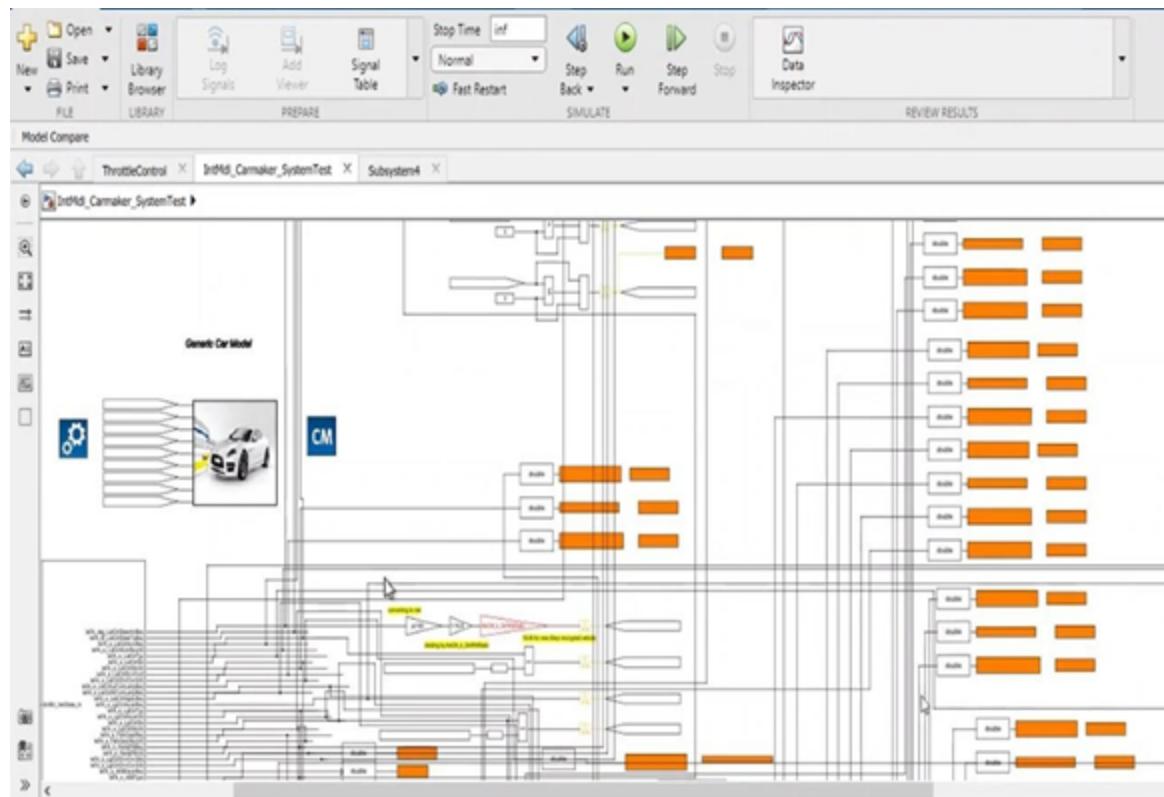
***From carmaker output file
opening cmd***



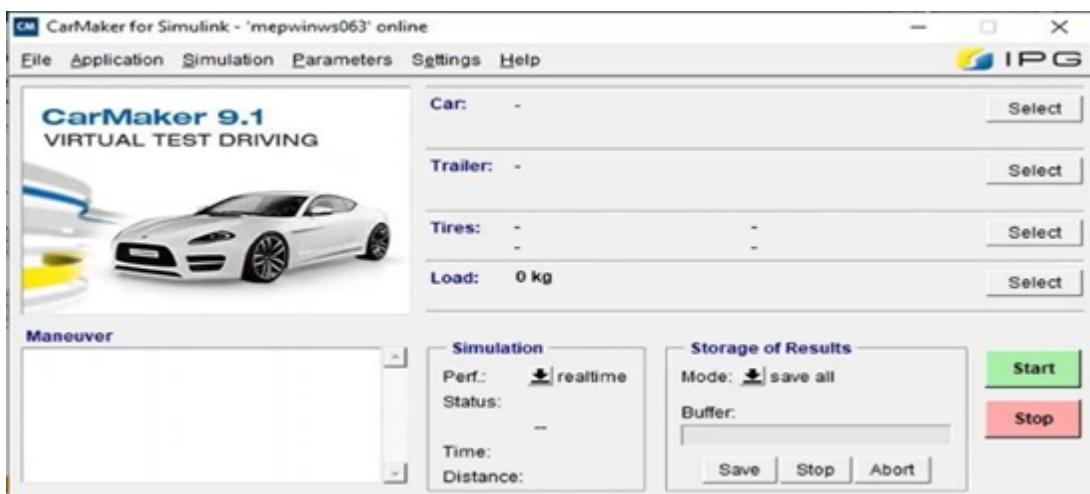
Matlab Launching

2. After launching Matlab, run the Carmaker model, the user will see that several additional files will be running (Neglect it)

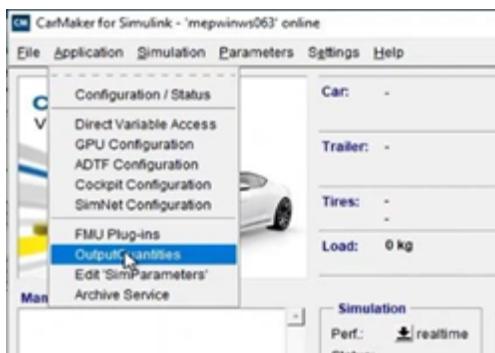
3. A frame will be opened as shown below



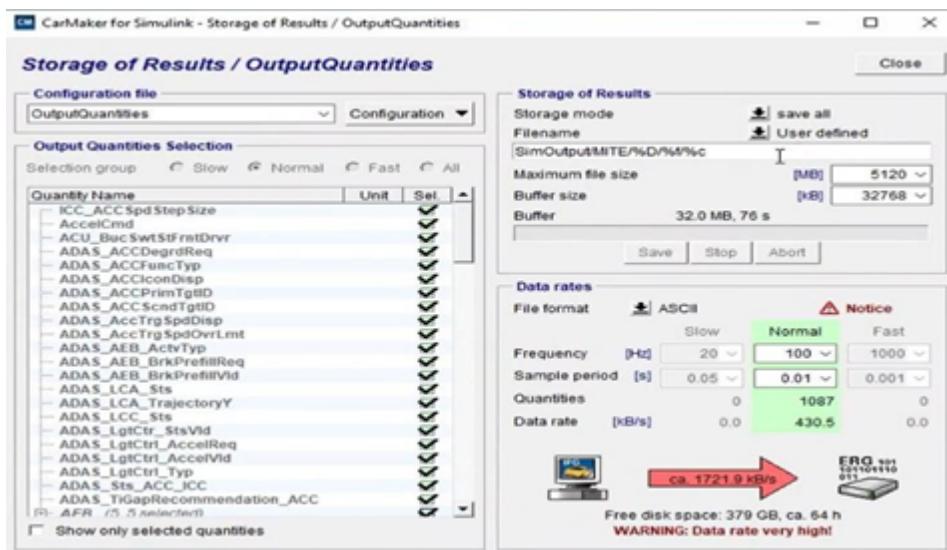
4. Click on CM button to open the carmaker application



5. Select the Application option in file menu, then select Output Quantities.



6. Output quantity frame will be opened



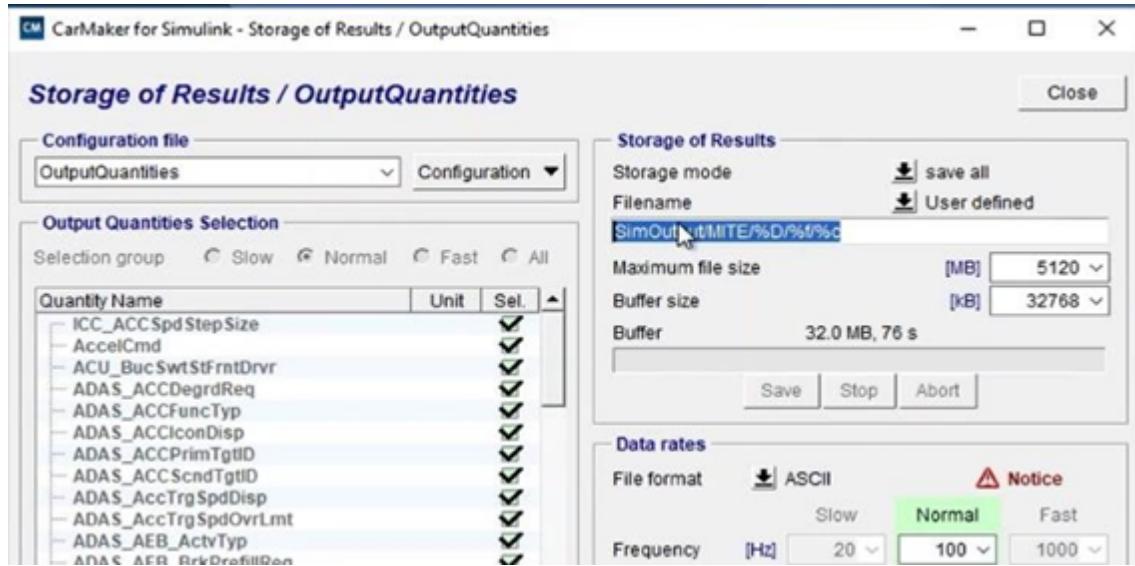
Output quantity frame

7. There is a file name tab where the user can see the carmaker output location (SimOut) to which the extension MITE/%D/%f%c is added. Whatever the extensions added will be saved into file location.

%D = Testrun start date

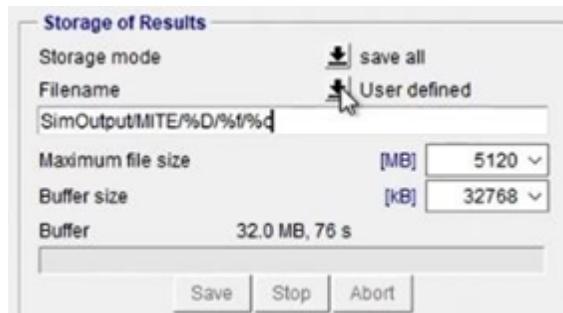
%f = Testrun name

%c = Current time

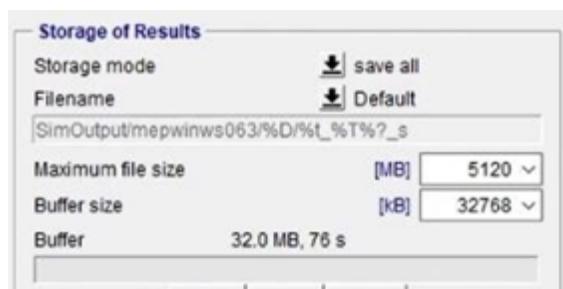


Carmaker output filename

User defined frame will look like



If user clicks on Default, it will be look like



Provide the carmaker output location from the local system, here provided the output location upto

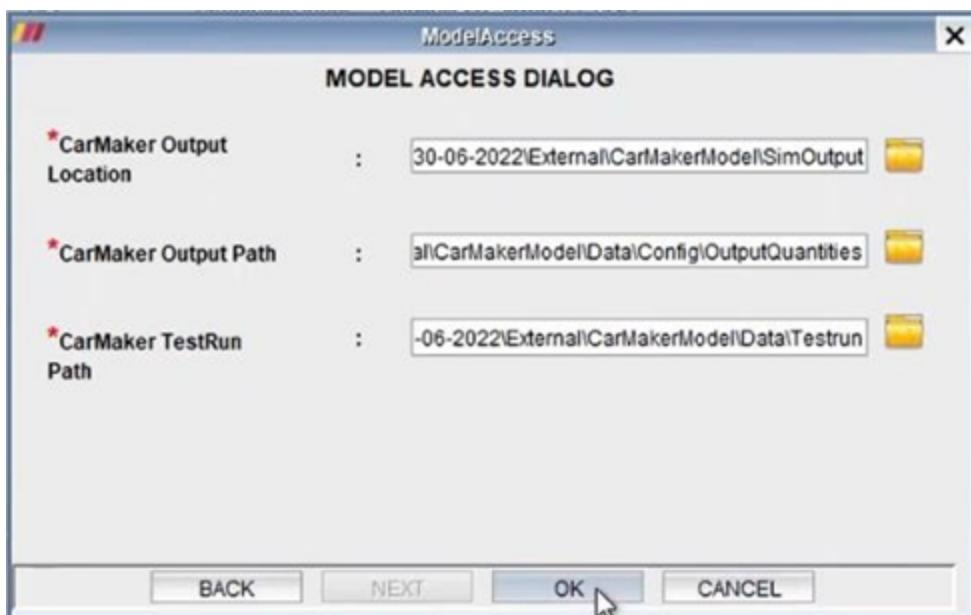
SimOutput



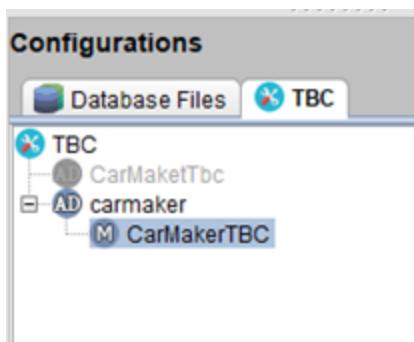
Carmaker output location

CarMaker Output Path: Provide the output file path same as provided in label database file path

CarMaker TestRun Path: Provide the test run file path same as provided in label database file path and click on ok.

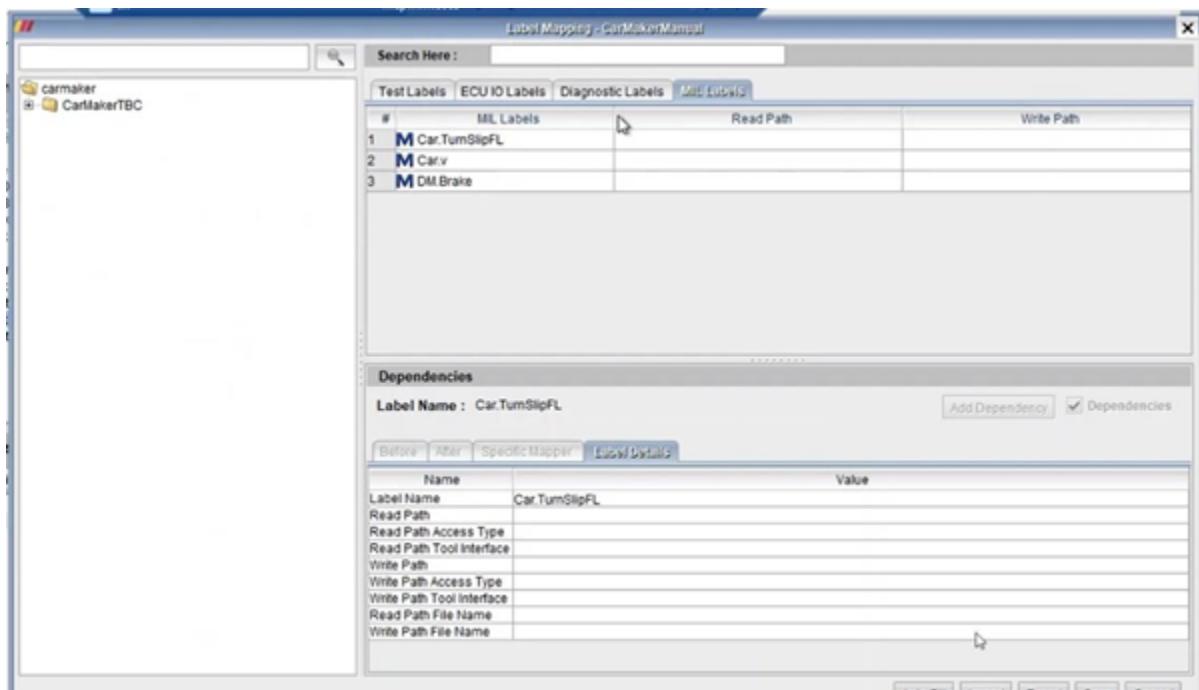


Model Access Dialog



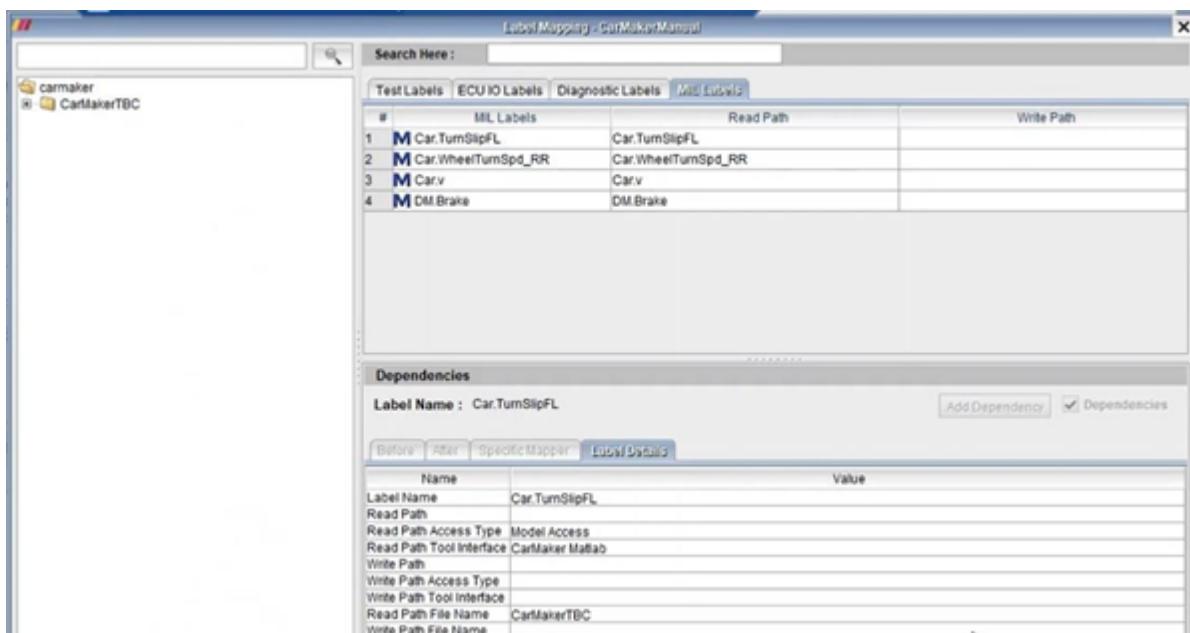
TBC Right click

Be in script mode, select the test suite and check the test cases, navigate to configuration window and Map labels by right click on your respective TBC. A label mapping frame will be opened.



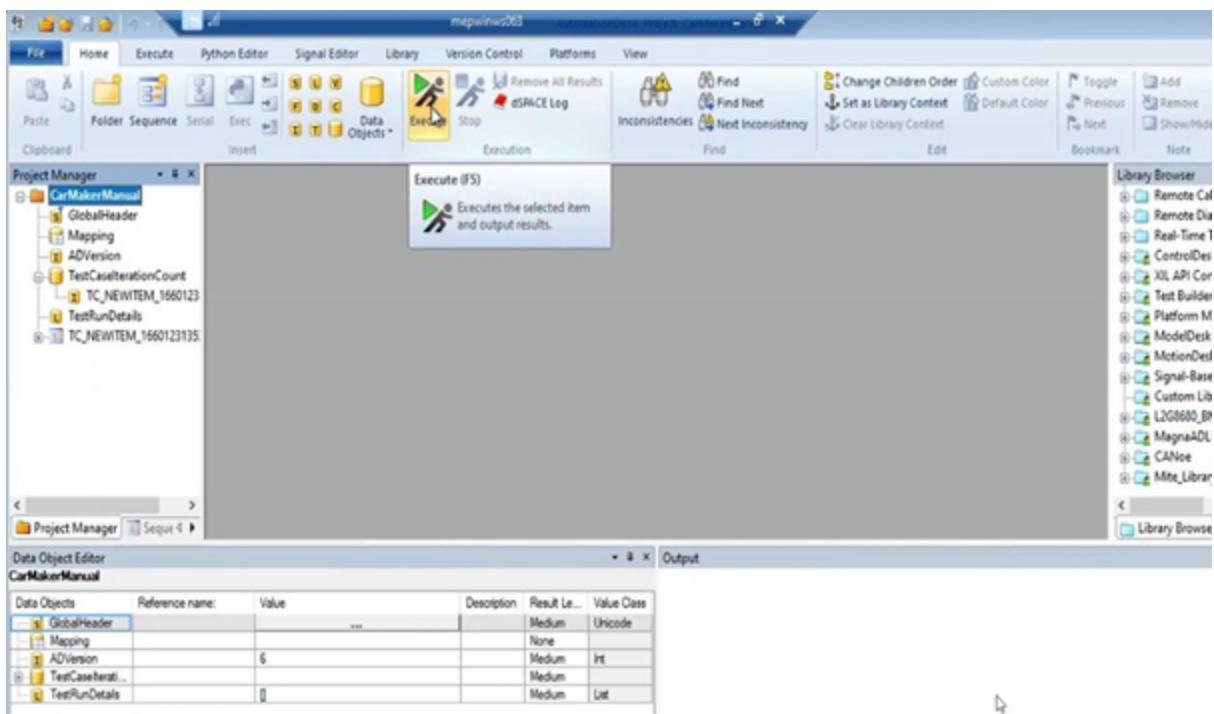
Label mapping frame

A tab called MIL Labels is displayed here, where the user can do label mapping and use the auto fill option to map the labels automatically. Once label mapping is done click on save and close the window.



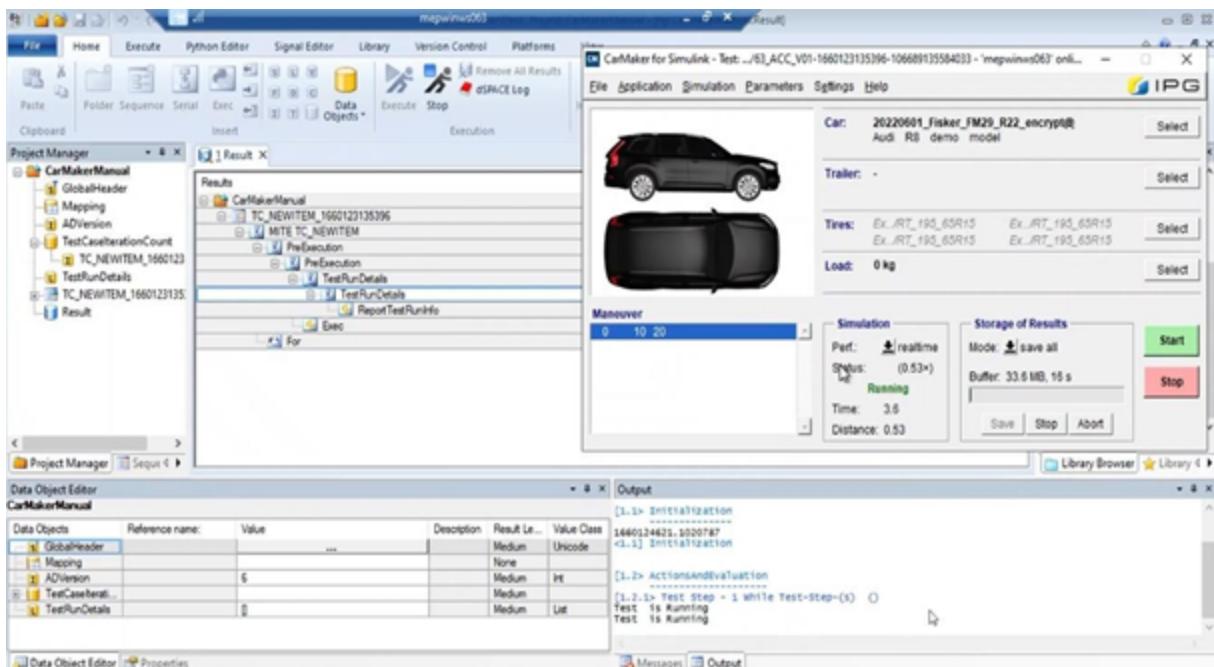
Note: If just the carmaker scenario is provided, no additional tabs are enabled.

Now do the script generation and project creation, open automation desk and execute the test case/suite



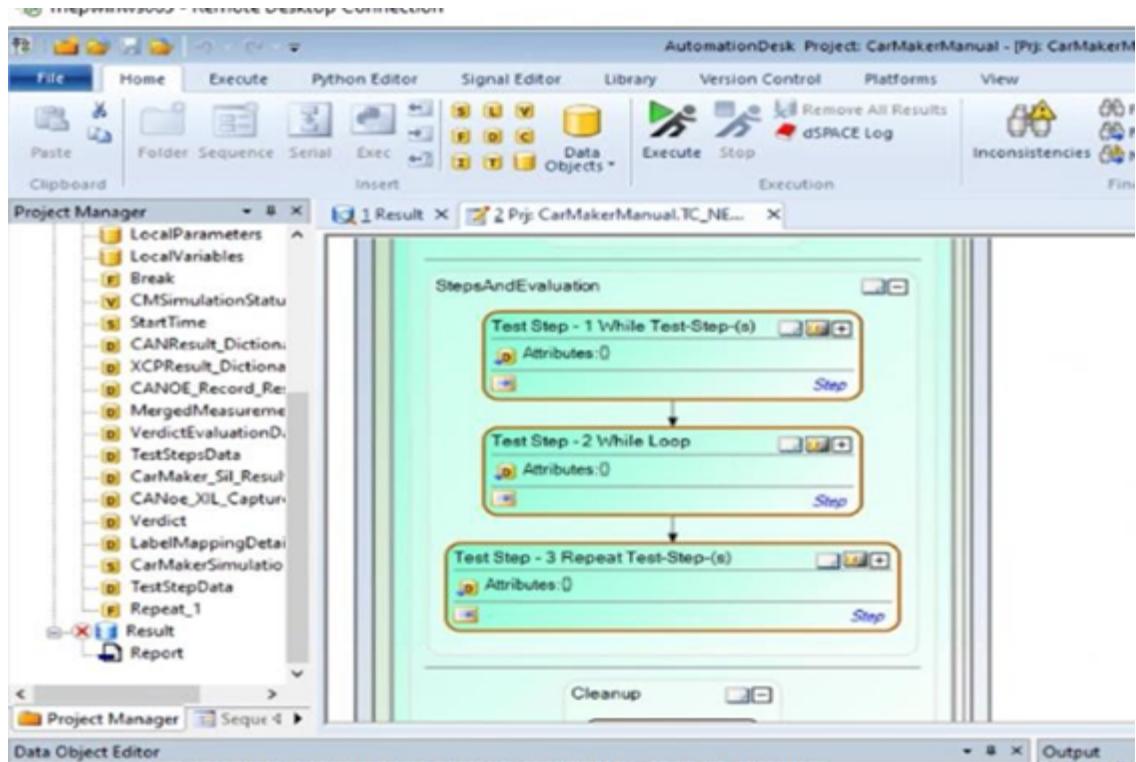
Executing

Execution will start as shown below



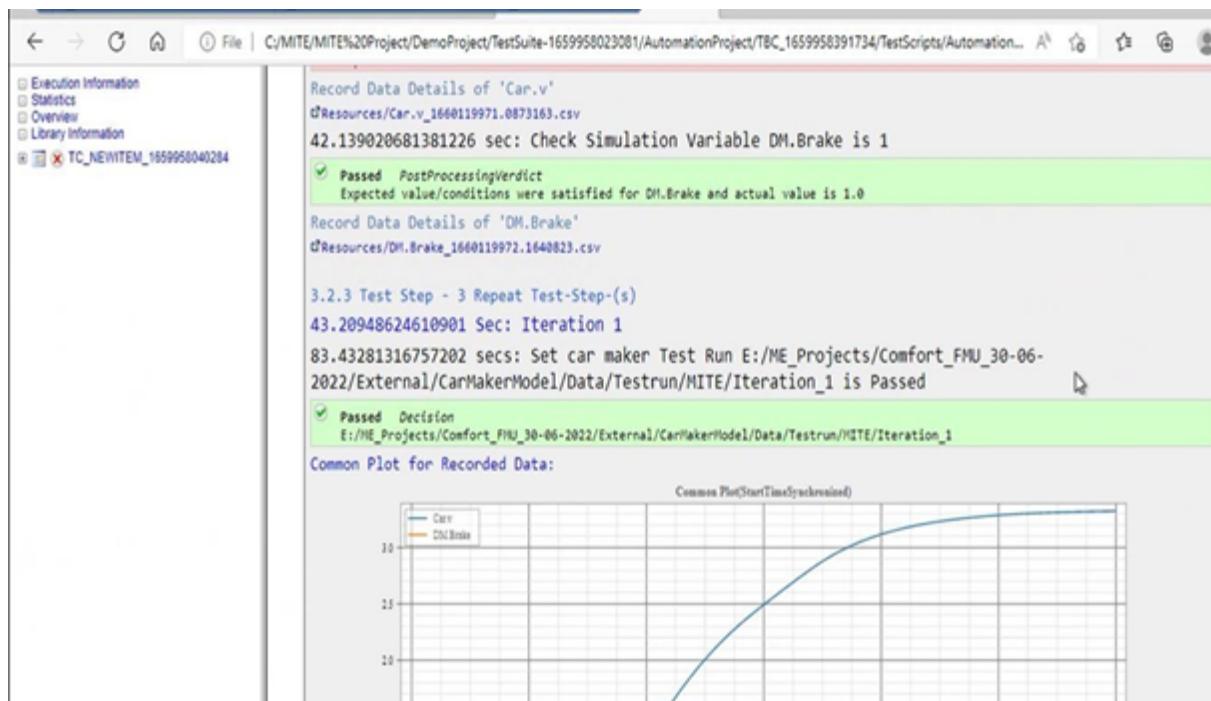
Execution started

Test case blocks will be generated



Blocks generated

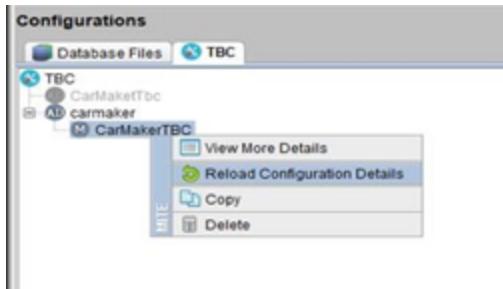
And finally the report will be generated and csv file is provided to show the detailed data of the variable.



Report

TBC Right click option:

User can reload the tbc file by clicking on reload configuration details option



TBC File Right click

- **Reload Configuration Details:**

The user can update the carmaker paths by right-clicking on the tbc file and selecting reload configuration. The tool interface and short name cannot be changed; instead, click Next, where the user can change/edit the carmaker output location, output path, testrun path.



Model Access Dialog 1

**Model Access Dialog 2**

Created with the Personal Edition of HelpNDoc: [Full-featured Help generator](#)

19.1.7 Car maker with matlab(remote Access files)

Car Maker MATLAB User Manual

To use the MATLAB function for the carmaker, In the MITE test case editor, write/include the test step as follows:

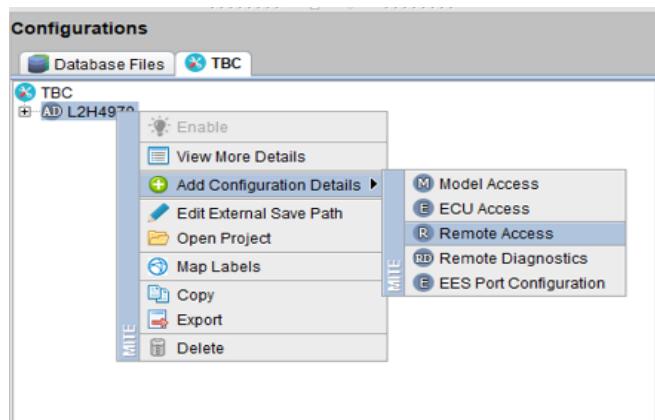
Here is an example test case that shows how to write test step.

#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	Addit...
#	Pre Conditions					
#	Test Sequence					
1	Run	Video	video	carmaker_videos		
2						
#	Post Conditions					

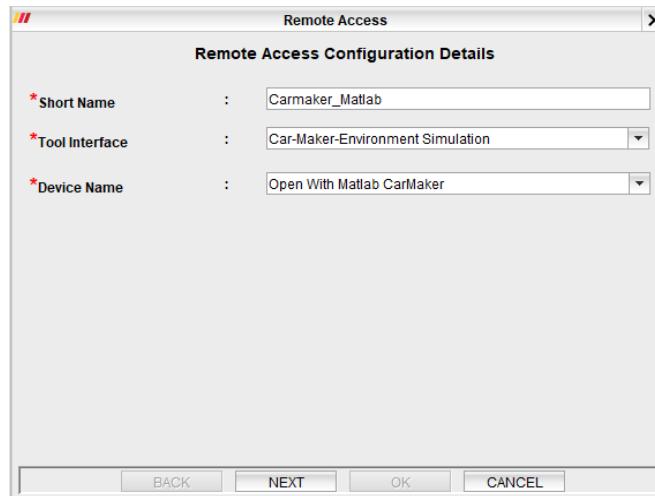
**Action Parameter/ServiceType
Parameter/Description Desired/Expected Value**

Run Video video carmaker_videos

- Choose a test suite. Change the role to the test script role. Go to the configuration window and choose TBC. Select Add configuration details, then Remote Access from the context menu when you right click the TBC file.



A frame will appear with the Remote Access Configuration Details.

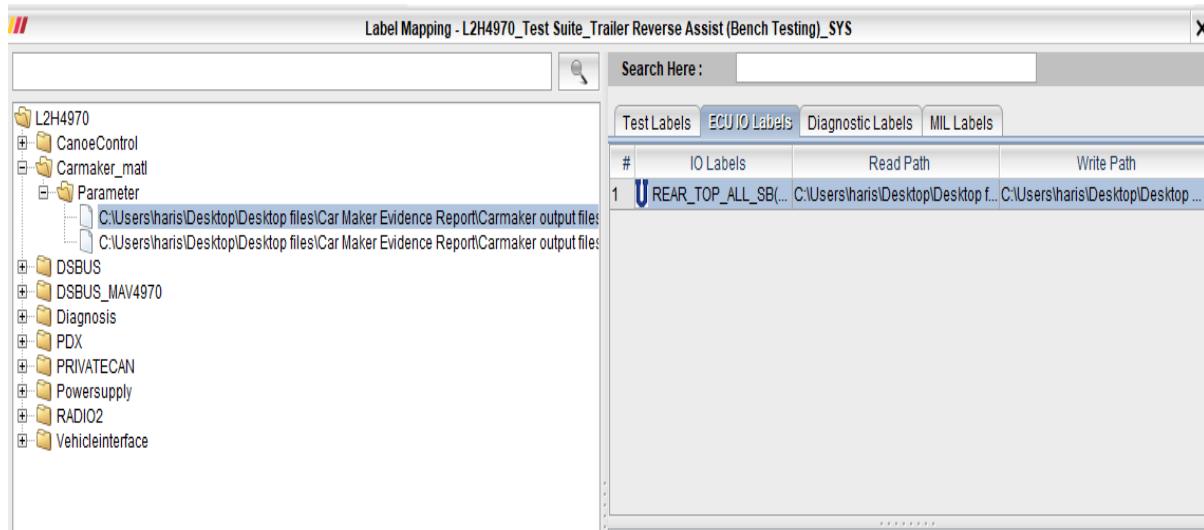


Provide the required files.

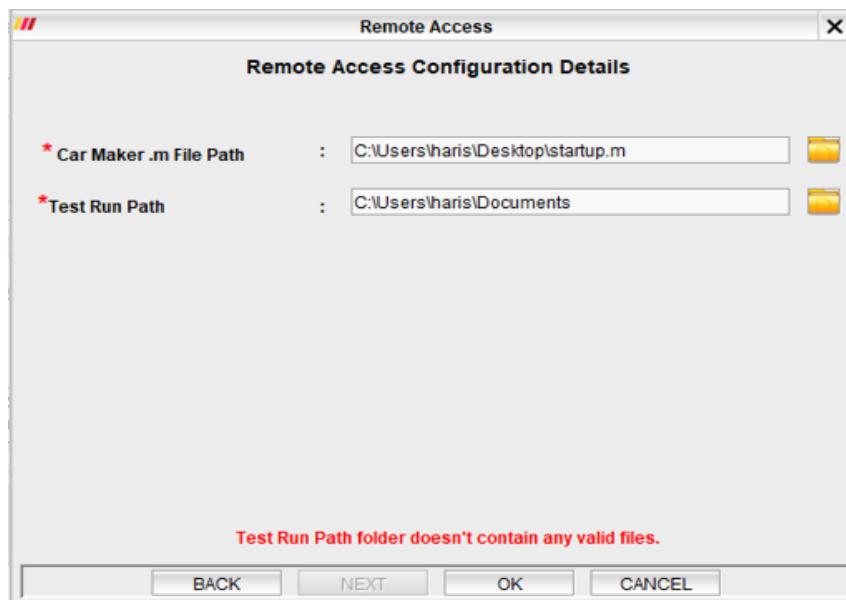
- a) Short Name: Give the short name
- b) Tool Interface: Select the tool interface as Car-Maker-Environment Simulation
- c) Device Name: Depending on the user's requirements, choose the name of the device. If the user uses MATLAB, they should select "Open With Matlab CarMaker".

Then click Next, and a frame will be open in that provide the required files

- a) Car Maker .m File path: Give the .m file
- b) Test Run Path: Provide the test run file path and click on OK
- If valid test run files path is provided, they will appear in the label mapping frame.

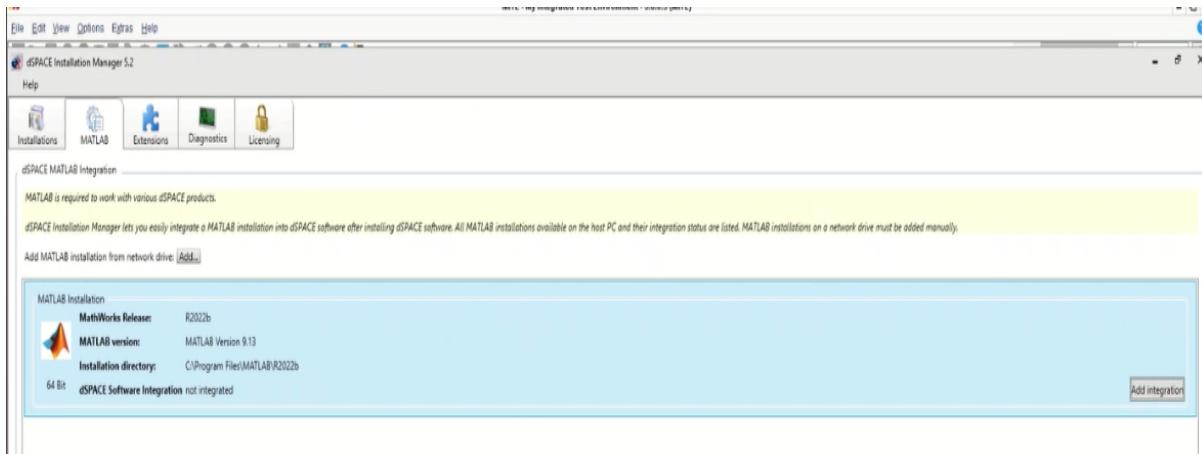


- if invalid files path were provided, an error message will be displayed like “Test Run Path folder doesn’t contain any valid files”.



Note: Admin rights are required to integrate the dSPACE with MATLAB.

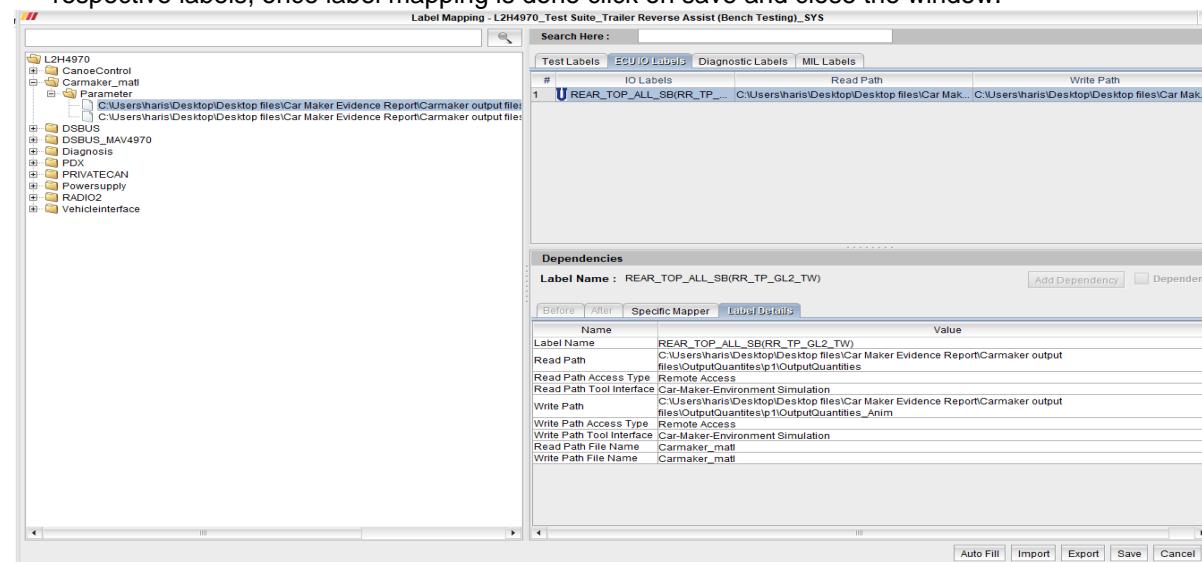
- To integrate Dspace with MATLAB for this, open the dSPACE installation Manager by select Run as administrator, then select the MATLAB tab and click on the Add integration button.



- To disintegrate Dspace with MATLAB, click on the Remove integration button.

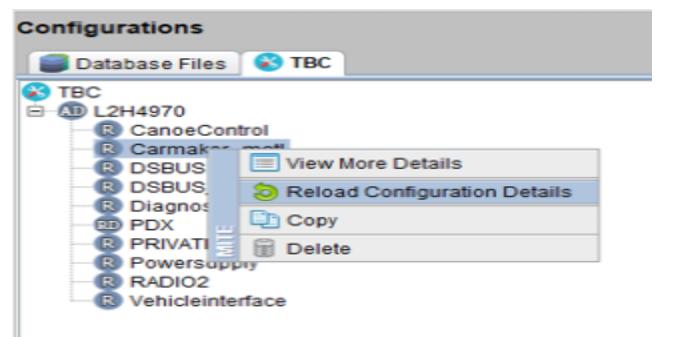


- Be in script mode, select the test suite and check the test cases, navigate to configuration window and Map labels by right click on your respective TBC. A label mapping frame will be opened. Map the respective labels, once label mapping is done click on save and close the window.

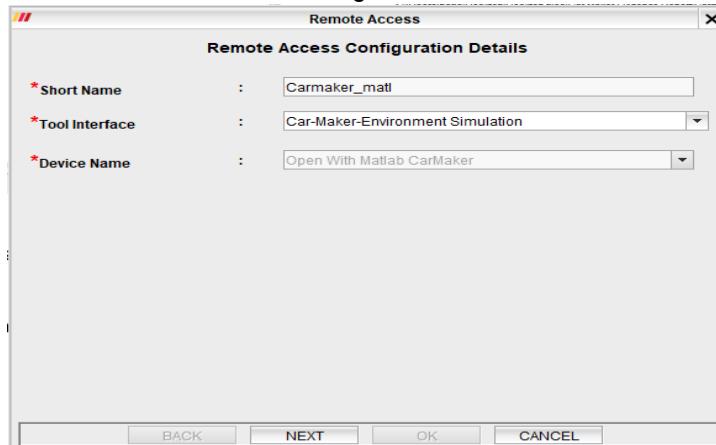


- Do the script generation and project creation when the label mapping is completed. For the particular MATLAB scenario that we were provided, the corresponding blocks will be generated in the automation desk.

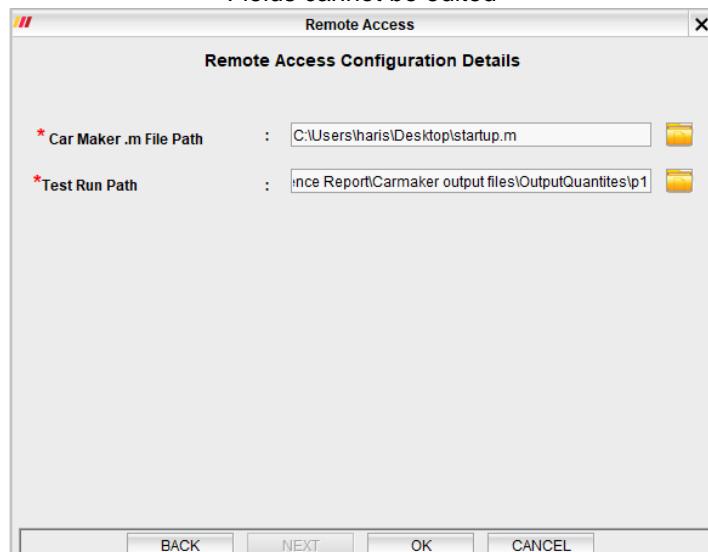
Note: The user cannot edit the short name, tool interface and device name in the TBC right-click reload configuration window, but they can edit the other fields.



Reload Configuration Details



Fields cannot be edited

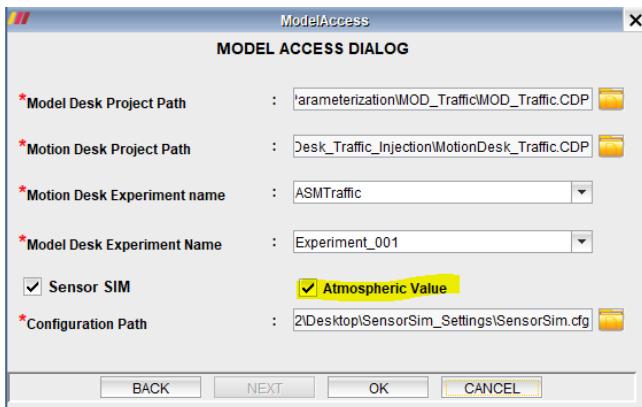


Fields can be edited

Created with the Personal Edition of HelpNDoc: [Easily create EPub books](#)

19.1.8 Atmospheric condition

1. To change Atmospheric scene in Motion Desk user need to select the option Atmospheric value in motion desk



2.User can set custom setting ,preset setting and normal atmosphere

TestCase –

System Test - Atmospheric...							PTC ID : NEW ITEM
#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	...	Script Settings
#	Pre Conditions						
1	Set	Environment	presetAtmospheric				
2	Set	Environment	normalAtmospheric				
3	Set	Environment	customAtmospheric				
4	Set	Environment	variables				
#	Test Sequence						
#	Post Conditions						

Mapping –

Label Mapping - I_2GB680_BMW_AOCAM_System_Test_Spec_1 TA			
Search Here :			
Test Labels ECU IO Labels Diagnostic Labels			
CLH			
+ Env_Variables			
- MD			
+ Atmospheric Variables			
+ Custom Variables			
+ Normal Variables			
+ Preset Variables			
+ Environment Simulation Variables			
+ Model Variables			
+ RD			
+ SYS_sys			
Test Labels ECU IO Labels Diagnostic Labels <th data-kind="ghost"></th> <th data-kind="ghost"></th> <th data-kind="ghost"></th>			
#	IO Labels	Read Path	Write Path
1	customAtmospheric		Custom.Fog
2	normalAtmospheric		Atmosphere.Night
3	presetAtmospheric		Preset.Snowy
4	variables	(.)\ASM_Traffic\Model Root\MDL\VehicleDynamic...	(.)\ASM_Traffic\Model Root\MDL\VehicleDynamic...

For Atmospheric value can be set

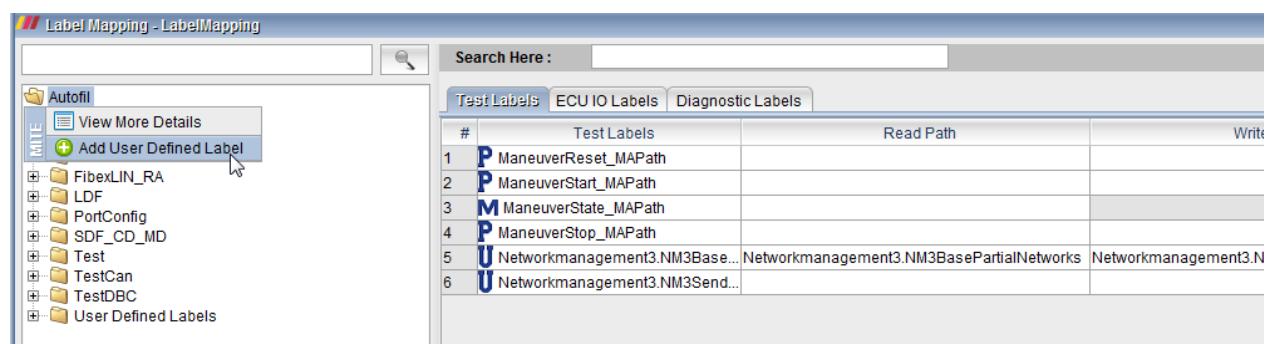
Atmospheric Values		
Name	Enable	Value
Intensity	<input type="checkbox"/>	
Rain	<input checked="" type="checkbox"/>	
Speed	<input checked="" type="checkbox"/>	7
Intensity	<input checked="" type="checkbox"/>	0.02
Particle Size	<input checked="" type="checkbox"/>	0.04
Falling angle	<input checked="" type="checkbox"/>	70
Heading angle	<input checked="" type="checkbox"/>	20
Snow	<input type="checkbox"/>	
Speed	<input checked="" type="checkbox"/>	8
Intensity	<input checked="" type="checkbox"/>	0.02
Particle Size	<input checked="" type="checkbox"/>	6
Falling angle	<input checked="" type="checkbox"/>	22
Heading angle	<input checked="" type="checkbox"/>	34

[Save](#) [Cancel](#)

Created with the Personal Edition of HelpNDoc: [Full-featured EPub generator](#)

19.1.9 User Defined labels in TBC

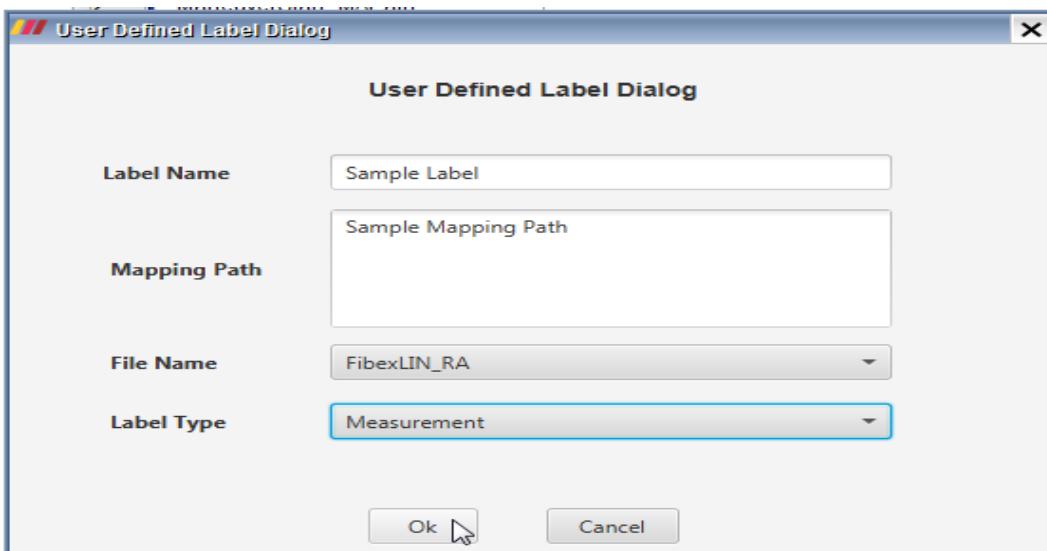
From Version 3.0.0.0, user can add custom label to the label Mapping Frame.



The screenshot shows the 'Label Mapping - LabelMapping' dialog. On the left, there is a tree view with nodes like 'Autofill', 'View More Details', 'Add User Defined Label' (which is highlighted with a blue box), 'FibexLIN_RA', 'LDF', 'PortConfig', 'SDF_CD_MD', 'Test', 'TestCan', 'TestDBC', and 'User Defined Labels'. On the right, there is a table with three tabs: 'Test Labels', 'ECU IO Labels', and 'Diagnostic Labels'. The 'Test Labels' tab is selected, showing the following data:

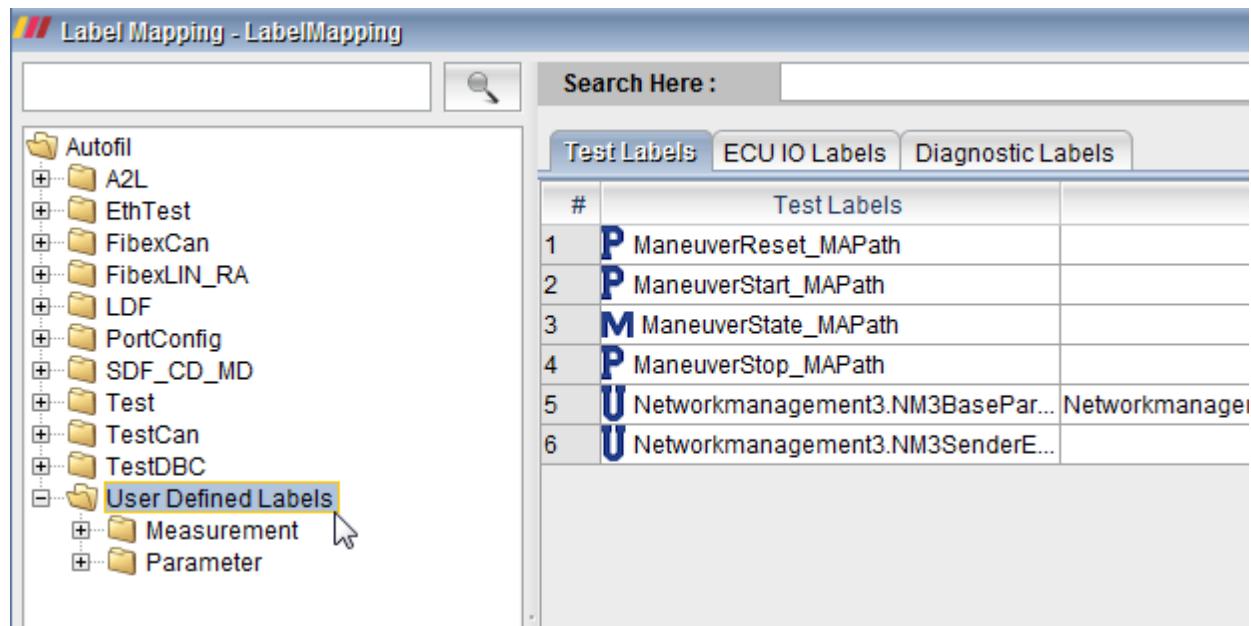
#	Test Labels	Read Path	Write Path
1	P ManeuverReset_MAPath		
2	P ManeuverStart_MAPath		
3	M ManeuverState_MAPath		
4	P ManeuverStop_MAPath		
5	U Networkmanagement3.NM3Base... Networkmanagement3.NM3BasePartialNetworks	Networkmanagement3.NM3Send...	Networkmanagement3.NM3BasePartialNetworks
6	U Networkmanagement3.NM3Send...		

For this we have a right click option named "Add User Defined Label", upon selection a dialog will be displayed asking for required label details.



We need to fill the required Label Name, Mapping Path, File Name(this is a combo box which displays imported Access files and we need to select one among them) and Label Type(this is also a combo box of three type i.e., Measurement, Parameter, Undefined).

After filling the details the newly added User Defined Label will be displayed in Label Mapping Tree.



Created with the Personal Edition of HelpNDoc: [Create cross-platform Qt Help files](#)

19.1.10 Single\Multiple Bites and Bytes Information

Check diag response bit and byte values

1) Hex: Hex value will start with 0x and followed by byte information.

Don't care conditions:

- a) XX...XX: In MITE, if user wants to ignore n bytes, then xx...xx can be used. This will be converted into ... to the python script.
In the python script ... is treated as don't care condition and the Verdict will be given as passed.
- b) XX, X0, 0X: To ignore a value in a byte X will be replaced with that value,

To ignore the complete Byte XX will be used, to ignore the first value of the byte X0 is used, to ignore the second value of the Byte 0X is used.

2) **Bit values:** Bit values can be written as b(), which consists of 0,1,x. Only 8 bits are allowed. To ignore any bit value, user can replace it with x (which will be considered as don't care condition)

Syntax for bit values: for 01 to binary --- b(00000001)

Different ways to write bit values:

b(00001111)

b(0000X1XX)- Including x

b(0x0ss100) – Wrong syntax, only 0,1,x can only be included.

Ex : 1. 0x b(00001111)

2 . 0x 45 b(00001111)

3) **ByteAt:** To evaluate a byte position with a value using relational operators, ByteAt is used

Syntax: ByteAt[(**BytePosition:RelationalOperatorExpectedValue**)]

E.g. To check the 0th index of Expected value is equal to 2f.

ByteAt[(0:==2f)]

0	---	Byte position
==	---	Relational Operator
2f	---	Expected Value _

Relational Operators:

Equal to	---	==
Not equal to	---	!=
Less than	<	
Greater than	>	
Less than equal to	---	<=
Greater than equal to	---	>=

Different ways to write ByteAt syntax :

ByteAt[(0:==2f)] ----- for single byte comparison

ByteAt[(0:==2f),(1:!=41)] ----- for multiple byte comparison

ByteAt[(0-3:==2f)] -----for range of byte positions

Basic Validations for ByteAt :

- 1) The byte position and relation Operator are separated with a colon (:)
- 2) Range can be separated by a dash (-) 0-3 means from 0th position to 3rd position.
- 3) Comma (,) can be used as a separator for a set of evaluation.
- 4) Square brackets and circular brackets to be placed at the positions as required.

19.1.11 Remote Batch Automation

Batch Automation

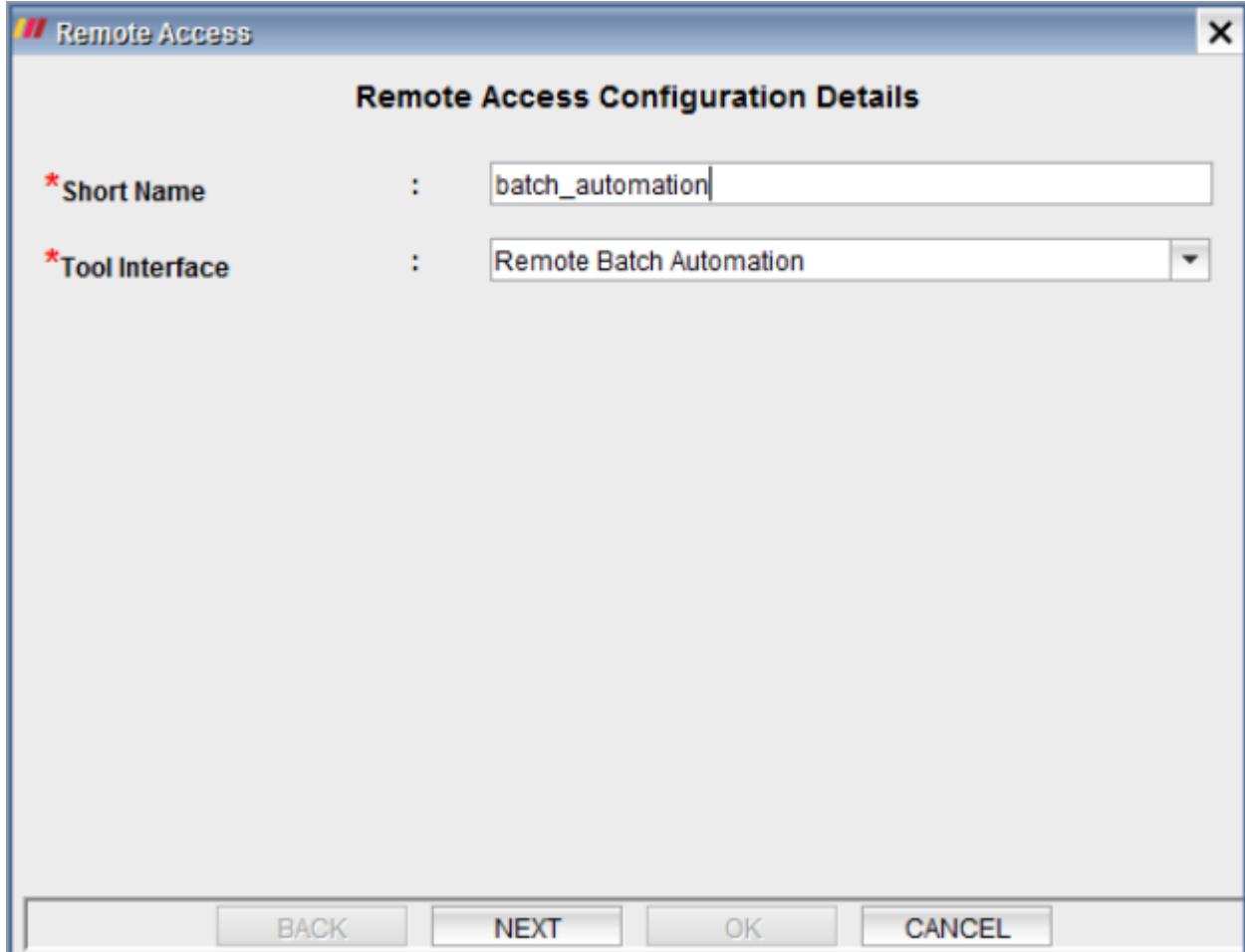
We have added a new feature from MITE -v for the batch automation in the TBC Remote Access

configuration.

To use this feature we need to follow below steps for the configuration:-

Scenario 1:-

Step 1:- Go the test bench configuration in the test script viewset add the configuration details in Remote access by giving the Short Name and Tool Interface.



Step 2:- On Clicking the next option give the Type of file,Batch file descriptor(.ini),Post Execution Validation string and default max execution time and click on OK option after submitting the details.

Remote Access

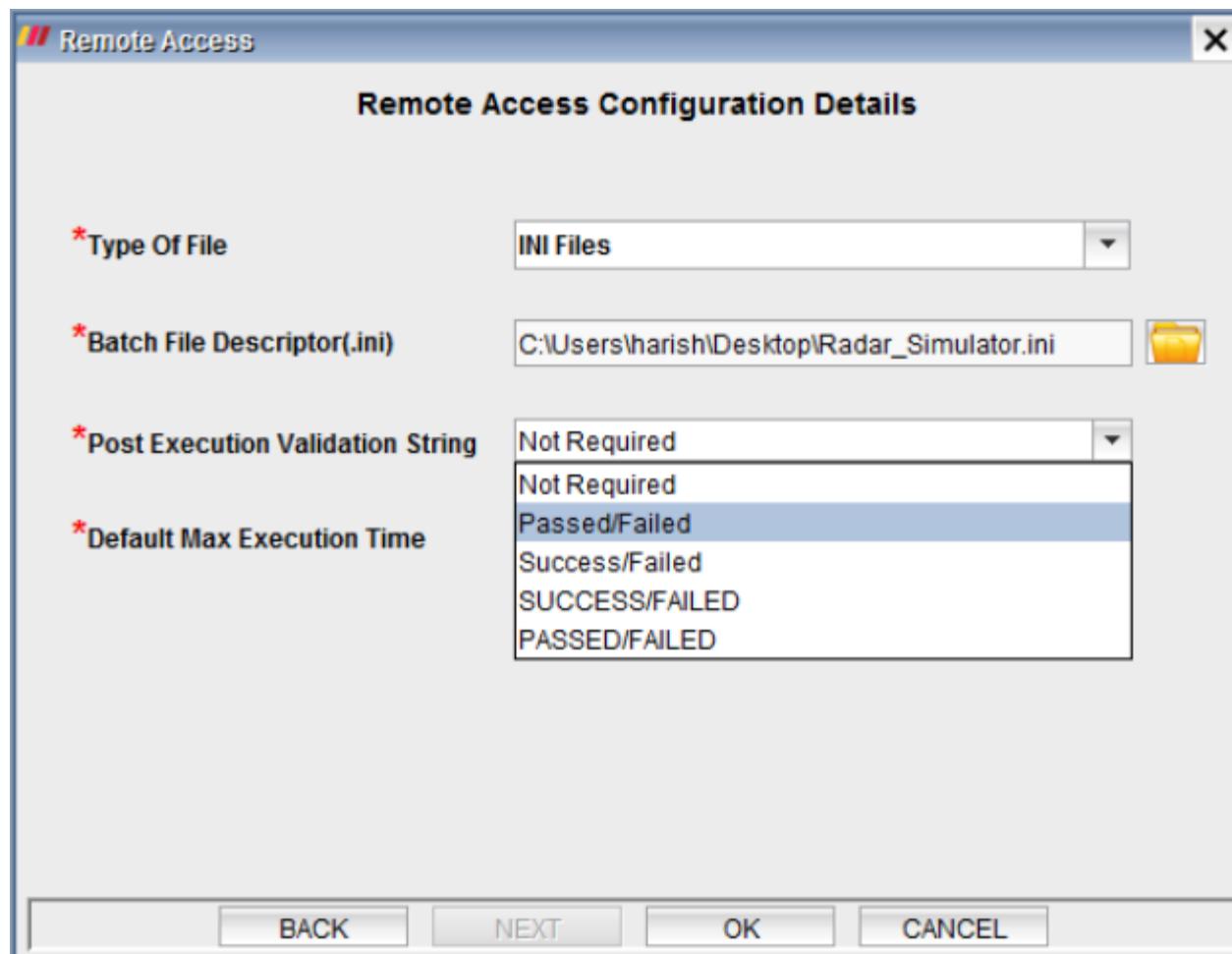
Remote Access Configuration Details

*Type Of File

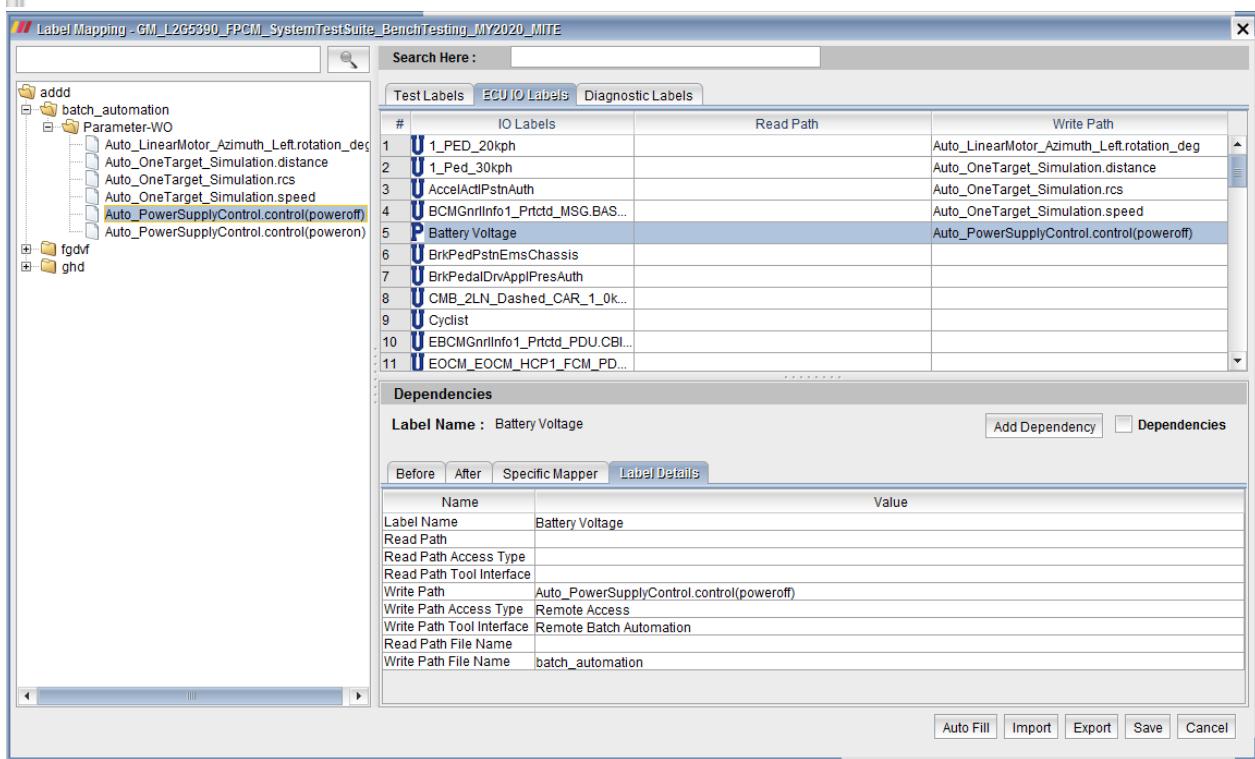
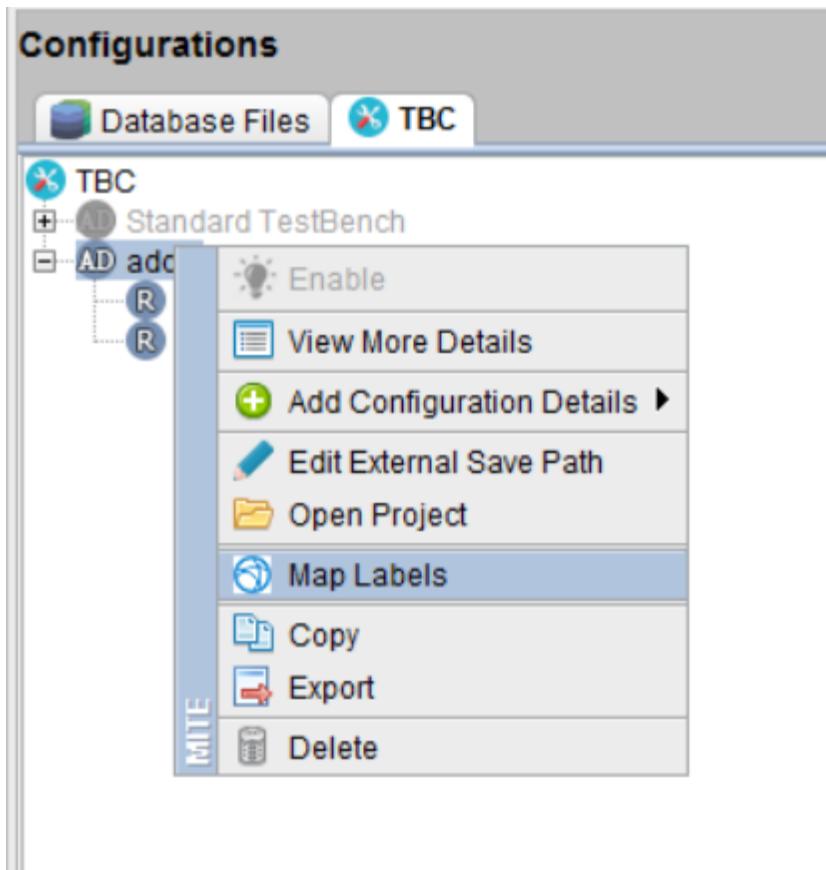
*Batch File Descriptor(.ini)

*Post Execution Validation String

*Default Max Execution Time Secs



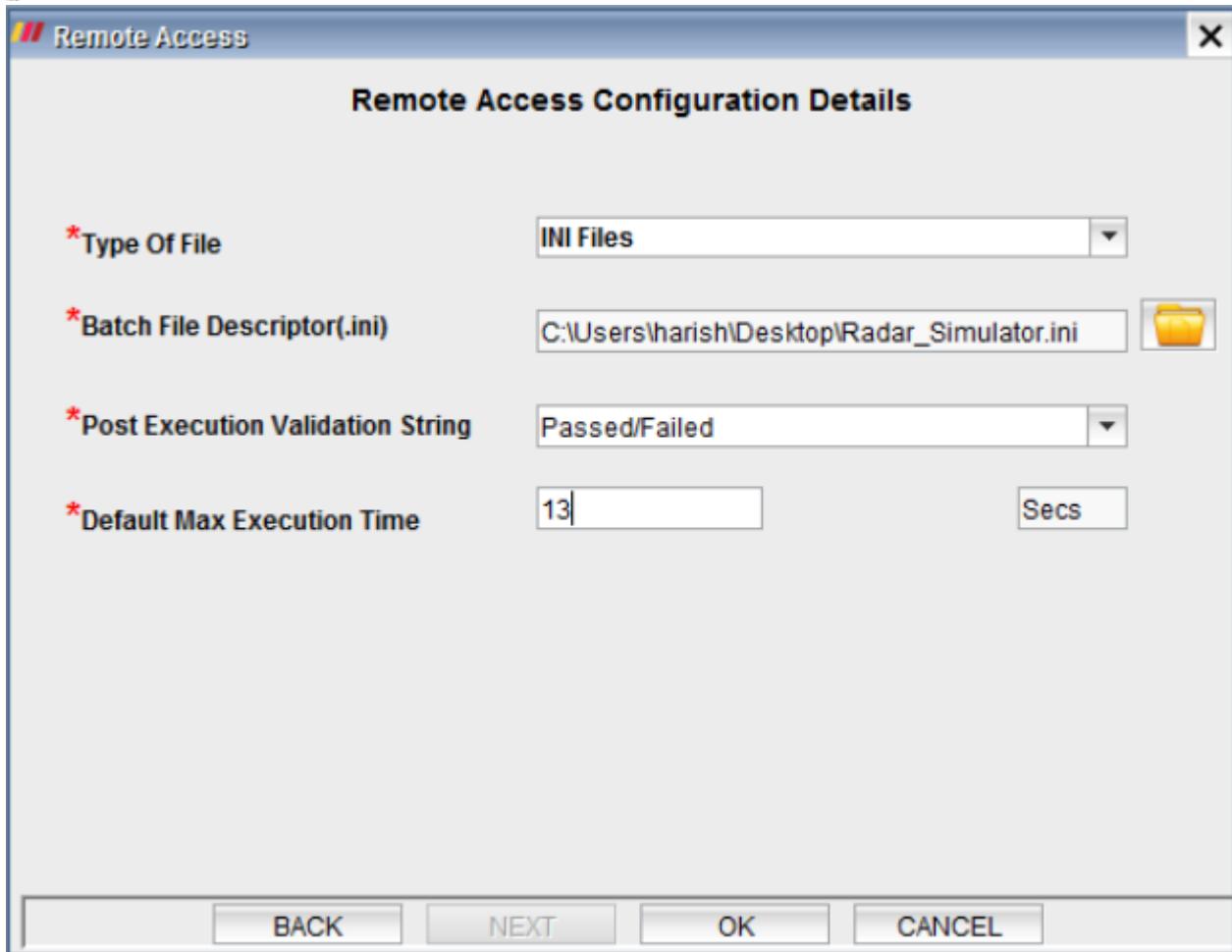
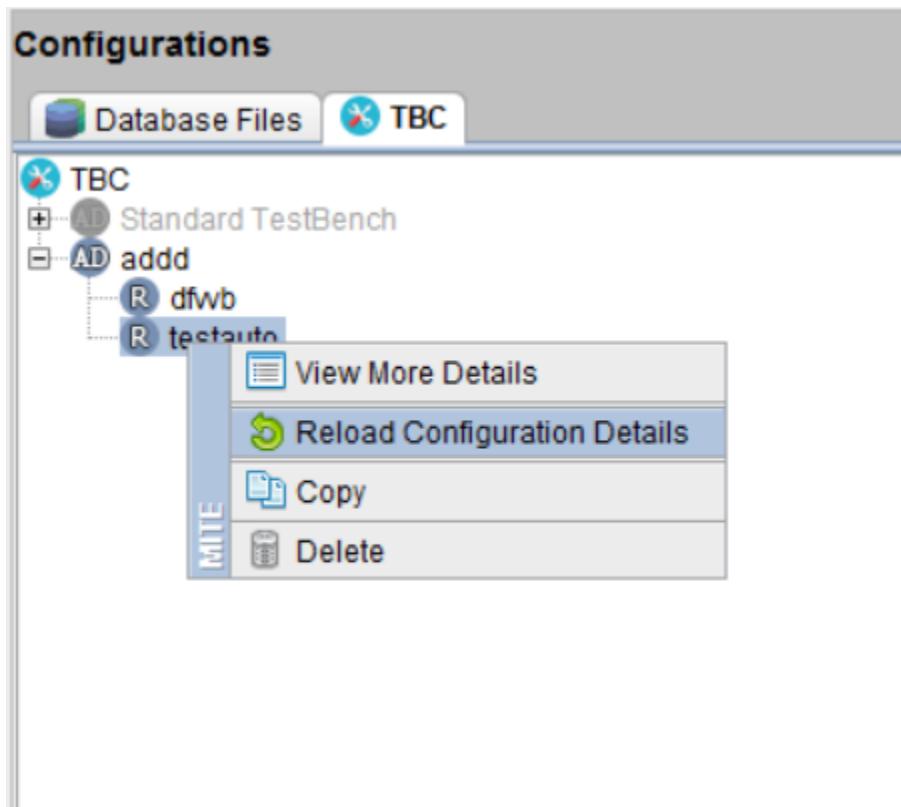
Step 3:-After importing the configuration file User needs to do label mapping for those particular files imported in remote access configuration.User can give the write path for each label in ECU IO labels tab in the map label frame.



After the label mapping is done User can execute the scripts and create the project creation in the Automation desk.

Scenario 2:-

Step 1:-User can reload the configuration file incase he/she wants to reload any new particular files and use them for creating the project.



Step 2:-After reloading the particular file user can do script generation and create the project in the automation desk.

19.1.12 Power Relay-Neo RAD

Power Relay- neoRAD-IO-2-PWRRLY

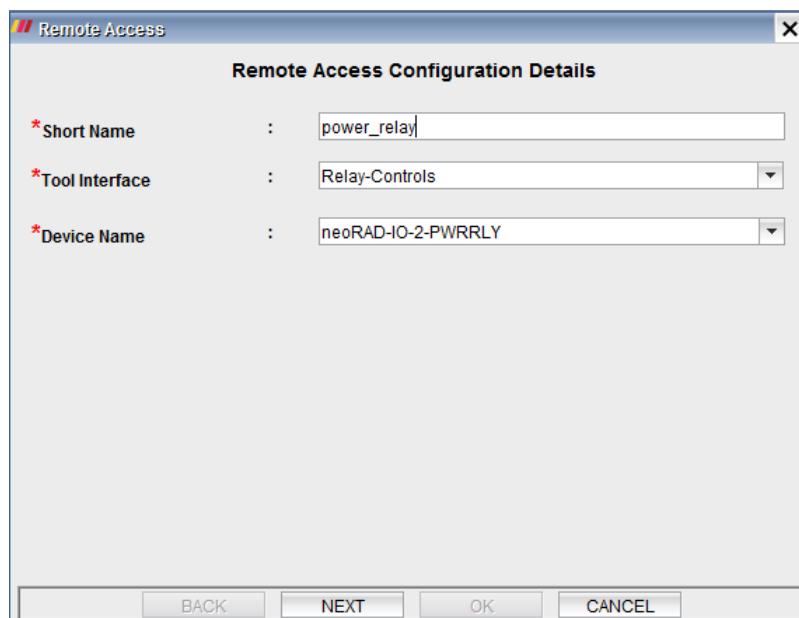
Install neoRadio2 through command Prompt:

Open the command prompt, and type “Pip install neoradio2 –user”. Click Enter then it will install for the user.

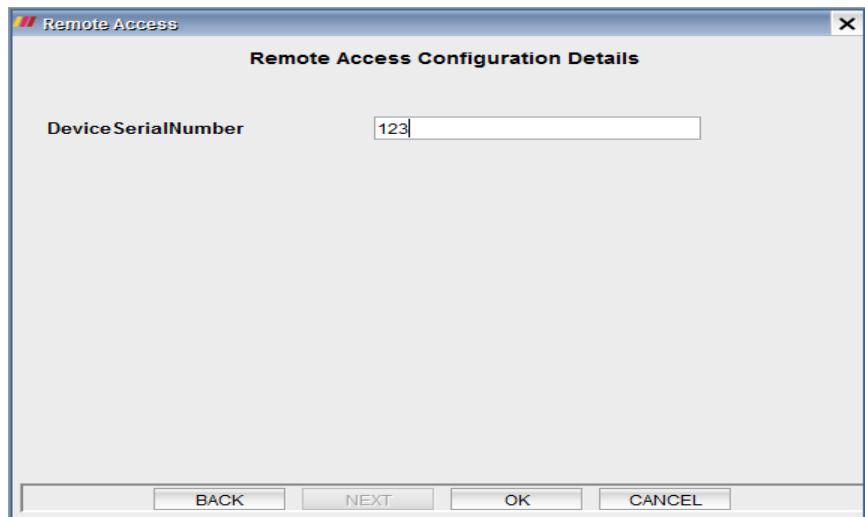
```
Microsoft Windows [Version 10.0.19043.1466]
(c) Microsoft Corporation. All rights reserved.

C:\Users\bharathik>Pip install neoradio2 --user
WARNING: pip is being invoked by an old script wrapper. This will fail in a future version of pip.
Please see https://github.com/pypa/pip/issues/5599 for advice on fixing the underlying issue.
To avoid this problem you can invoke Python with '-m pip' instead of running pip directly.
Requirement already satisfied: neoradio2 in c:\users\bharathik\appdata\roaming\python\python36\site-packages (1.2.0)
Requirement already satisfied: pybind11 in c:\users\bharathik\appdata\roaming\python\python36\site-packages (from neoradio2) (2.8.1)
```

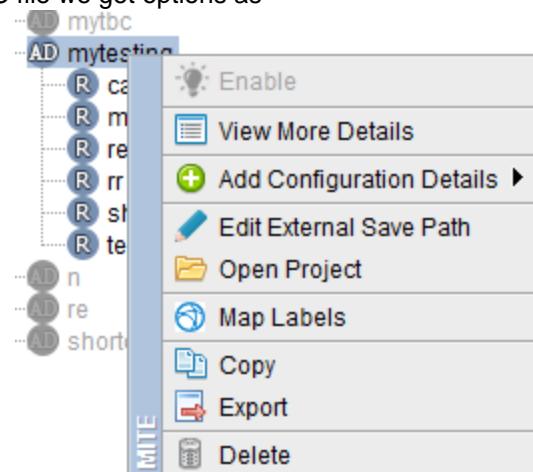
Go to the Test Script view. Then, in the configuration window, select TBC. Add a TBC file by right-clicking the Add configuration or by importing the tbc. A file with tbc name is added to the tree. Right click on the file and select Add Configuration Details. Select Remote Access after clicking on Add Configuration Details. Provide a short name and choose Relay-Controls as the tool interface. By default, neoRAD-IO-2-PWRRLY will be selected.



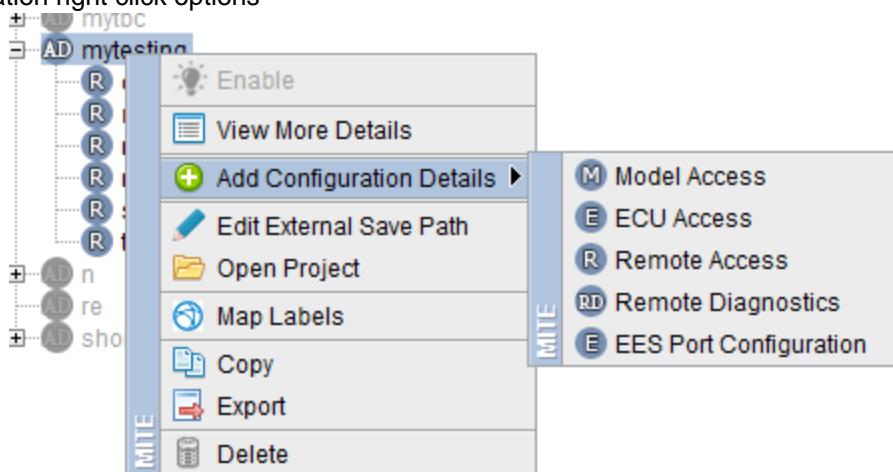
After giving all the fields click on next and frame with Device Serial Number will be loaded, provide the device serial number per user requirement and click on ok.



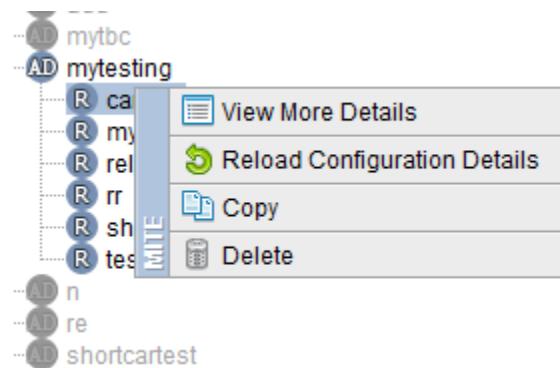
Right click on the parent TBC file we get options as



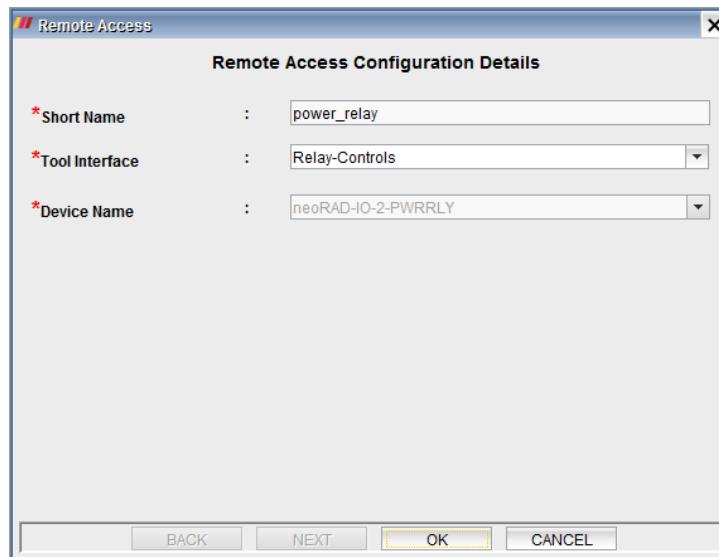
Add configuration right click options



Upon right click on child TBC we get options as



User can access all the options as required. Whenever user wants to reload the existing TBC configuration, the click on Reload Configuration Details and frame named Remote Access Configuration Details will be opened.

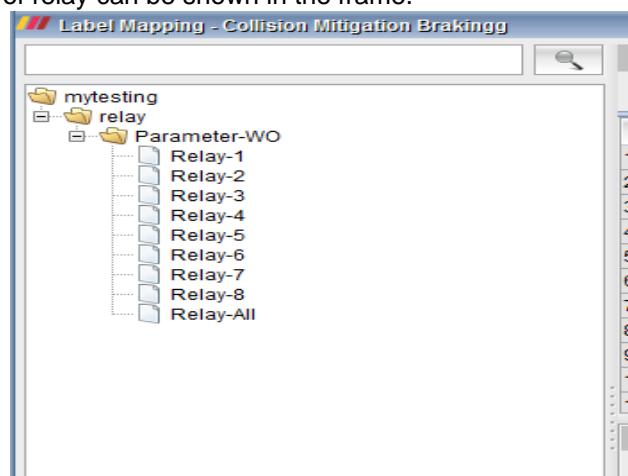


For testing the relay control the user needs to provide the test case, action as "Set" and service type as "ECU Input" user can write any own description in the desired Expected value user can write the value based on the relay setting.

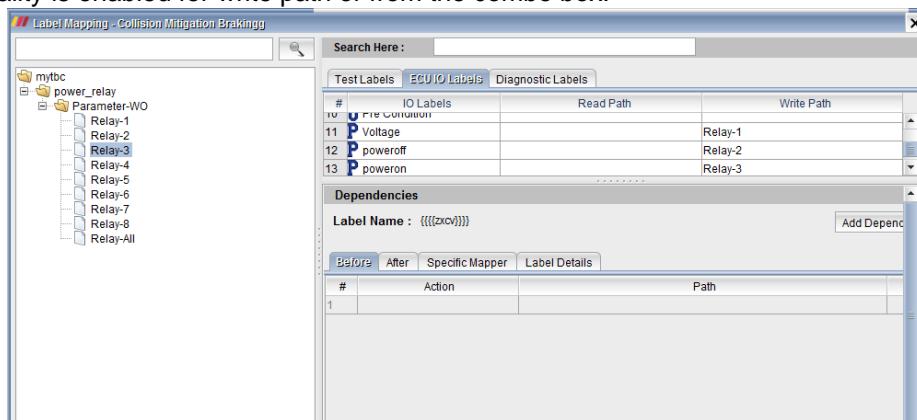
- User can write the integer value that will be from 1– 255.based on the value given the respective relays will be perform the operation.
- Zero (0) will be used to off the relay.
- They can write the respective hex value also.

And map the required labels in label mapping frame.

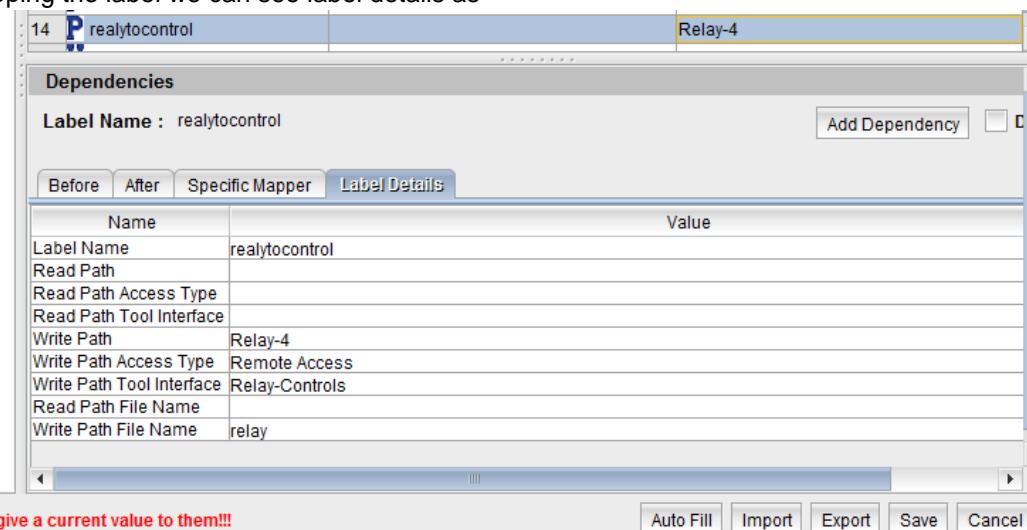
To map the labels right click on parent TBC file, then select Map Labels, a frame with label mapping will be opened. The parameters of relay can be shown in the frame.



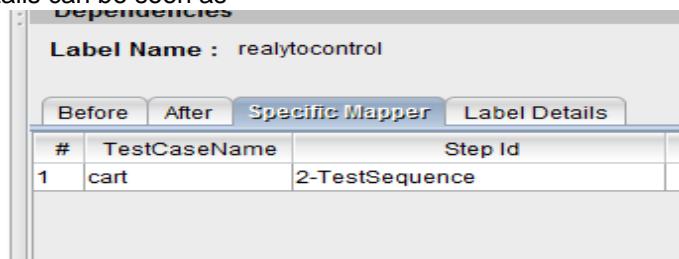
Based on the user labels will come to label mapping frame, user can map the labels using the drag and drop functionality is enabled for write path or from the combo box.



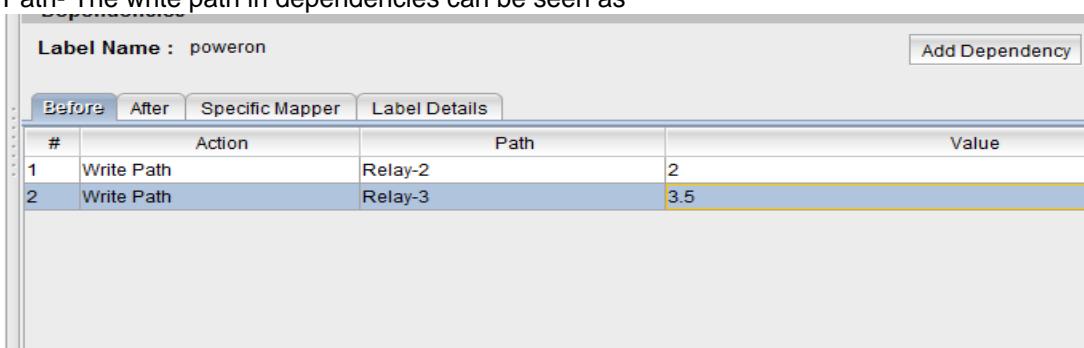
After mapping the label we can see label details as



The specific mapper details can be seen as



Before Path- The write path in dependencies can be seen as



After Path- The write path in dependencies can be seen as

Dependency Details			
#	Action	Path	Value
1	Write Path	Relay-1	6
2	Write Path	Relay-5	5.2
3			

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

19.1.13 SCPI Power Supply

Power Supply on SCPI

Power Supply on SCPI means it will work for all the new Power Supplies which is having SCPI compatibility.

Ex: Tenma

Installation:

The user must install the PyVISA software in accordance with the version of Python that is installed on the PC (Otherwise we get an exception stating that no module of PyVISA) and that python must support the automation desk. (It is recommended to install Python 2.7 version)

How to Install PyVISA in Python (Automation Desk):

1. Navigate to System Properties and then click on Environmental Variables.
2. In user preferences, select PATH --> click on Edit. Within the window, check to see whether Python 2.7 is available, and if not, add it to the path. If there is a higher version of Python > 2.7, make sure you delete it.
3. In System Variables, select PATH --> click on Edit. Within the window, check to see whether Python 2.7 is available, and if not, add it to the path. If there is a higher version of Python > 2.7, make sure you delete it.
4. Restart the system and then try to install PyVISA.

Command to install the PyVISA in python:

pip install pyvisa-py

(or)

pip install PyVISA

(or)

pip install -U pyvisa

- This feature Power Supply on SCPI is available in the TBC Remote Access configuration
- To use this feature we need to follow below steps for the configuration

Note: In TBC both the power supplies, BK Precision 1697 and SCPI Power supplies, cannot be used at the same time (Either any one of RS-232 /SCPI related power supply can be configured in one TBC at a time). It will work for the power supply BK precision 920x and SCPI.

System Requirements:

1. NI Max with VISA application
2. Python

Please download the Ni software from below link

<https://www.ni.com/en-in/support/downloads/software-products/download.package-manager.html#322516>

Ni version 21.6

If you face any issue from above link then try to download from This link

<https://www.ni.com/en-in/support/downloads/software-products/download.labview.html#443865>

after click on the link—Select Install offline –Click on download option.

Included Versions

2021 SP1

➤ **Supported OS**

➤ **Language**

➤ **Checksum**

DOWNLOAD

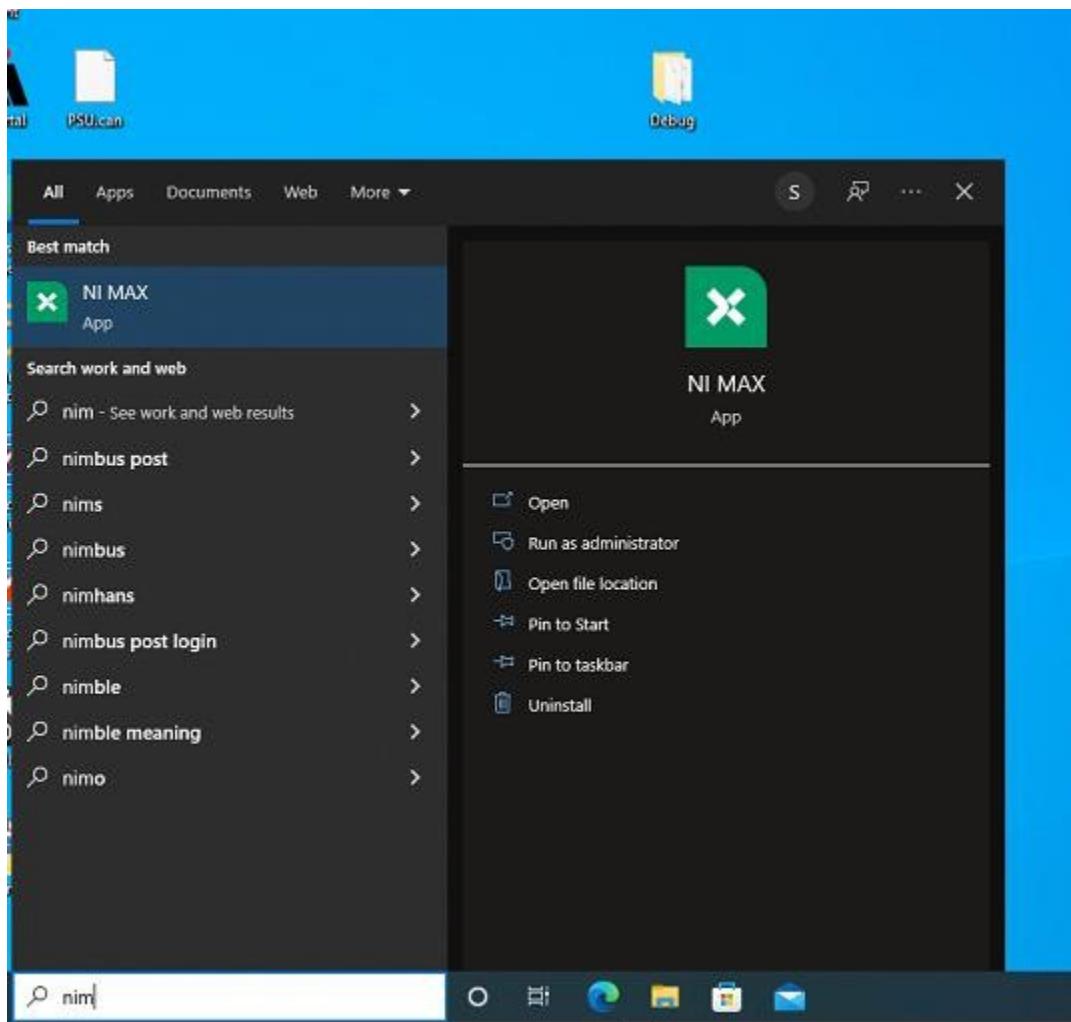
INSTALL OFFLINE

File Size

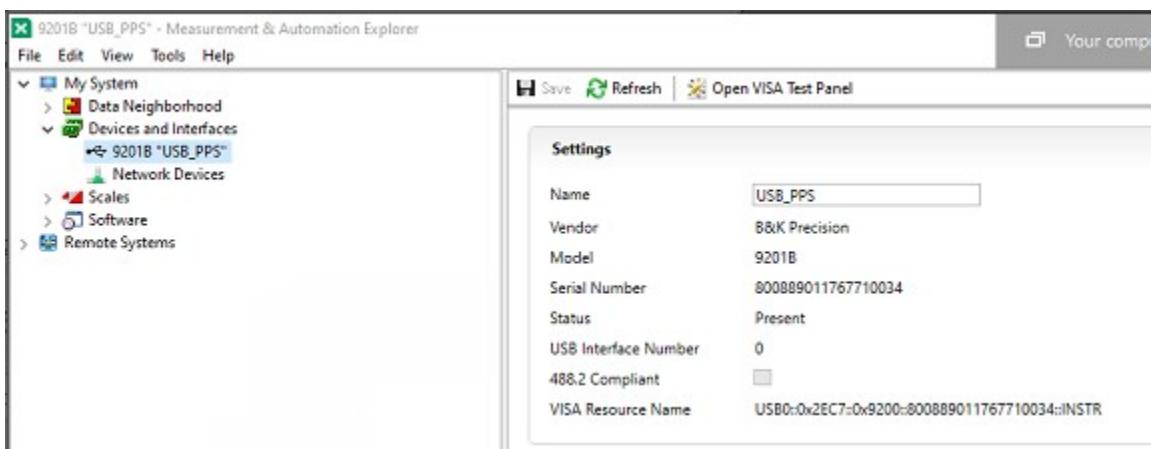
6.46 MB

Setup Procedure:

1. Connect the PPS via USB port
2. Set Connectivity as UBS through PPS menu
3. Open NI MAX application



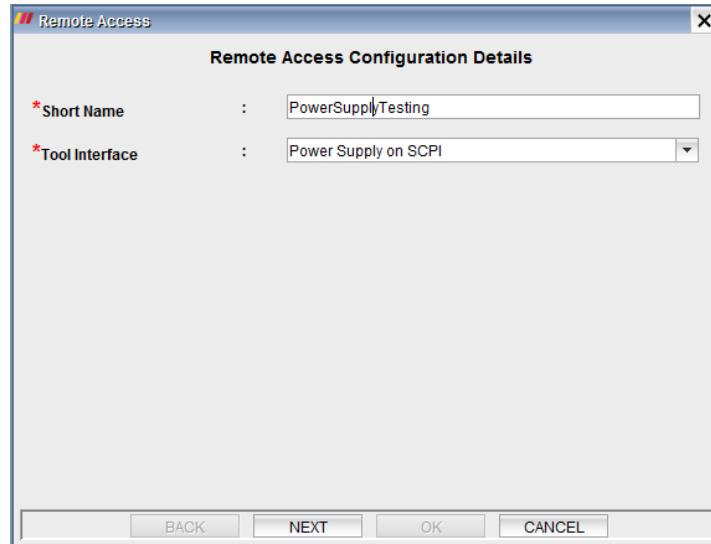
4. Provide the device name to “Example :USB_PPS” as shown below



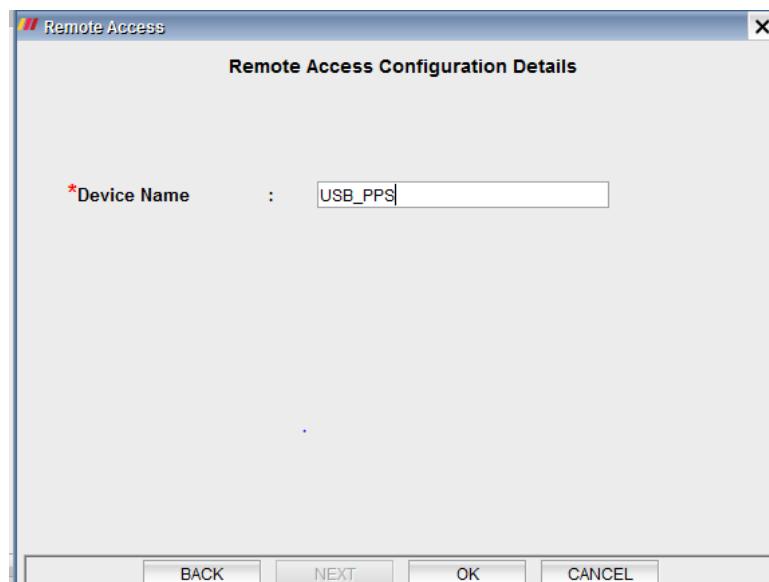
Scenario:

Step 1: Go the test bench configuration in the test script view set; add the configuration details in Remote access by giving the Short Name and Tool Interface (Power Supply on

SCPI)



Step 2: Give the Device name as per the name in the NI MAX application.



Step 3: User needs to do the label mapping for the ECUIO Labels in map label frame

#	IO Labels	Read Path	Write Path
1	P current	RS232-Check Current	RS232-Set Current
2	P power_cycle	RS232-PowerCycle	RS232-PowerCycle
3	P power_off	RS232-PowerSupply-OFF	RS232-PowerSupply-OFF
4	P power_on	RS232-PowerSupply-ON	RS232-PowerSupply-ON
5	P voltage	RS232-Check Voltage	RS232-Set Voltage

Dependencies

Label Name : voltage

- After the label mapping is done, user can execute the scripts and create the project creation in the Automation desk.

Created with the Personal Edition of HelpNDoc: [Free CHM Help documentation generator](#)

19.1.14 Export/Import TBC

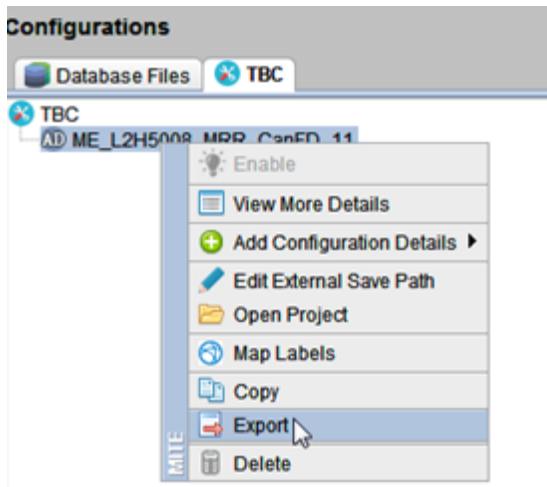
A. Steps to Configure TBC:

- Right-Click on TBC
- Select the option –“ Add Configuration”



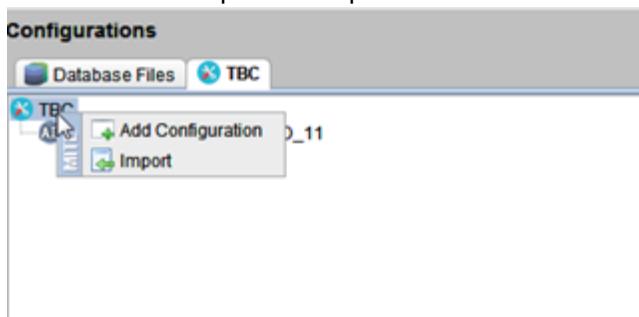
B. Steps to Export TBC:

- Right-Click on TBC created
- Select the option –“ Export”



C. Steps to Import TBC:

- 1.Right-Click on TBC
- 2.Select the option –“ Import”



Note:- In Copy/Paste of TBC from one bench To another bench user will not get the reference file of Automation project(Videos, Images),
User need to Do label Mapping again.

To over come

1.User can copy mapper File of Label Mapping (Which is available in label mapping frame using Export/Import option) from one bench to another Bench and save the Label mapping

Or

2.User need to Do Flush in test script Role and once it done Then user can take Reload test script from PTC operation in MITE in the second bench.

Created with the Personal Edition of HelpNDoc: [Free PDF documentation generator](#)

19.1.15 Image Comparison

Image Comparison

Image comparison:

We have to compare each and every particular image using some particular coordinates whether it having same coordinates in the same image or not.

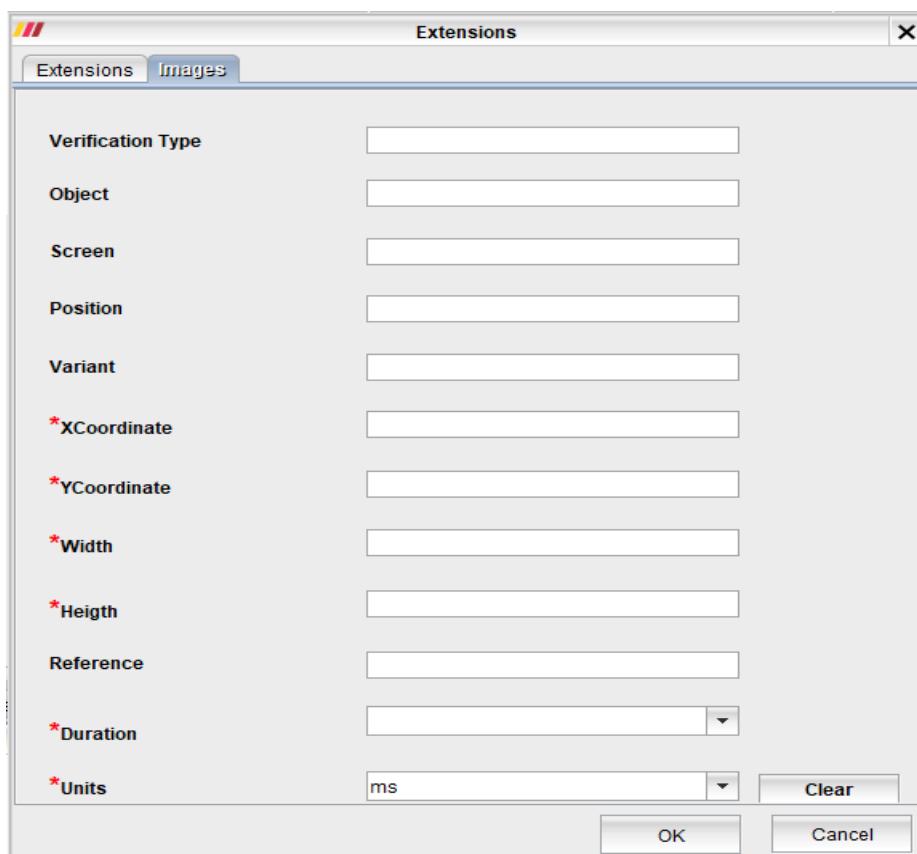
To use the image comparison function, In the MITE test case editor, write/include the test step as follows:
Here is an example test case that shows how to write test step.

#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	Additio
#	Pre Conditions					
#	Test Sequence					
1	Run	Video	video	carmaker_videos		
2						
#	Post Conditions					

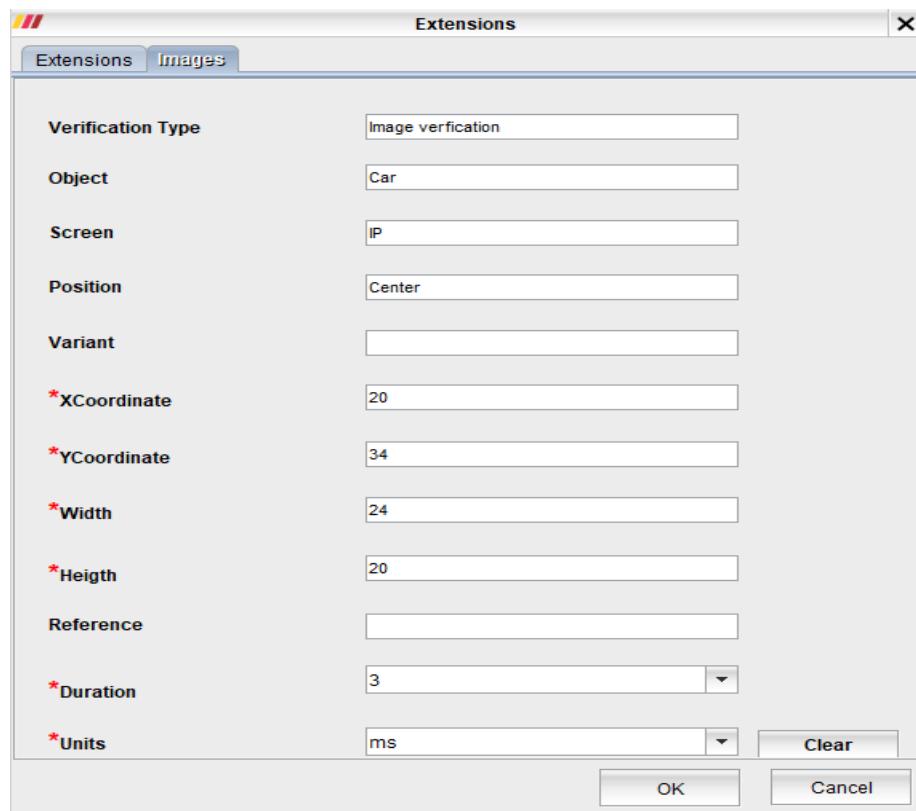
**Action Parameter/ServiceType
Parameter/Description Desired/Expected Value**

Check Image video front top all

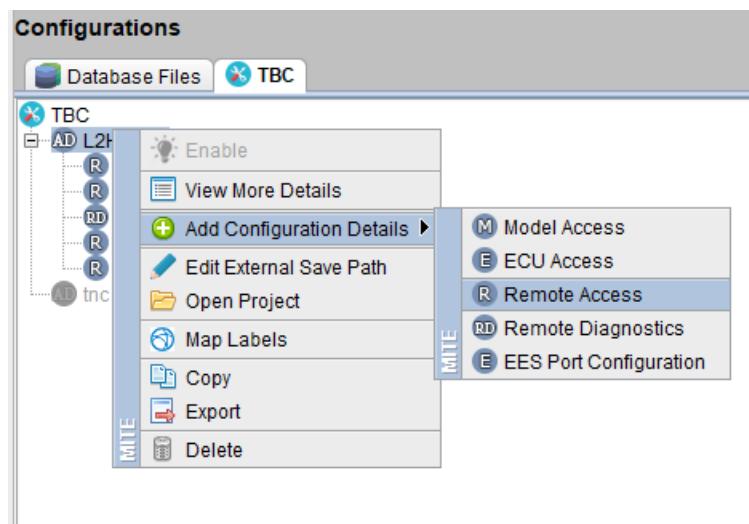
User can double click on the extensions block if they want to check or write the coordinates for a particular image. The Extensions window will appear; choose the Images Tab there.



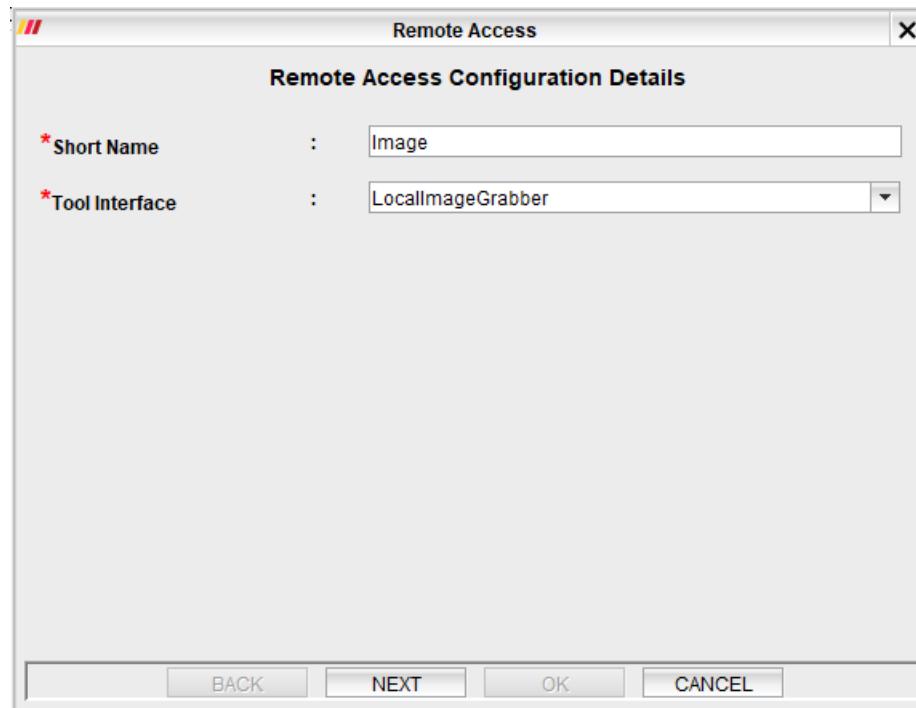
We can see that a list of fields will be displayed. The user must correctly fill those data, and the fields with an asterisk (*) must be filled. Once all details filled the click on OK to save



- Choose a test suite. Change the role to the test script role. Go to the configuration window and choose TBC. Select Add configuration details, then select Remote Access from the context menu.



- Remote Access Configuration window will be opened. In that provide the Short Name & Tool Interface as LocalImageGrabber and click on Next.



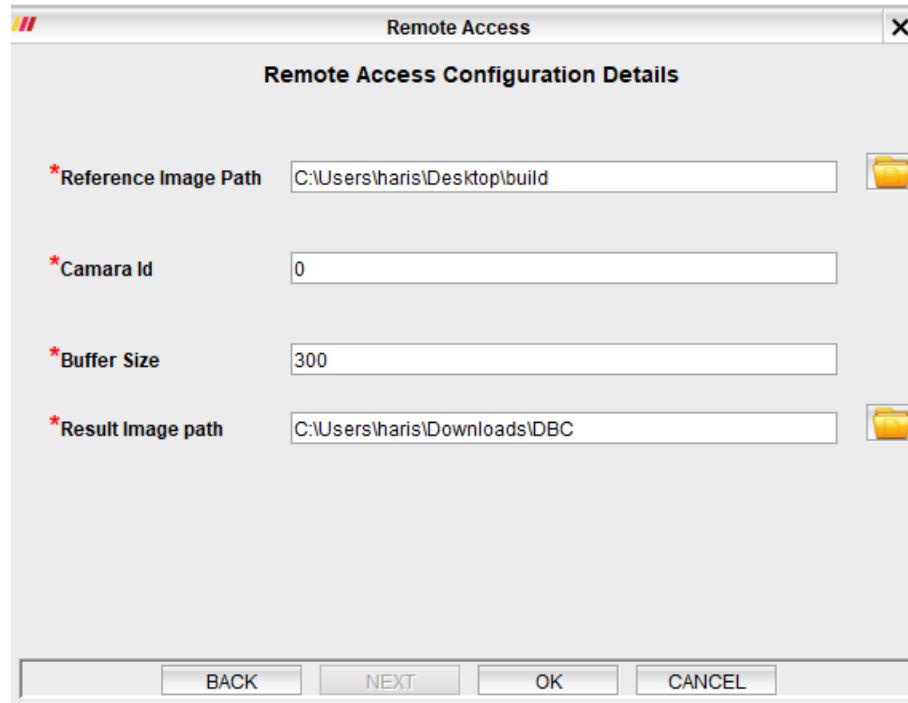
- Provide the reference image path, camera id, Buffer Size, Result Image Path.

Reference Image Path: Path of reference images

Camera Id: Provide the device id. (For example if PC is connected with two cameras 1 & 2, then camera 1 is noted 0 and camera 2 is noted as 1)

Buffer Size: 300 approx. (not mandatory)

Result Image Path: Whatever the images captured while execution those are back up in the provided location and once all fields are provided click on OK.



- Next do the label mapping by selecting the appropriate labels and click save.
- Then do the script generation and project creation and execute the test case in the automation desk.

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

19.2 Configuration Enable – Disable

User can choose between the added configurations using “Enable – Disable” on right click options as shown below

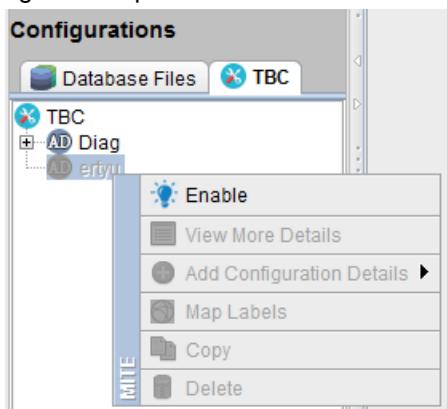


Figure 83: configuration Enable-Disable

Created with the Personal Edition of HelpNDoc: [Easily create iPhone documentation](#)

19.3 Label Mapping

MITE shall import those files and create tool label database which later can be mapped to test labels used in test cases and create a configuration file.

1. To view available labels in test suite , Right click on Configuration node and Select “Map Labels”

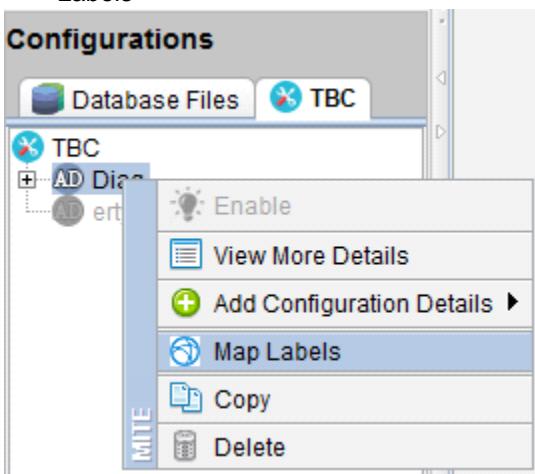


Figure 84: To map a label

2. Label Mapping window appears as below :
- Left side tree defines all the Label data base files imported in test suite
 - Test Labels Tab defines list of labels used in the test suite
 - Dependencies section allows user to add dependency labels if any otherwise can uncheck the option

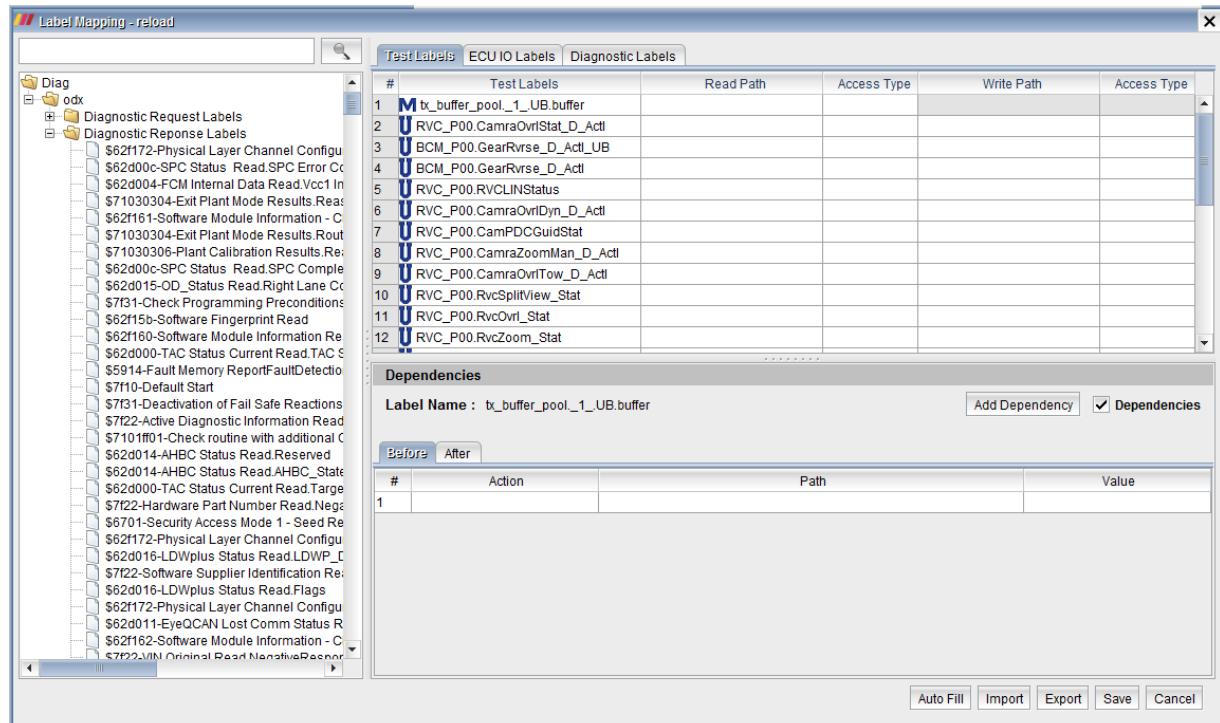


Figure 85 : Label Mapping Window

3. Mapping Labels can be done using “Drag and Drop” action
 - On drag and drop from tree to table gives “write path and read path” as shown below

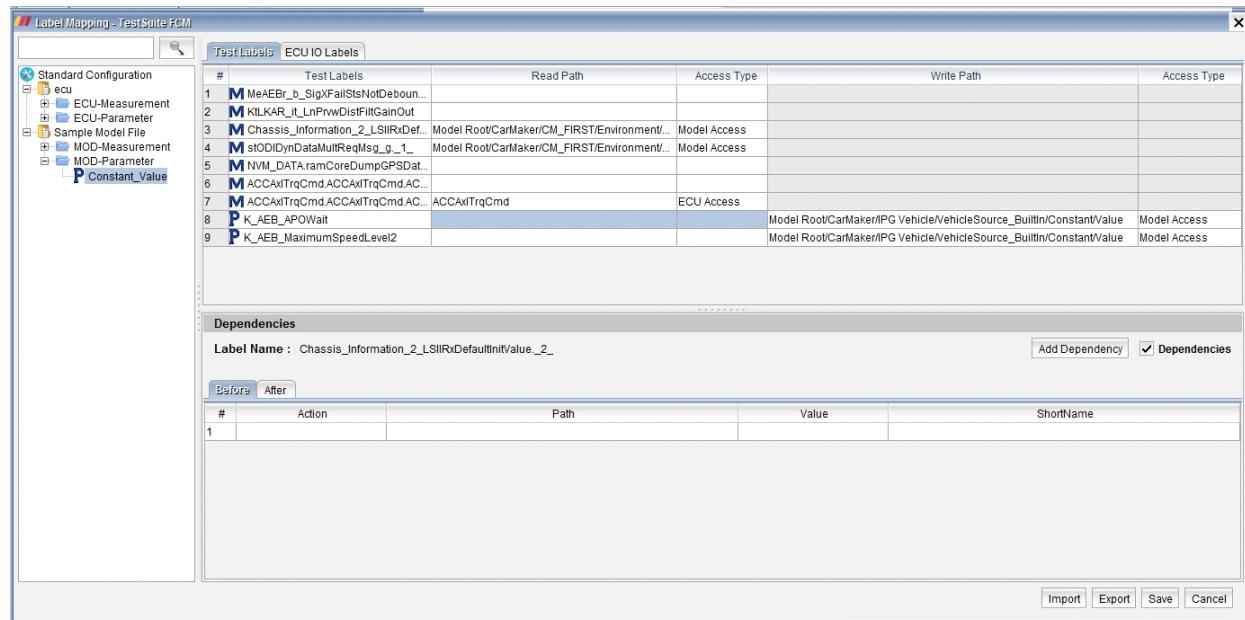


Figure 86: Read/Write Path in label Mapping

4. Dependency Label Mapping

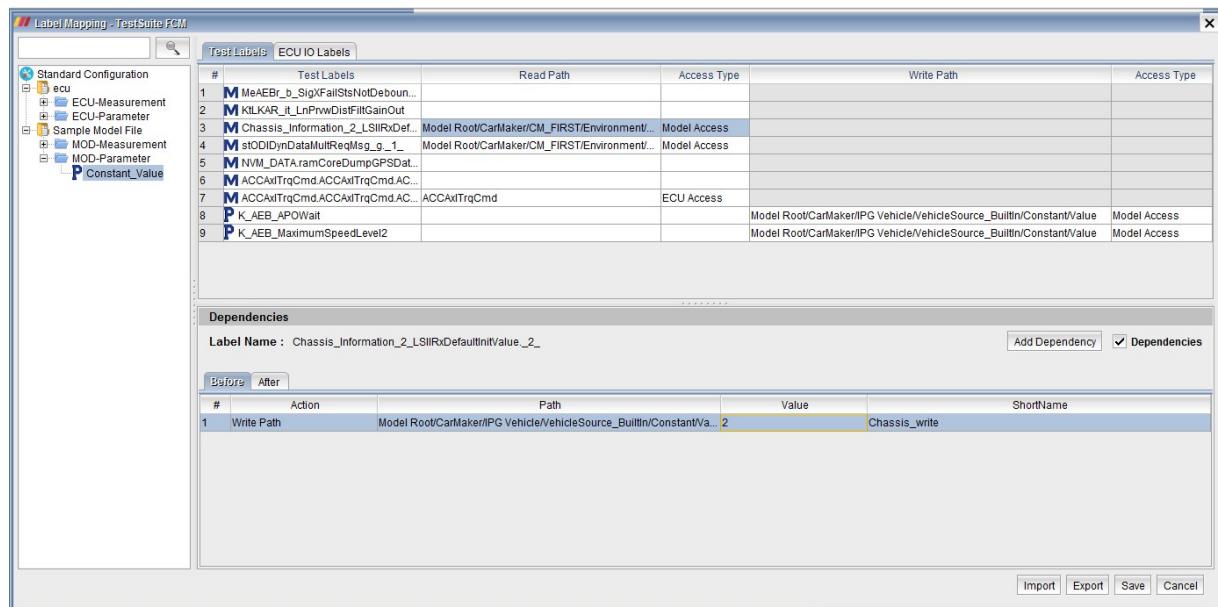


Figure 87: Dependency Label Mapping

5. Label Filtering

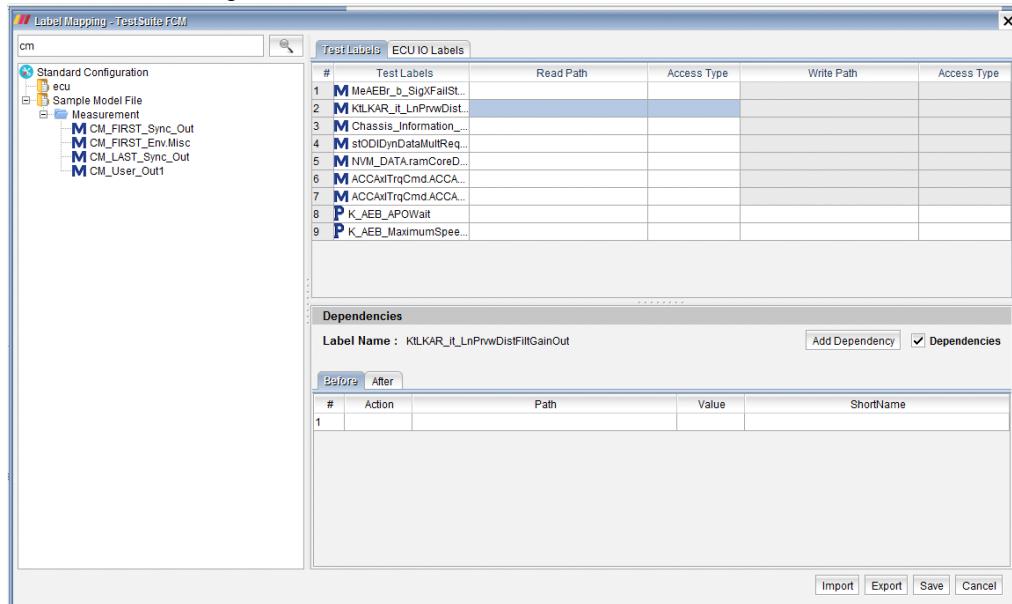


Figure 88: Filtering Label

6. Label Mapping file Import/Export Feature

This feature is to import or export a Label mapping file which is already mapped or reuse of mapper files. After mapping respective read paths and write paths to all the labels, the data can be exported and imported in another configuration.

The below is the screenshot of the frame before mapping:

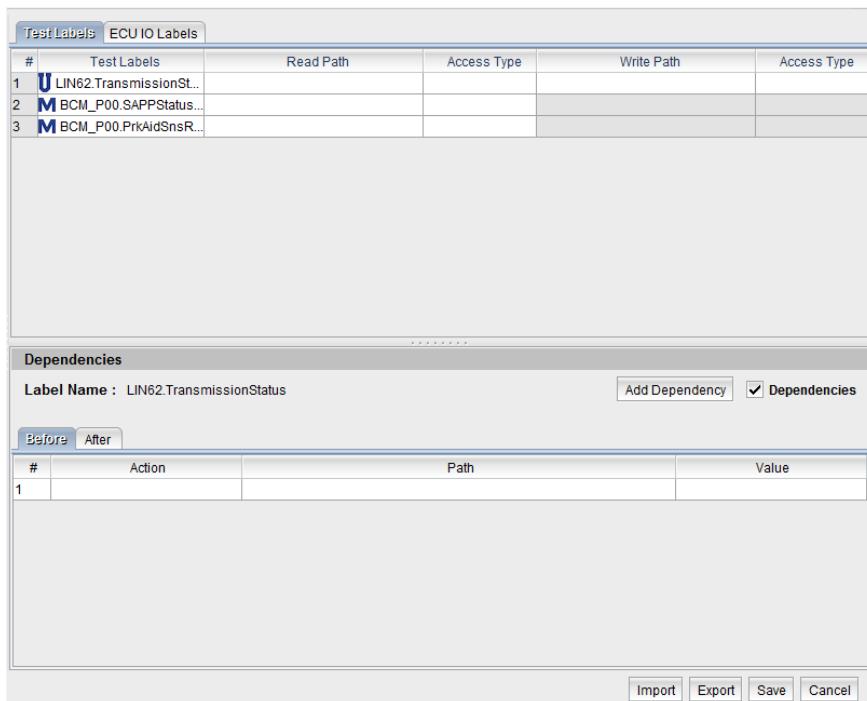


Figure 89: Before Mapping Label

Now, Click on import and give the exported label mapping file as zip. After importing the zip, the data of the current mapping tables gets overwritten by the label mapping from the zip file for the matched labels.

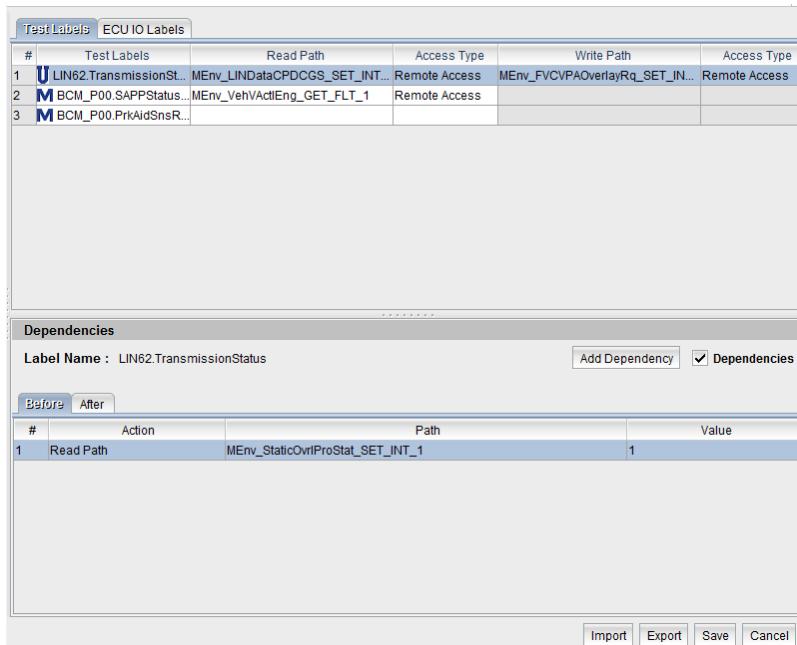
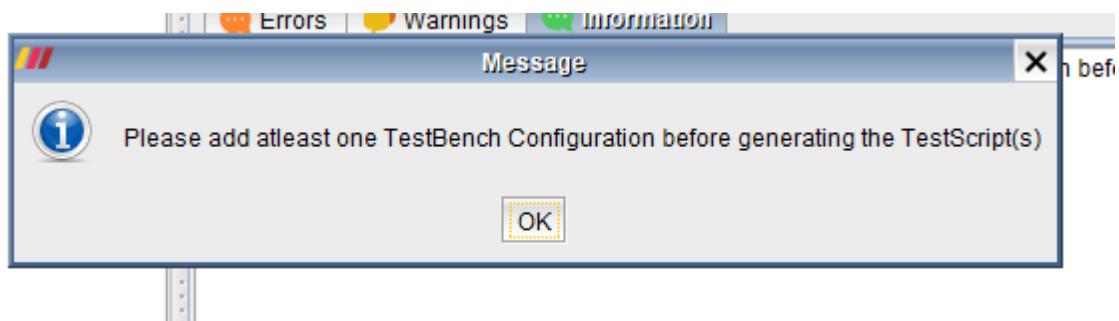


Figure 90: After Mapping Label

Created with the Personal Edition of HelpNDoc: [Easily create Help documents](#)

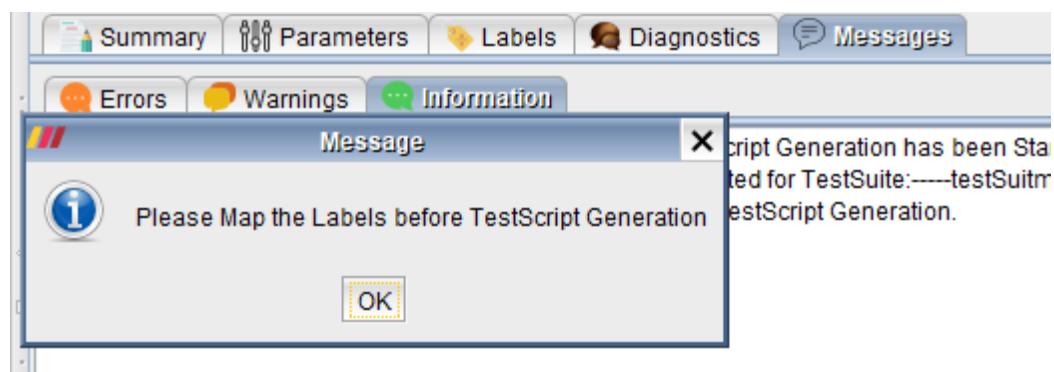
19.3.1 MAP Labels

Step 1:-If User is having Test Cases and shifts to script level at that time he/she have to add TBC configuration files otherwise will get following popup:



Step 2:-If there is any label mapping file and are not mapped in script level then at that time of script generation

User will get the following pop message .



Step 3:-After mapping the Label user can generate the script .

Created with the Personal Edition of HelpNDoc: [Free Web Help generator](#)

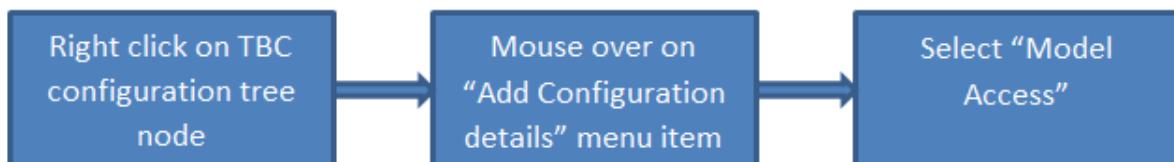
19.3.2 TBC Frames MAP Labels

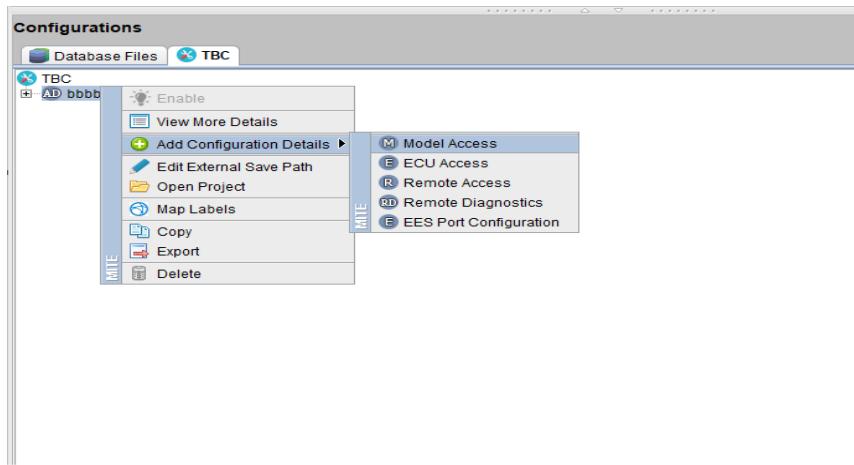
In TBC we have multiple frames,
Those are:

1. Model Access
2. Remote Diagnostics
3. Remote Access

1.MODEL ACCESS:

1.1 To Open Model Access Window

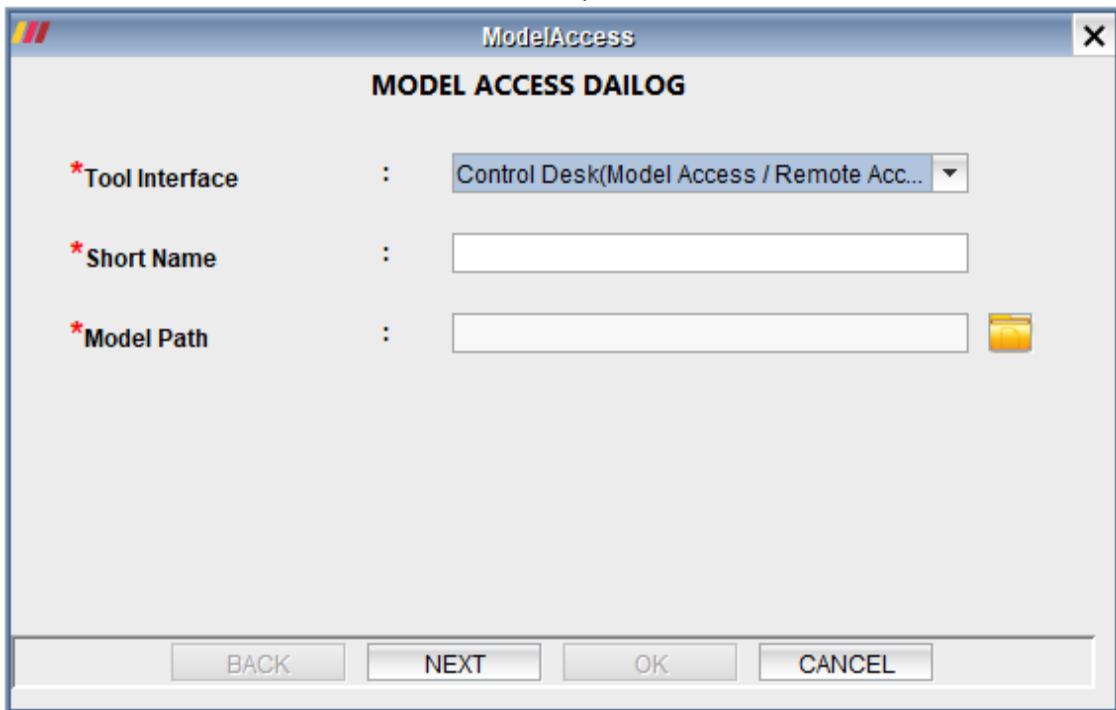




Open model Access

1.1 Model Access window :

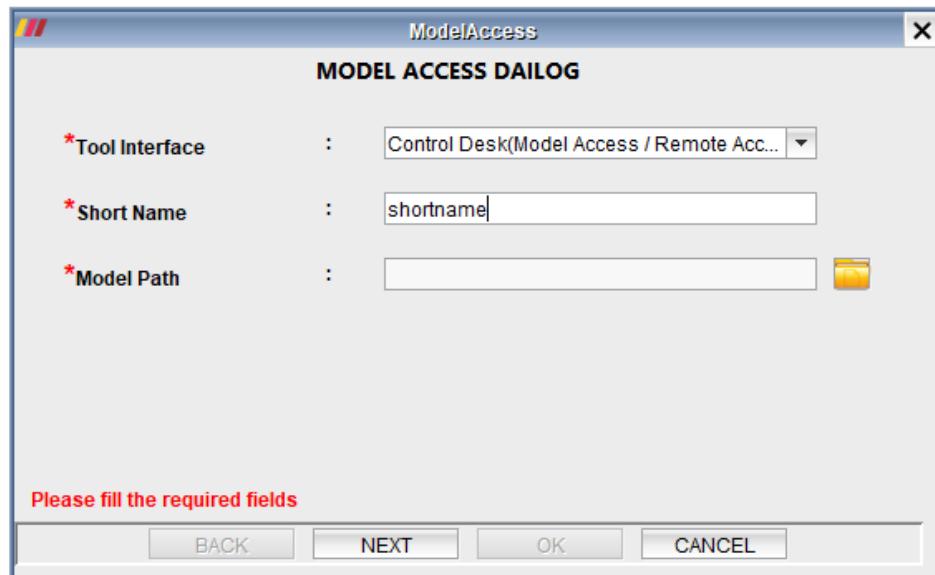
After click on model access menu item, a window will open. That window is Model Access window.



IMG: Window with the default selection of Control desk model in tool interface.

1.1 "Please Fill Required Fields" Validation:

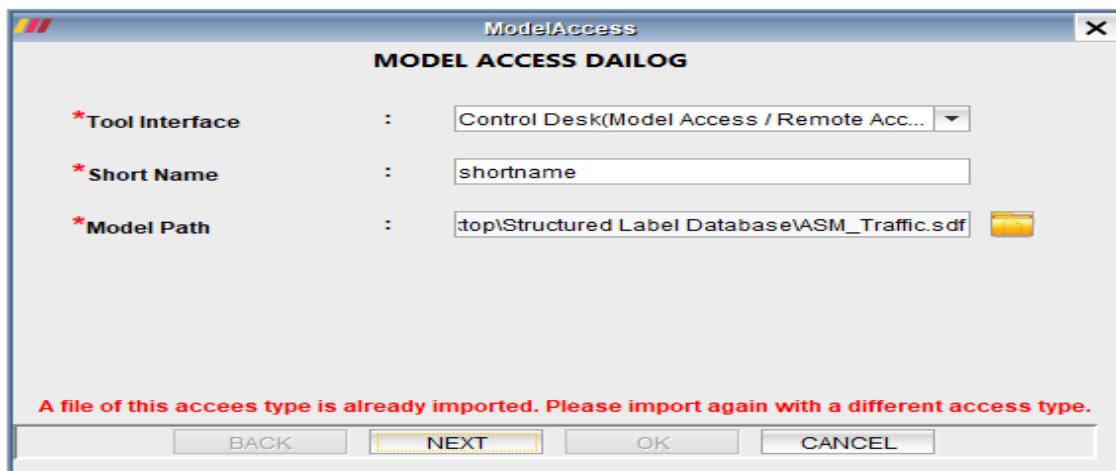
If required fields are empty while submit data using next button or ok button then "Please fill required fields" validation message will arrive on screen.



IMG: “Please fill required details” validation message

1.1 Model PAth Alreadu Imported condition:

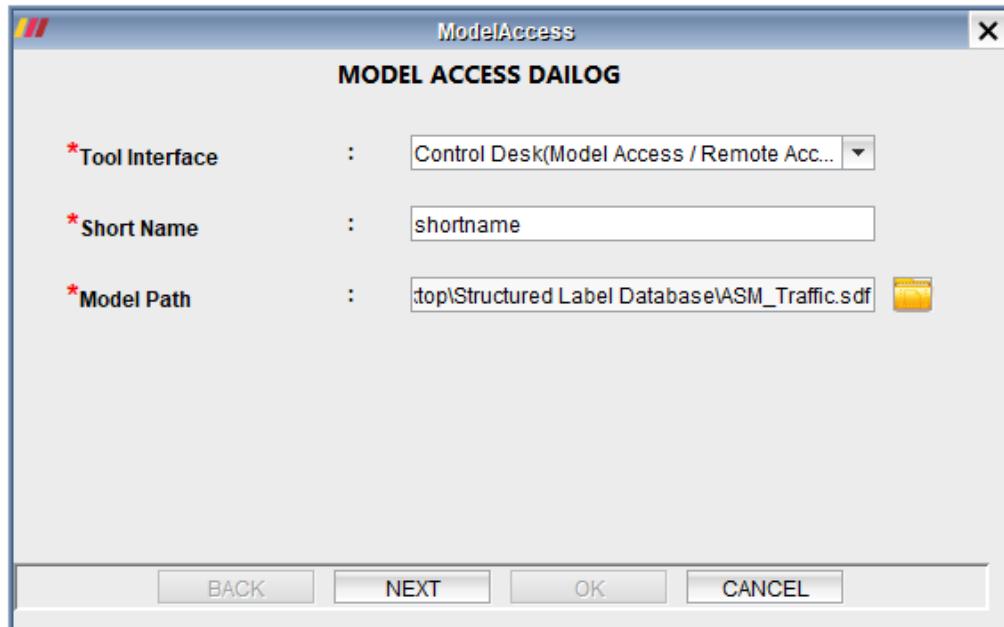
If current adding configuration details already existed then it will not allow duplicates. So a validation message will appear on screen as like below image.



IMG: Model path already imported condition.

1.1 First Panel with Complete details

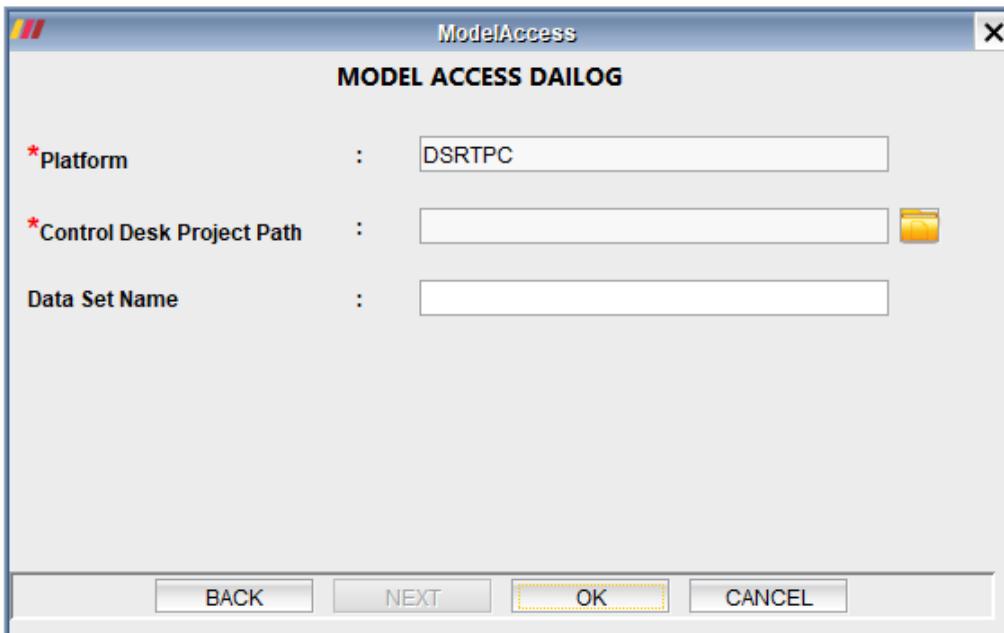
If we want to move from first panel to second panel, we need to fill complete details in first panel without violate all conditions.



IMG: First panel with complete details.

1.1 Second Panel with Model Access

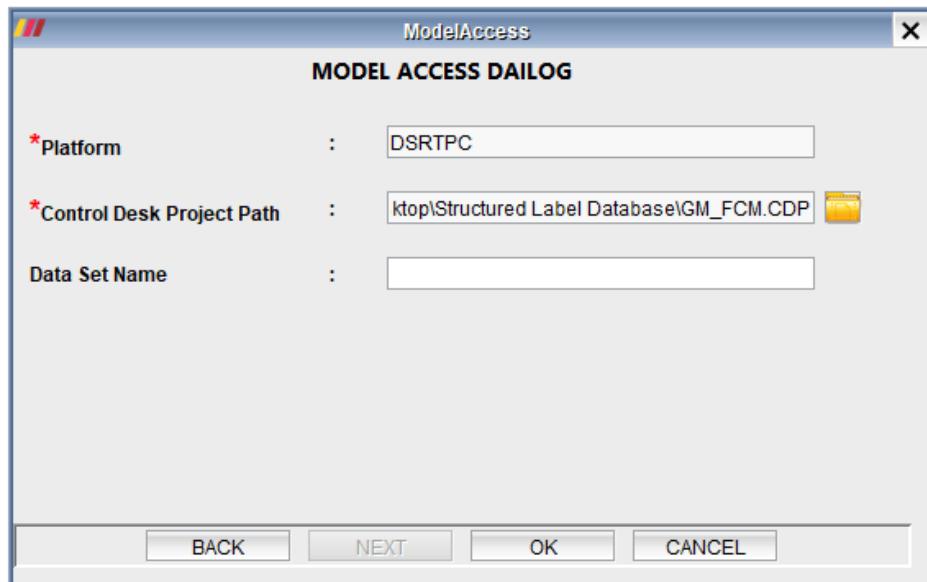
While click on next button in first panel, if we fill all details in first panel then it will move to second panel. That second panel as like below.



IMG: Model access dialog with second panel.

1.1 Second panel with complete details:

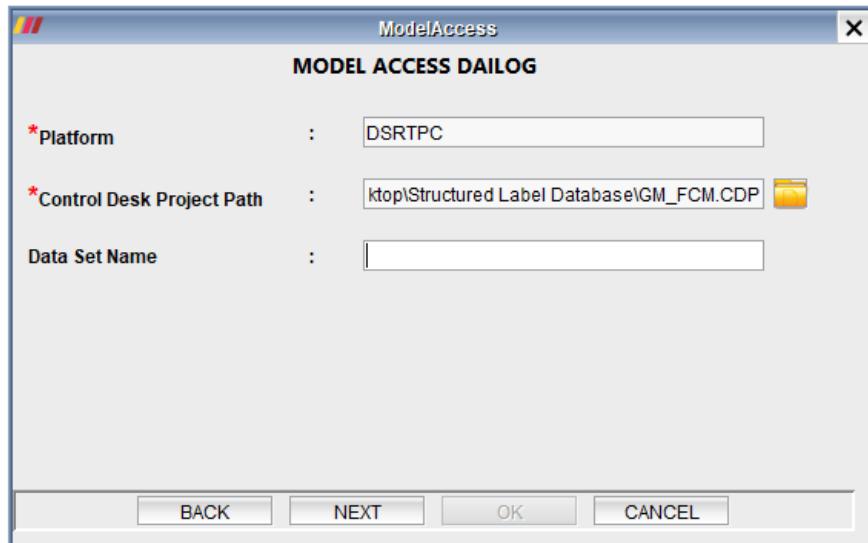
Second panel in model access with complete details as like below. If tool interface selected value is "Control Desk (Model Access / Remote Access)" then next button will be disabling and Ok button will be enabling in second panel.



IMG: Second panel with complete details.

1.1 Second Panel with Enable Next button:

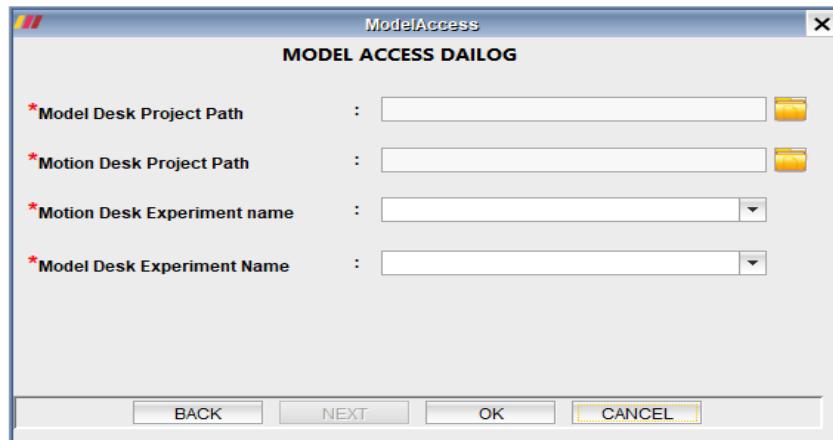
If selected **Tool interface** is “Control Desk/Motion Desk/Model Desk” then **next button** will be enabling in second panel. If we click on that next panel in second panel then it moves to third panel.



IMG: Second panel with enabled next button

1.1 Third panel with Disable OK button:

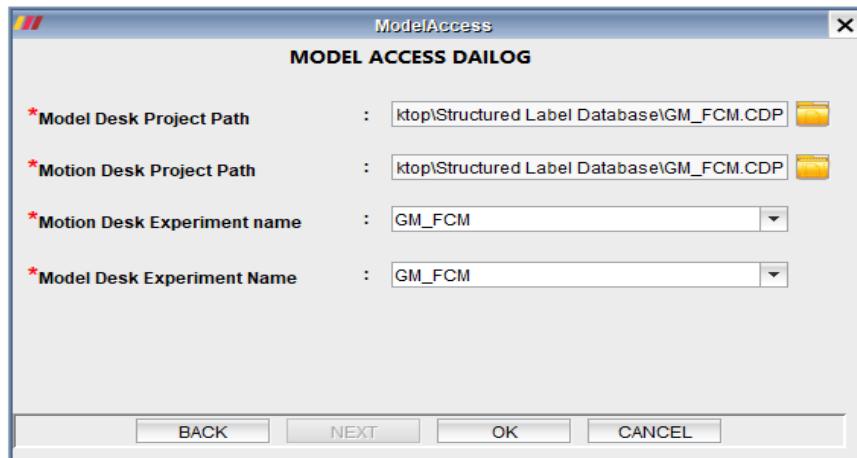
While click on next button in second panel at that time we can see third panel of Model access dialog. Third panel of Model access as like below.



IMG: Model access third panel

1.1 Third Panel with Complete Data:

Before we click on ok button in third panel we need to fill complete fields what we have in third panel. Otherwise validation will invoke.



IMG: Model access third panel with complete data

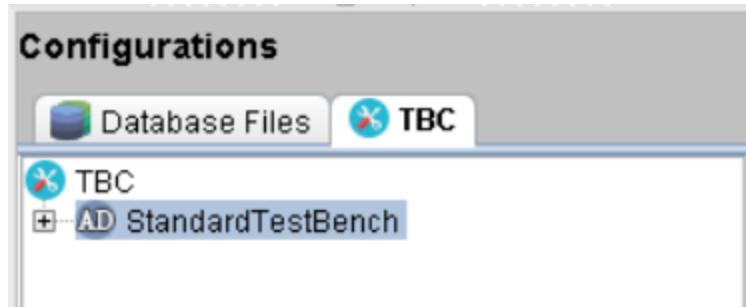
Created with the Personal Edition of HelpNDoc: Produce electronic books easily

19.3.3 LabelMapping based on the selected TestCases and their dependent TestCases Selection & Search ,Alphabetically order of Map labels

Map Labels For Selected Test Case

After selecting a particular Project from the Project Navigator, a list of flushed suites and unflushed suits are displayed.

Select a suit from the Project Tree and switch the environment from Test Case to Test script.In the Configuration window, under the TBC select a file from the list of TBC file.



Selection of a TestCases:

As soon as you select a file in the configuration window, a list of test cases is displayed in the summary.

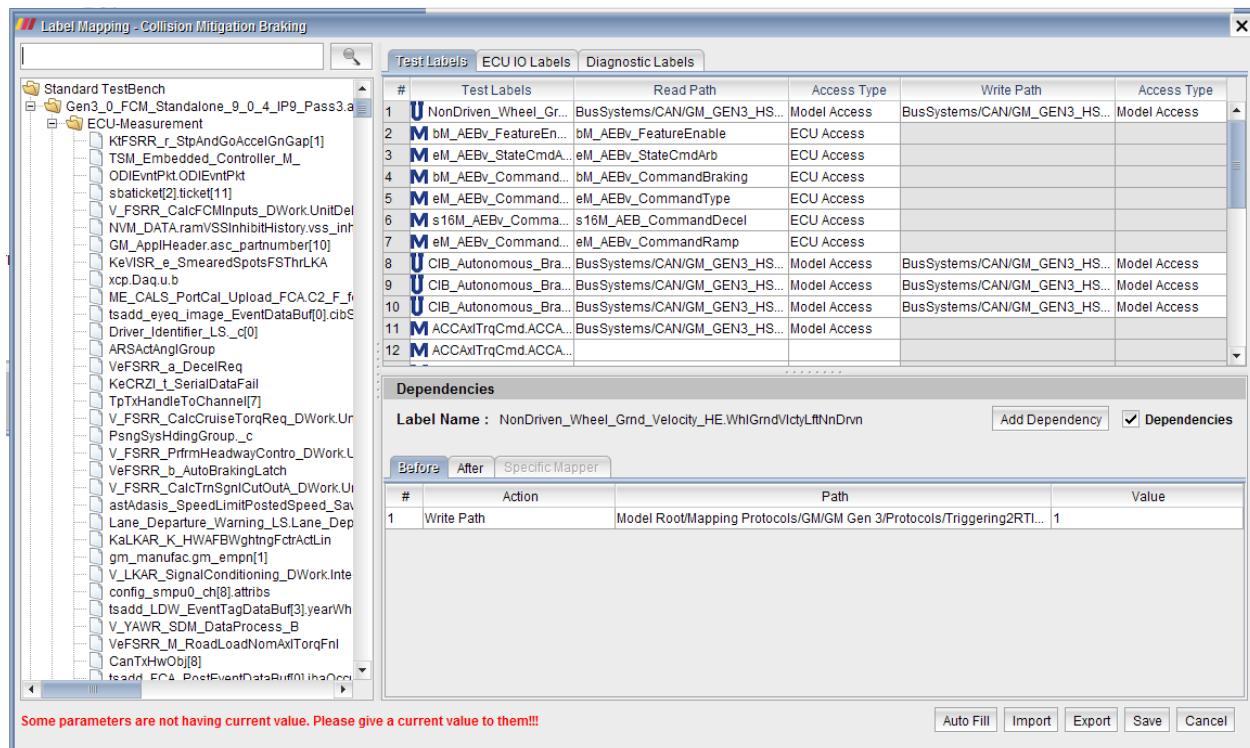
#	All	TestCase ID	Category	Description	Script Name	Automation Status	Shared Id	Shared TestSuite Id	Shared TestS...	Valid For Proj...	Priority
1	<input checked="" type="checkbox"/>	7502733	System Test	To test the camera with invalid configuration with Vss sleep Disabled (Dynamic_Available ...	TestScript-7502732-1529...						
2	<input type="checkbox"/>	7502734	System Test	To test the camera with invalid configuration with Vss sleep is Enabled that is Dynamic_Av...	TestScript-7502734-1530...						
3	<input type="checkbox"/>	7502736	System Test	To test the camera with invalid Configuration with Vss sleep is Enabled (invalid overlayset)	TestScript-7502736-1530...						
4	<input type="checkbox"/>	7502738	System Test	To test the camera with invalid Configuration with Vss sleep is Disabled (invalid overlayset)	TestScript-7502738-1530...						
5	<input checked="" type="checkbox"/>	7502740	System Test	To test the camera with invalid Configuration with Remote start status equal to 1 and Vss s...	TestScript-7502740-1530...						
6	<input type="checkbox"/>	7502742	System Test	To test the camera with invalid Configuration with Remote start status equal to 1 and Vss s...	TestScript-7502742-1530...						
7	<input type="checkbox"/>	7502744	System Test	To test the camera with valid Configuration with Vss sleep is Enabled.	TestScript-7502744-1530...						
8	<input checked="" type="checkbox"/>	7502746	System Test	To test the camera with valid Configuration with Vss sleep is Disabled.	TestScript-7502746-1530...						
9	<input type="checkbox"/>	7502748	System Test	To test the camera with valid Configuration with Vss sleep is Present and remote start stat...	TestScript-7502748-1530...						
10	<input type="checkbox"/>	7502750	System Test	To test the camera with valid Configuration with Vss sleep is Enabled and channin GearR...	TestScript-7502750-1530...						

Select a particular TestCase from the list of TestCases present and then in the configuration window navigate toTBC file.

Select Standard TestBench, a list files are displayed under it. Right click on Standard TestBench,

Select Map Labels.

A pop-up window is displayed where Label Mapping of a particular selected TestCase is displayed.



Where you get the information of TestLabels, ECI IO Labels, Diagnostic Labels of a selected file. If the user requires Label mapping of Multiple TestCases, then the user can select Multiple TestCase at a time. If the user has not selected any particular TestCase, then Label Mapping information of all the test cases are displayed.

Under this case, some labels might be present in different test cases.

Filter Option:

There is a filter option present in the pop-up window present on the top. If a user wants to access the Label Mapping information of a particular suit, the user can use **Search Bar** present in the Label Mapping pop-up window. All the detail related to the Label Mapping of that suit are displayed.

Search Here :					
Test Labels		ECU IO Labels	Diagnostic Labels		
#	IO Labels	Read Path	Access Type	Write Path	Access Type
1	P Battery_Voltage				
2	U Standard				

When the user enters the required Test Labels in the search bar, then all the Test Labels related to the requirement are highlighted in yellow are displayed.

Search Here :

Test Labels ECU IO Labels Diagnostic Labels

#	Test Labels	Read Path	Access Type	Write Path	Access Type
1	BCM_P00.GearRvrse_D...	MEnv_GearRvrseDActl_GET_INT_1	Remote Access	MEnv_GearRvrseDActl_SET_INT_1	Remote Access
2	BCM_P00.GearRvrse_D...	MEnv_GearRvrseDActlUB_GET_IN...	Remote Access	MEnv_GearRvrseDActlUB_SET_IN...	Remote Access
3	BCM_P01.CamraZoomMa...	MEnv_CamraZoomManDRq_GET...	Remote Access	MEnv_CamraZoomManDRq_SET...	Remote Access
4	BCM_P01.Remote_Start...	MEnv_RemoteStartStat_GET_INT_1	Remote Access	MEnv_RemoteStartStat_SET_INT_1	Remote Access
5	BCM_P01.Veh_V_ActlEng	MEnv_VehVActlEng_GET_FLT_1	Remote Access	MEnv_VehVActlEng_Set_FLT_1	Remote Access
6	BCM_P01.Veh_V_ActlEng...	MEnv_VehVActlEngUB_GET_INT_1	Remote Access	MEnv_VehVActlEngUB_SET_INT_1	Remote Access
7	BCM_P02.RVCCfgData0	MEnv_CfgData0_GET_INT_1	Remote Access	MEnv_CfgData0_SET_INT_1	Remote Access
8	BCM_P02.RVCCfgData1	MEnv_CfgData1_GET_INT_1	Remote Access	MEnv_CfgData1_SET_INT_1	Remote Access
9	BCM_P02.RVCCfgData2	MEnv_CfgData2_GET_INT_1	Remote Access	MEnv_CfgData2_SET_INT_1	Remote Access
10	BCM_P02.RVCCfgData3	MEnv_CfgData3_GET_INT_1	Remote Access	MEnv_CfgData3_SET_INT_1	Remote Access

Created with the Personal Edition of HelpNDoc: [Full-featured multi-format Help generator](#)

19.3.4 Autofill in the Label Mapping

AutoFill in Label Mapping Frame:

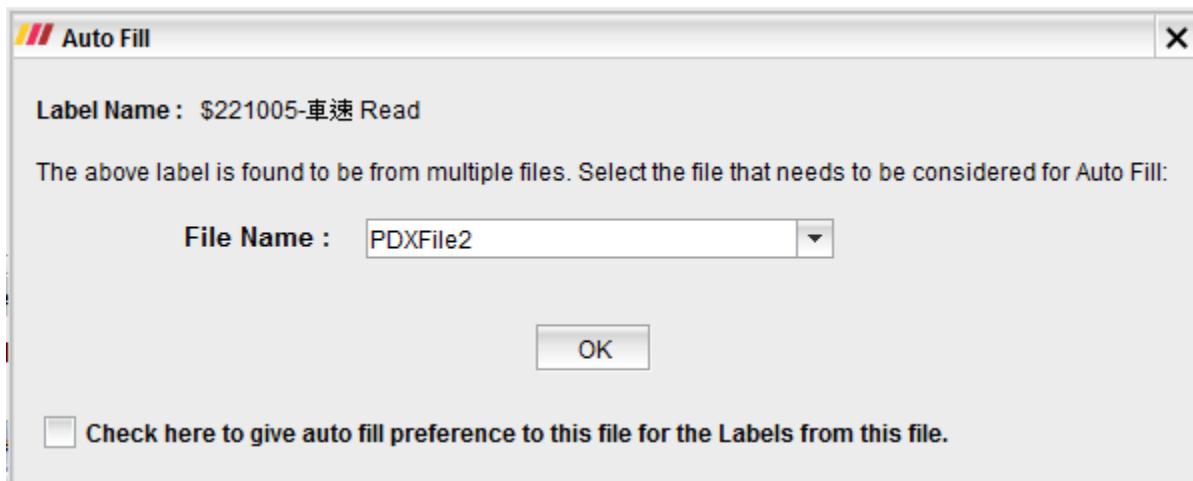
Description: Generally, label mapping is mapping the Test labels, IO labels and Diagnostic Labels with their corresponding read paths and write paths. Most of the times, there will be 100% percent matching between the labels from the Test cases and the Labels in the TBC. In those cases, MITE offers you this option of "Auto fill" in order to reduce the manual work to a greater extent.

In MITE, "Auto Fill" option is available for the labels from "A2L Files", "DBC Files", "ARXML CAN Files", "FIBEX CAN Files", "FIBEX LIN Files", "LDF Files" and "Diagnostic Files". These labels generally appear in Test Labels and Diagnostic Labels window. So, user can use the option of auto filling in these two windows.

Auto Fill option works in the following conditions:

- 1) When there is no read path or write path, then the path which is empty will be auto filled when user clicks on "Auto Fill"
- 2) Auto fill will not work on the paths which are already mapped.
- 3) Auto fill works for the paths which are pink in color (Pink labels are the labels which are not present in the TBC tree now)

4) If a label is available in two different configuration files then a confirmation dialog will be displayed, where user needs to select his required configuration file.



File Name Confirmation Dialog

Search Here :					
Test Labels ECU IO Labels Diagnostic Labels					
#	Diagnostic Labels	Request Path	Access Type	Response Path	Access Type
1	Req_\$1001-DEFAULT_S...				
2	Req_\$14-FS_LOESCHEN				
3	Req_\$1902-ReadDTCInfor...				
4	Req_\$22-STATUS_LESEN				
5	Req_\$22a8e2-STATUS_...				
6	Req_\$2e4048-STEUERN...				
7	Req_\$2e4048-STEUERN...				
8	Req_\$2e4048-STEUERN...				
9	Req_\$2e4048-STEUERN...				
10	Req_\$2e4048-STEUERN...				
11	Req_\$2e4048-STEUERN...				

Dependencies

Before Auto Fill

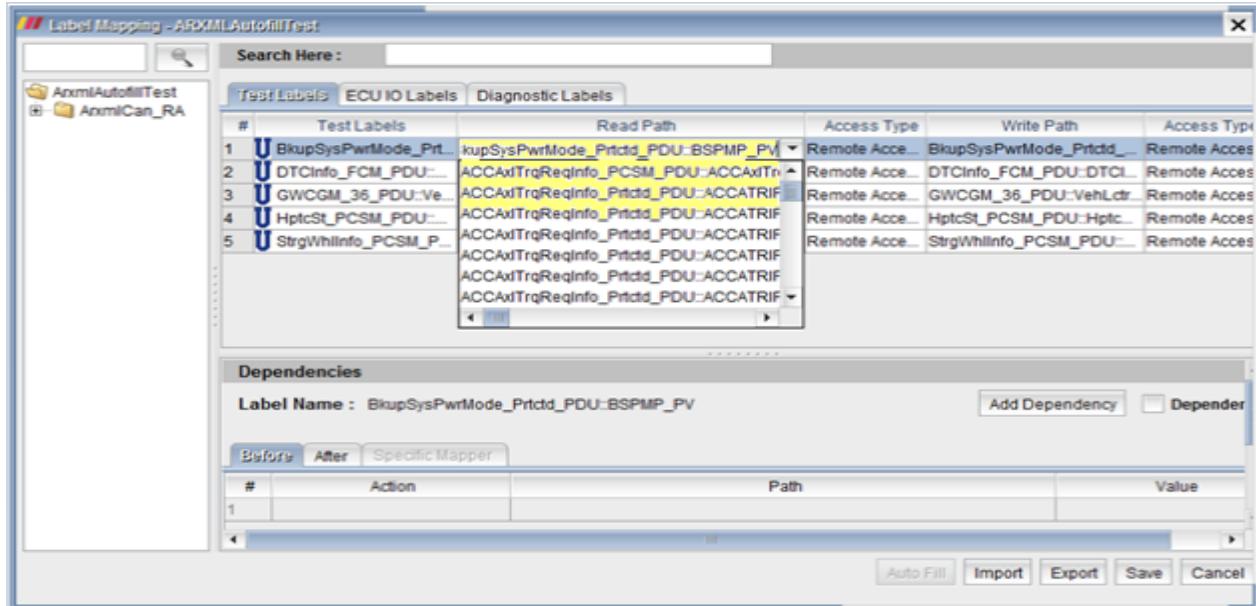
Search Here :					
Test Labels ECU IO Labels Diagnostic Labels					
#	Diagnostic Labels	Request Path	Access Type	Response Path	Access Type
1	Req_\$1001-DEFAULT_S...	\$1001-DEFAULT_SESSION	Remote Diagnostics		
2	Req_\$14-FS_LOESCHEN	\$14-FS_LOESCHEN	Remote Diagnostics		
3	Req_\$1902-ReadDTCInfor...	\$1902-ReadDTCInformation 02 r...	Remote Diagnostics		
4	Req_\$22-STATUS_LESEN	\$22-STATUS_LESEN	Remote Diagnostics		
5	Req_\$22a8e2-STATUS_...	\$22a8e2-STATUS_LESEN/FASTA...	Remote Diagnostics		
6	Req_\$2e4048-STEUERN...	\$2e4048-STEUERN/_IBRAKE_A...	Remote Diagnostics		
7	Req_\$2e4048-STEUERN...	\$2e4048-STEUERN/_IBRAKE_A...	Remote Diagnostics		
8	Req_\$2e4048-STEUERN...	\$2e4048-STEUERN/_IBRAKE_A...	Remote Diagnostics		
9	Req_\$2e4048-STEUERN...	\$2e4048-STEUERN/_IBRAKE_A...	Remote Diagnostics		
10	Req_\$2e4048-STEUERN...	\$2e4048-STEUERN/_IBRAKE_A...	Remote Diagnostics		
11	Req_\$2e4048-STEUERN...	\$2e4048-STEUERN/_IBRAKE_A...	Remote Diagnostics		

Dependencies

After AutoFill

- 5) If the label hasn't matched with any label of imported configuration file then it will be compared with imported SDF labels and mostly matched (compared based on signal) label will be highlighted yellow and

the remaining matched labels will be highlighted yellow in combo box drop down.

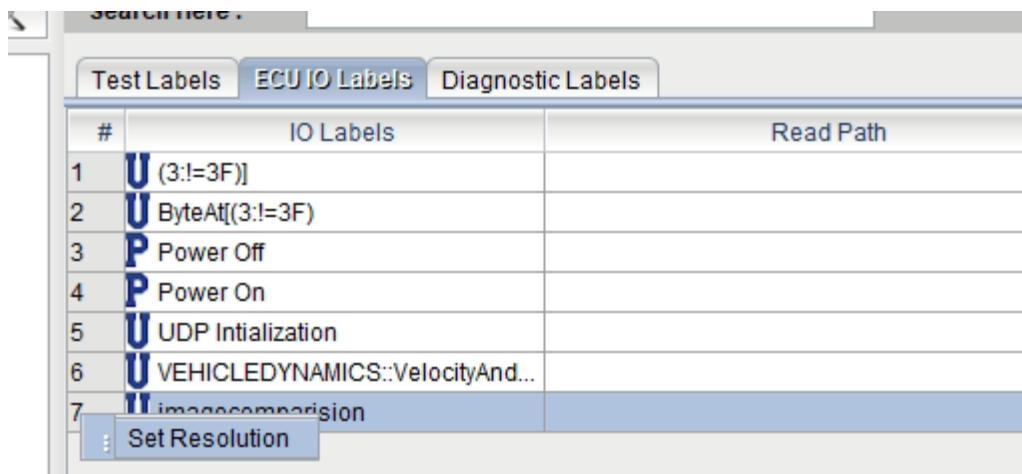


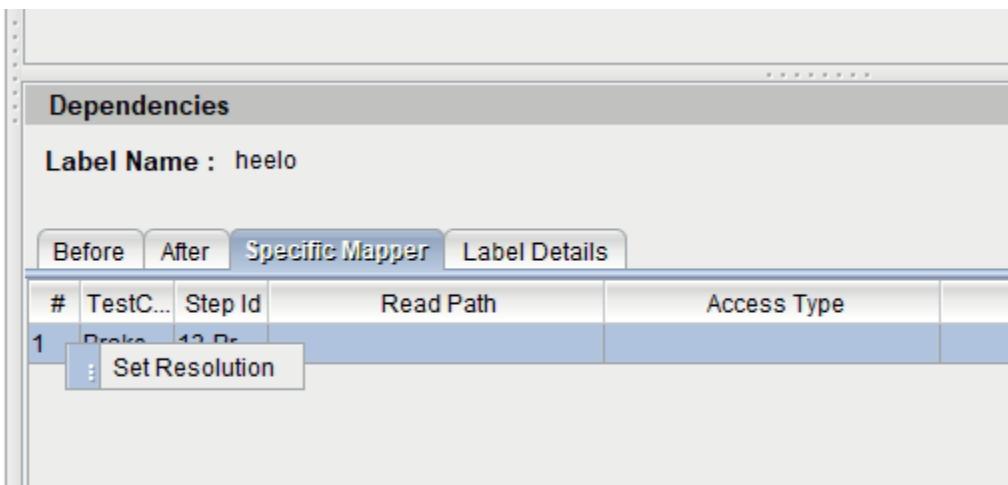
Sample image of highlighting signal matched labels.

Created with the Personal Edition of HelpNDoc: [Generate EPub eBooks with ease](#)

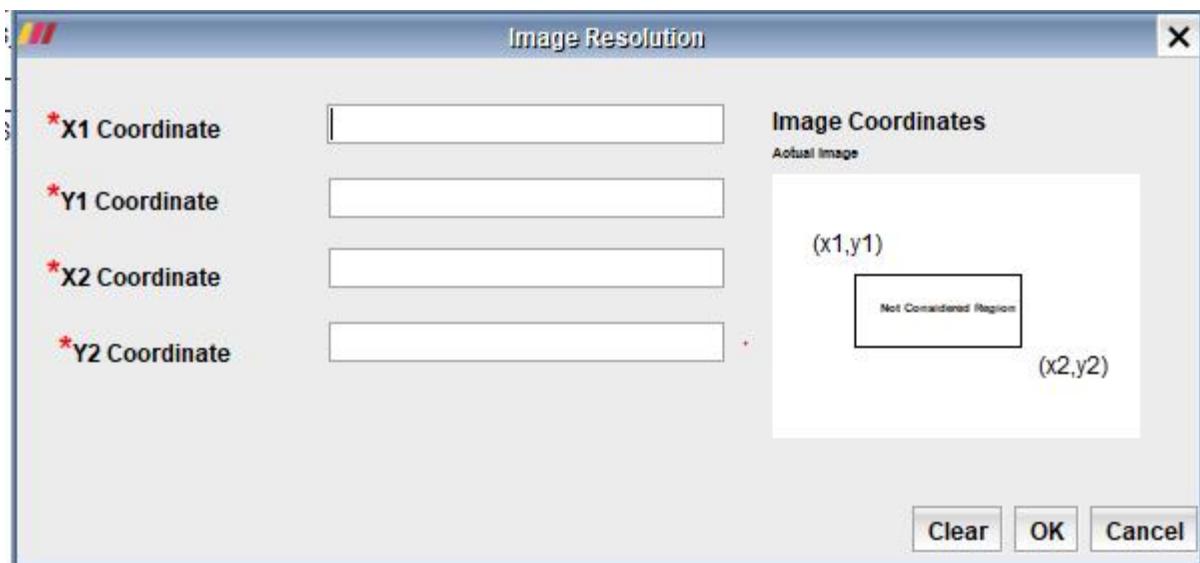
19.3.5 Image Resolution setting in label Mapping frame

With MITE-v3.0.0.0 we have image comparison set resolution process where we can set the resolution through different coordinates of the images to be compared. The process of setting different resolution to different images in a Test Case is possible now. In the test bench configuration when user open “Map labels”, there in ECU IO Labels where User can find “Set Resolution”.





On clicking it a following pop up window occurs:-



Once user submits the coordinates then User can do the test script generation which will compare the images and give the output of the images whether it is passed/failed.

Created with the Personal Edition of HelpNDoc: [What is a Help Authoring tool?](#)

19.4 Test Script Generation

To generate test script(s) in MITE, follow below steps:

1. Make sure to change the View set to “Test Script” using eye icon
2. Select a Test Suite under a Project
3. Select require Configuration from “TBC window”
4. Now click on “Generate Project Script(s)” icon from tool bar to start script generation
 - This will generate the test case(s) under selected test suite with the selected configuration
5. Wait until script generation process completes can get updates from Status bar
6. Check the generation process in Messages – Information tab

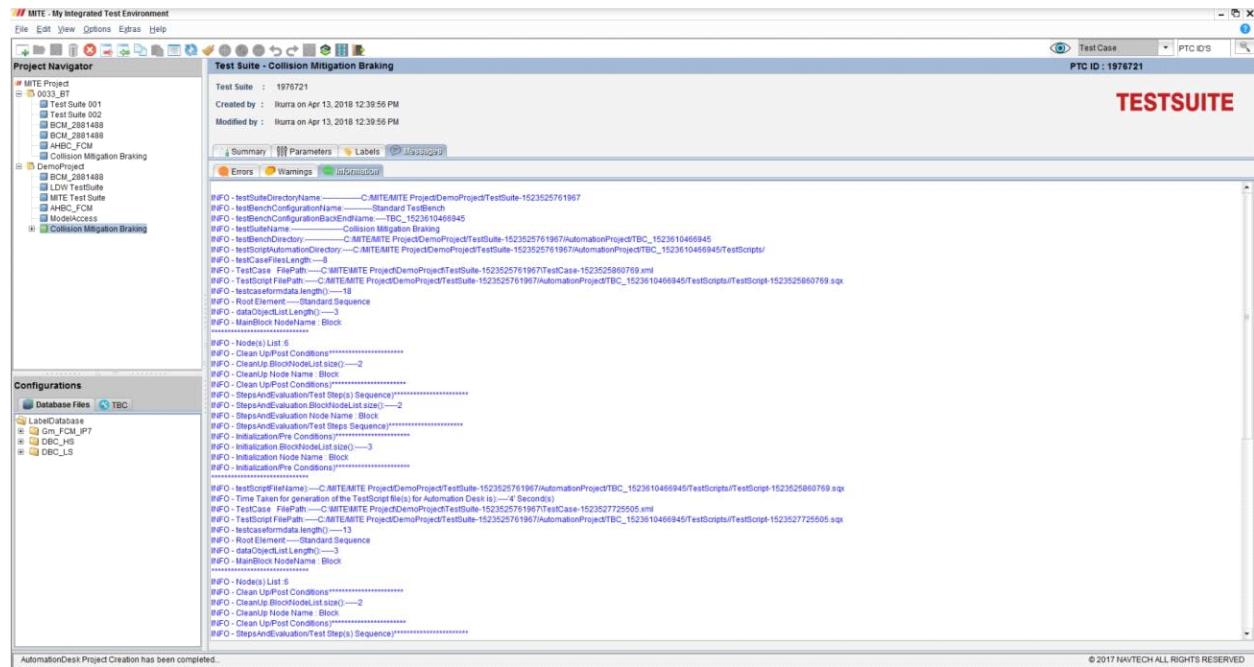


Figure 91: Script Generation information

Created with the Personal Edition of HelpNDoc: [Easy EPub and documentation editor](#)

19.4.1 MITE Test Script Generation Classification

Script generation is divided into two parts with respect to the Automation Desk license dongle as listed below:

1. Test Script Generation
2. Project Creation

Created with the Personal Edition of HelpNDoc: [Full-featured Kindle eBooks generator](#)

19.4.1.1 Test Script Generation

This provides the opportunity to the MITE user to generate test scripts from the developed test cases without the AD license dongle connected to the system.

As shown in below figure , Test Script Generation icon will generate scripts for the selected test case under a test suite and saves it in MITE folders for further use.

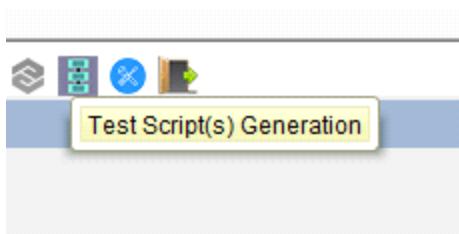


Figure 92: Test Script Generation icon

❖ ***Pre-requisites for Project Creation in Automation Desk***

1. For successful Project Creation in Automation Desk through MITE , please follow below Pre- requisites :
 - i. Python – Python needs to be installed on system. Version should be 2.7 and above
 - ii. Environmental path

```
settings : "C:
\PYTHON27;C:
\PYTHON27\Scripts"
```

2. For all other data base files successfully import Python is not mandatory

Created with the Personal Edition of HelpNDoc: [Easily create Qt Help files](#)

19.4.2 Project Creation

Here AD license dongle is compulsory to create project. Using this option a MITE user can generate or create an AD (Automation Desk) Project. This project includes all the test scripts generated for the selected test suite in MITE. Project once created can be executed directly in AD.

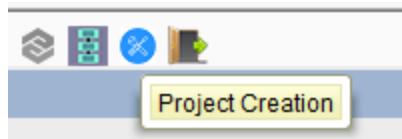


Figure 93: Project Creation icon

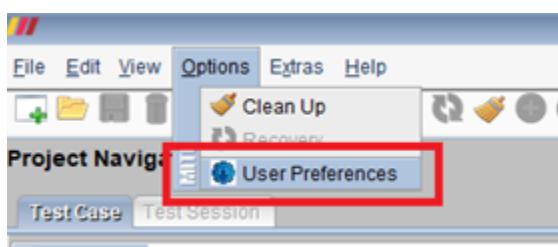
1. When user click on Project Creation button it will ask for administrator permission click YES.
2. Restart MITE.

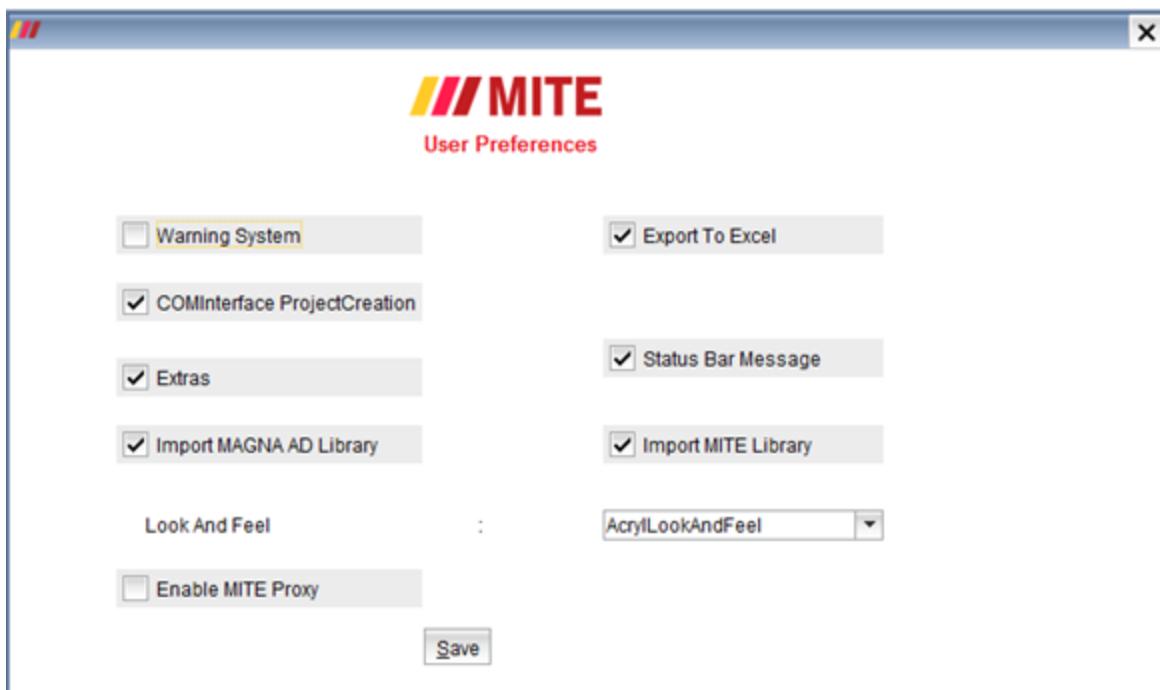


NOTE 1: If any Control Desk configuration is there user need to uncheck "JAVA PROJECT CREATION" in user preferences.

CTRL+SHIFT+F.and create project.

NOTE 2:- User need to click on option tab which is available in Menu bar and click on User Preferences as shown below:-





Created with the Personal Edition of HelpNDoc: [Easily create EPub books](#)

19.4.3 Project Creation: Diagnostic

MITE follows the below mentioned steps before creating Diagnostic Project

1. After clicking on Project Creation – Project creation process starts up
2. If project details match project gets created
3. If project details do not match - MITE Compares Diagnostic details with ControlDesk details such as – Vehicle Name and Logical Name as shown below. In this case project will not be created. The process will be stopped.
4. User required to update the details using “Update LabelData base files” option at Test Case view set

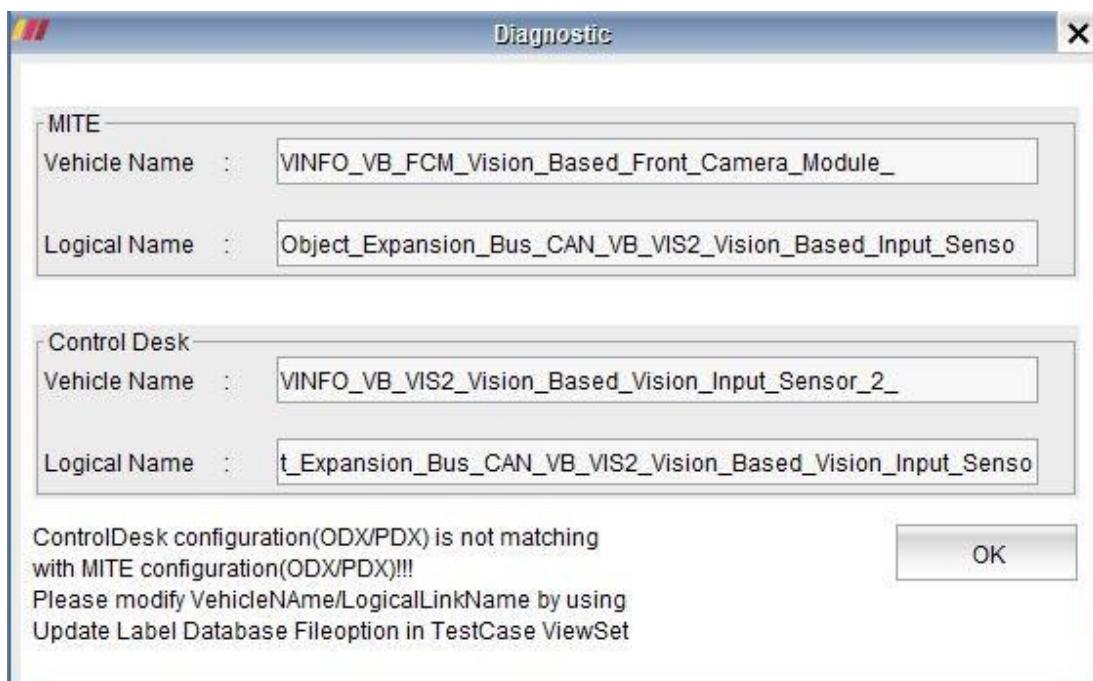


Figure 94: Comparison window Project Creation

5. Respective information will be displayed in information tab as shown below

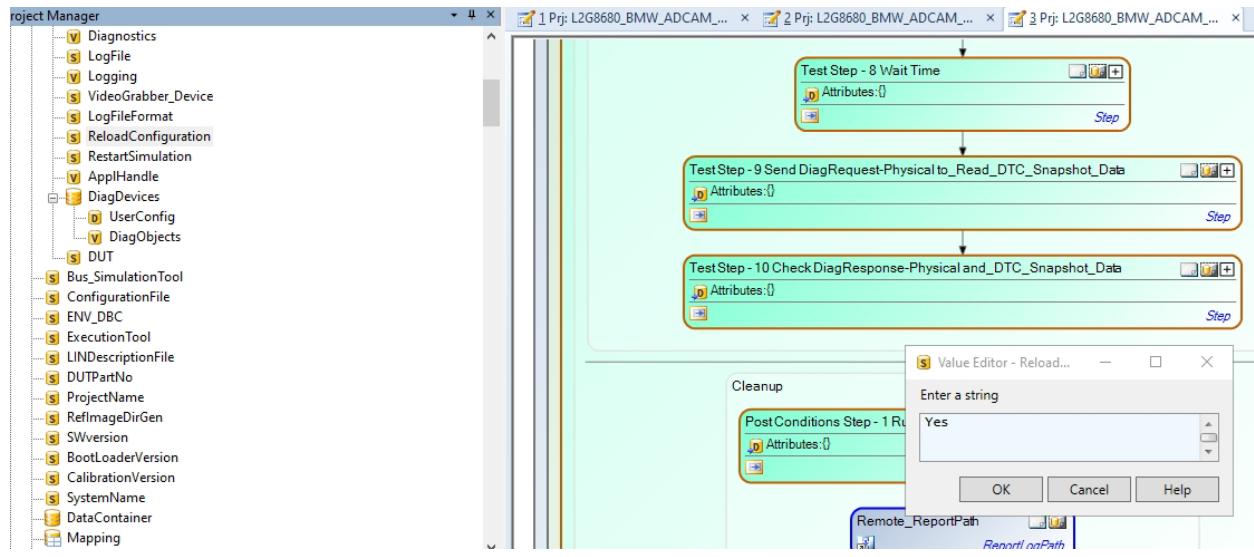


Figure 95: Project Creation information

Created with the Personal Edition of HelpNDoc: Produce Kindle eBooks easily

19.4.4 Reload Configuration in Automation Desk

- 1.In MITE 2.4 version, the reload configuration was by-default set to “YES” and user need not change it.
From
MITE -v 2.6.0.0 the reload configuration is by-default set to “NO” for using Automation desk versions 5.6 and below
- 2.With MITE V -2.6.0.0 user need to make Reload Configuration YES with the Usage of Automation Desk 5.6 or Below.



Created with the Personal Edition of HelpNDoc: Produce electronic books easily

19.5 Test Script Execution

To execute generated test script(s) in MITE, follow below steps :

1. On successful generation of Test Script(s) , Automation Desk icon will be activated
2. Click on “Hidden menu” in the system/PC
3. Right click on “Automation Desk” icon and select “Open Application”
4. Automation Desk will open with “Project node added”
5. Click on Test case sequence(s) to modify further
6. Select the sequence and click on “Execution” button in AD
7. Check the results in generated in HTML format

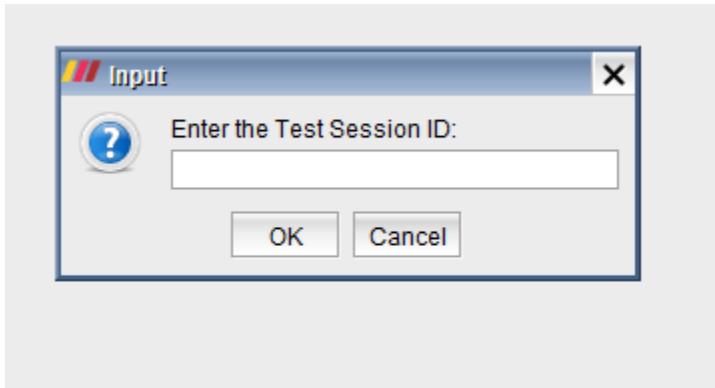
Created with the Personal Edition of HelpNDoc: Create iPhone web-based documentation

19.6 Test Script Generation using Test Session

- 1.The test suite summary table in the test script is having an option called “Test Session” to generate the scripts using test session.

#	no...	TestCase ID	Category	Table Auto Adjust
1	<input checked="" type="checkbox"/>	1976722	System Test	To check the variables in ENABLED State
2	<input checked="" type="checkbox"/>	1976724	System Test	To check transition from ENABLED State to OFF State when FCM is not in RTA state (FCM is in RTA state)
3	<input checked="" type="checkbox"/>	1976723	System Test	To check signals CPS_Collision_Preparation_Occurred, CrntStng and CPS_Collision...
4	<input checked="" type="checkbox"/>		System Test	To check transition from ENABLED State to OFF State when FCM is not in RTA state (FCM is in RTA state)
5	<input checked="" type="checkbox"/>		System Test	To check transition from ENABLED State to OFF State when FCM is not in RTA state (FCM is in RTA state)
6	<input checked="" type="checkbox"/>		System Test	To check the variables in ENABLED State

A popup as shown below will be popped up which will ask to enter the test session id:



- 1). Give a valid test session id and click on “OK”. Now a popup will be shown which will ask to enter the PTC Credentials.Enter the valid PTC Credentials.
- 2). Now if there are any common cases in both the imported test session and the test suite then those test cases will be checked in the test suite summary table in the test script role.
- 3). All the corresponding fields like shared id, shared test suite name will be filled with the details from the test session.

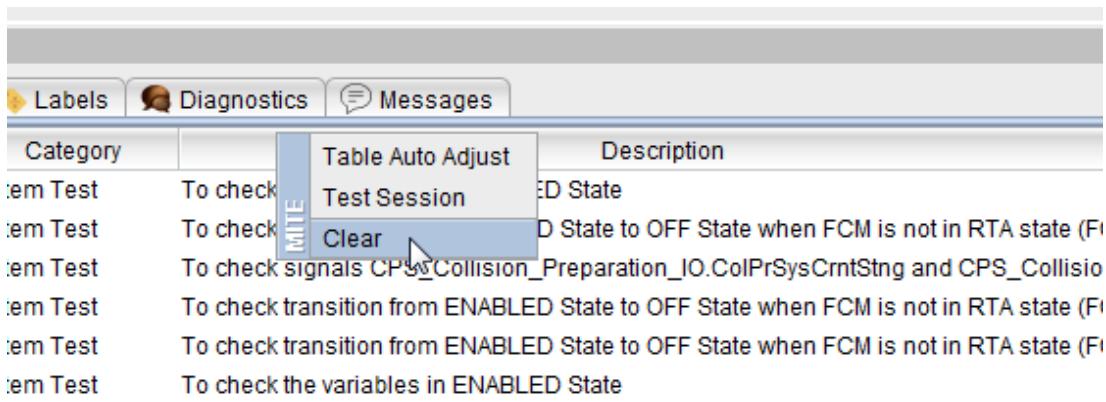
#	All	TestCase ID	Category	Description	Script Name	Automation Status	Shared Id	Shared TestSuite Id	Shared Test...
1	<input type="checkbox"/>	2023397	Heading	GenericTestCases For ASDR					
2	<input checked="" type="checkbox"/>	2023399	System Test	Clear_ASDR_Data	TestScript-2029948-154...	2029948	2029875		Shared Tess...
3	<input checked="" type="checkbox"/>	2023401	System Test	ASDR_Shutdown_Sequence	TestScript-2029947-154...	2029947	2029875		Shared Tess...
4	<input checked="" type="checkbox"/>	2023403	System Test	Enable_FCM_Features_For_ASDR_Testing	TestScript-2029946-154...	2029946	2029875		Shared Tess...
5	<input checked="" type="checkbox"/>	2023405	System Test	Set_Veh_Speed_To_60MPH	TestScript-2029944-154...	2029944	2029875		Shared Tess...
6	<input checked="" type="checkbox"/>	2023407	System Test	Set_Veh_Speed_To_0KPH	TestScript-2029942-154...	2029942	2029875		Shared Tess...
7	<input checked="" type="checkbox"/>	2023409	System Test	Precondition_PCMC_Active_True	TestScript-2029941-154...	2029941	2029875		Shared Tess...
8	<input checked="" type="checkbox"/>	2023411	System Test	Precondition_PCMC_Active_False	TestScript-2029938-154...	2029938	2029875		Shared Tess...
9	<input checked="" type="checkbox"/>	2023413	System Test	Precondition_LSCMB_Active_True	TestScript-2029936-154...	2029936	2029875		Shared Tess...
10	<input checked="" type="checkbox"/>	2023415	System Test	Precondition_LSCMB_Active_False	TestScript-2029934-154...	2029934	2029875		Shared Tess...
11	<input checked="" type="checkbox"/>	2023417	System Test	FCA_Active	TestScript-2029932-154...	2029932	2029875		Shared Tess...
12	<input type="checkbox"/>	2023419	Heading	ASDR Events Triggering					
13	<input checked="" type="checkbox"/>	2023421	System Test	Trigger_ASDR_Airbag_Deploy_Event	TestScript-2029928-154...	2029928	2029875		Shared Tess...
14	<input checked="" type="checkbox"/>	2023423	System Test	Trigger_ASDR_Lateral_Collision_Event	TestScript-2029926-154...	2029926	2029875		Shared Tess...
15	<input checked="" type="checkbox"/>	2023425	System Test	Trigger_ASDR_Near_Airbag_Deploy_Event	TestScript-2029988-154...	2029988	2029875		Shared Tess...
16	<input checked="" type="checkbox"/>	2023427	System Test	Trigger_ASDR_Low_Impact_Collision_Event	TestScript-2029985-154...	2029985	2029875		Shared Tess...
17	<input checked="" type="checkbox"/>	2023429	System Test	Trigger_ASDR_Pedestrian_Collision_Mitigation_Warning_Event	TestScript-2029983-154...	2029983	2029875		Shared Tess...
18	<input checked="" type="checkbox"/>	2023431	System Test	Trigger_ASDR_Autonomous_Emergency_Braking_Event	TestScript-2029981-154...	2029981	2029875		Shared Tess...
19	<input checked="" type="checkbox"/>	2023433	System Test	Trigger_ASDR_Pedestrian_Collision_Mitigation_Braking_Event	TestScript-2029989-154...	2029989	2029875		Shared Tess...
20	<input checked="" type="checkbox"/>	2023435	System Test	Trigger_ASDR_Autonomous_Emergency_Braking_Event	TestScript-2029987-154...	2029987	2029875		Shared Tess...
21	<input checked="" type="checkbox"/>	2023437	System Test	Trigger_ASDR_Foward_Collision_Warning_Event	TestScript-2029985-154...	2029985	2029875		Shared Tess...
22	<input type="checkbox"/>	2023439	Heading	1.5 Default Values (3330868)					
23	<input type="checkbox"/>	2023441	System Test	3330861.Unless specified in the particular data otherwise, whenever there is unwrite...	TestScript-2029983-154...	2029983	2029875		Shared Tess...
24	<input type="checkbox"/>	2023443	Heading	2 ASDR Consumed Interfaces - 3330921					
25	<input type="checkbox"/>	2023445	Heading	2.7 Invalid Interfaces - 3986175					
26	<input checked="" type="checkbox"/>	2023447	System Test	UTC YearThis test case is to verify the default data being considered when signal val...	TestScript-2029880-154...	2029880	2029875		Shared Tess...
27	<input checked="" type="checkbox"/>	2023449	System Test	UTC Month -3330929This test case is to verify the default data being considered whe...	TestScript-2029879-154...	2029879	2029875		Shared Tess...
28	<input checked="" type="checkbox"/>	2023451	System Test	UTC Day [Header]- 3330933.This test case is to verify the default data being consider...	TestScript-2029878-154...	2029878	2029875		Shared Tess...
29	<input checked="" type="checkbox"/>	2023453	System Test	UTC Hours [Header]- 3330719 .This test case is to verify if the UTC Hours event dat...	TestScript-2029877-154...	2029877	2029875		Shared Tess...
30	<input checked="" type="checkbox"/>	2023455	System Test	UTC Minutes [Header]- 3330972.This test case is to verify if the Time of CA event dat...	TestScript-2029876-154...	2029876	2029875		Shared Tess...
31	<input checked="" type="checkbox"/>	2023457	System Test	UTC Seconds [Header]- 3330973.This test case is to verify if the UTC Seconds event dat...	TestScript-2029874-154...	2029874	2029875		Shared Tess...
32	<input checked="" type="checkbox"/>	2023459	System Test	Positioning System Latitude [Header]- 3330982.This test case is to verify if the Position...	TestScript-2029913-154...	2029913	2029875		Shared Tess...
33	<input checked="" type="checkbox"/>	2023461	System Test	Positioning System Longitude [Header]- 3330990.This test case is to verify if the Posit...	TestScript-2029912-154...	2029912	2029875		Shared Tess...
34	<input checked="" type="checkbox"/>	2023463	System Test	Vehicle Odometer Display Value Authenticated [Header]- 3330853.This test case is to ...	TestScript-2029911-154...	2029911	2029875		Shared Tess...
35	<input checked="" type="checkbox"/>	2023465	System Test	Torque Overlay Active Authenticated - 3732358.This test case is to verify the default da...	TestScript-2029910-154...	2029910	2029875		Shared Tess...

- 1). Now scripts can be generated for the checked test cases.
- 2). Now the field test script name will be updated with the shared id instead of test case id.

#	All	TestCase ID	Category	Description	Script Name	Automati...	Shared Id	Shared Te...	Share...
	<input type="checkbox"/>	2023397	Heading	GenericTestCas...					
	<input checked="" type="checkbox"/>	2023399	System Test	Clear_ASDR_Data TestScript-2029948-1540297123000.sqx	2029948	2029875			Share...
	<input checked="" type="checkbox"/>	2023401	System Test	ASDR_Shutdown... TestScript-2029947-1540297123082.sqx	2029947	2029875			Share...
	<input checked="" type="checkbox"/>	2023403	System Test	Enable_FCM_Fe... TestScript-2029946-1540297124059.sqx	2029946	2029875			Share...
	<input checked="" type="checkbox"/>	2023405	System Test	Set Veh Speed TestScript-2029944-1542101839435.sqx	2029944	2029875			Share...

- 3).Now, the project generation and the execution process can be followed as usual.
- 4).To delete the data regarding already imported test session, the option

"Clear" can be used. This makes all the test cases unchecked and deletes the data related to the previously imported test session.



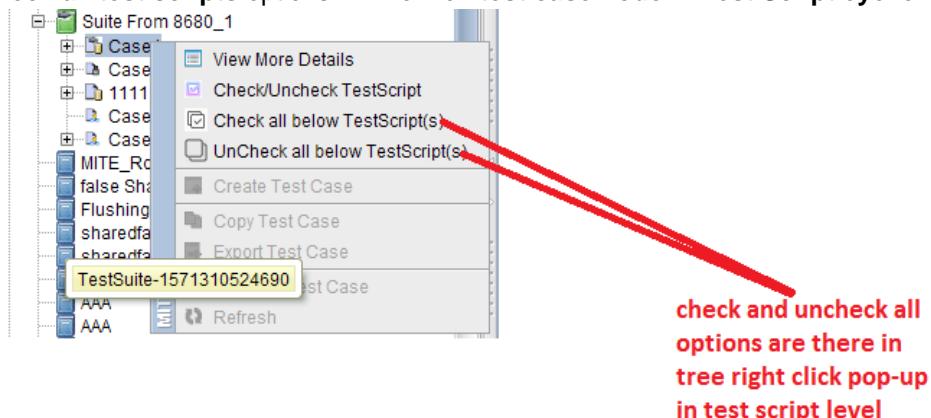
Category		Description
Item Test	To check	Table Auto Adjust
Item Test	To check	Test Session
Item Test	To check	MITE State
Item Test	To check	D State to OFF State when FCM is not in RTA state (F)
Item Test	To check	To check signals CPS_Collision_Preparation_IO.ColPrSysCrntStng and CPS_Collisio
Item Test	To check	To check transition from ENABLED State to OFF State when FCM is not in RTA state (F)
Item Test	To check	To check transition from ENABLED State to OFF State when FCM is not in RTA state (F)
Item Test	To check	To check the variables in ENABLED State

Created with the Personal Edition of HelpNDoc: [Easily create Qt Help files](#)

19.7 Check & Uncheck all child cases from tree right click option

Check all test scripts and uncheck all test scripts option in tree right click pop-up menu

1. If we choose **check all test scripts** option from tree right click pop-up, then all children of selected test cases will be going to check in test script summary table.
2. If we choose **uncheck all test scripts** option from tree right click pop-up, then all children of selected test cases will be going to uncheck in test script summary table.
3. **Check and uncheck all test scripts** options will work on **test case node** in **Test Script eye level**



Check and uncheck all options are in disable mode

1. If selected test case does not contain children then both of those options are going to disable.

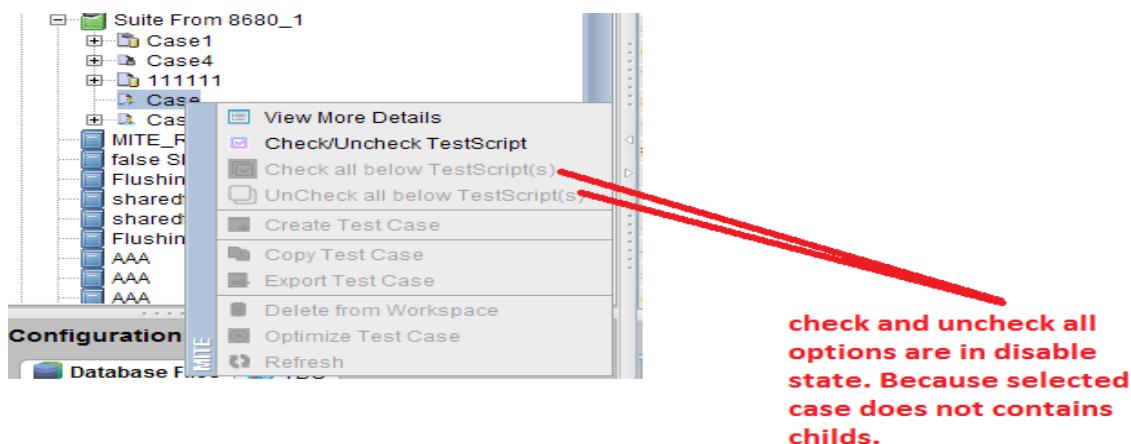


Figure: Check and uncheck all test script options are going to be disable.

2. If selected tree nodes count more than one, both of those options are going to disable

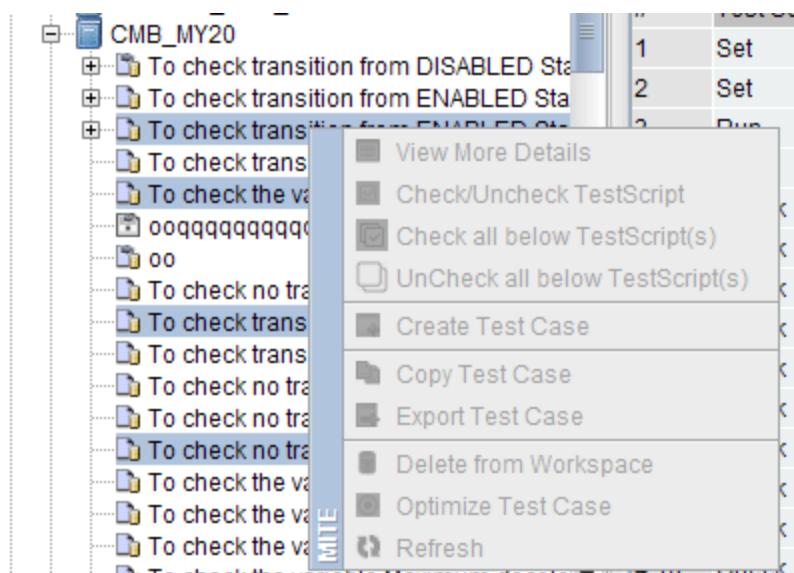


Figure: Check and uncheck all test script options are going to be disable due to the multiple selection of nodes

Check all test scripts option functionality on test script summary table

If we choose **check all test scripts** option from tree right click pop-up, then all children of selected test cases will be going to check in test script summary table as like below.

#	All	TestCase ID	Category	Description	Script Name	Automation Status	Shared Id	Shared TestSuite Id	Shared Test... Valid For Proj...	Priority
1	<input type="checkbox"/>	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Ena...	TestScript-NEW ITEM-15...						
2	<input type="checkbox"/>	System Test	To check variable "M_AEBr_AccelPedalHome" when transition form prefill to Auto br...	TestScript-NEW ITEM-15...						
3	<input checked="" type="checkbox"/>	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Ena...	TestScript-NEW ITEM-15...						
4	<input type="checkbox"/>	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Ena...	TestScript-NEW ITEM-15...						
5	<input type="checkbox"/>	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Ena...	TestScript-NEW ITEM-15...						
6	<input checked="" type="checkbox"/>	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_Valid..."	TestScript-NEW ITEM-15...						
7	<input checked="" type="checkbox"/>	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Ena...	TestScript-NEW ITEM-15...						
8	<input type="checkbox"/>	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_Valid..."	TestScript-NEW ITEM-15...						
9	<input type="checkbox"/>	System Test	To check the variable "M_AEBv_Autonomous_Braking_Magnitude" when system is E...	TestScript-NEW ITEM-15...						
10	<input type="checkbox"/>	Heading	oooooooooooooo							
11	<input type="checkbox"/>	System Test	oo		TestScript-NEW ITEM-15...					
12	<input type="checkbox"/>	System Test	To check no transition from Inactive to IBA State when variable "M_AEBv_BrakeAssi...	TestScript-NEW ITEM-15...						
13	<input type="checkbox"/>	System Test	To check transition from Hold to Inactive state when variable "M_AEBv_OverrideAccel..."	TestScript-NEW ITEM-15...						
14	<input type="checkbox"/>	System Test	To check transition from Hold to Inactive state when variable "M_AEBv_HoldExpired =...	TestScript-NEW ITEM-15...						
15	<input type="checkbox"/>	System Test	To check no transition from Prefill State to IBA State when variable "M_AEBv_BrakeAs..."	TestScript-NEW ITEM-15...						
16	<input type="checkbox"/>	System Test	To check no transition from FLB4 State to IBA State when variable "M_AEBv_Override..."	TestScript-NEW ITEM-15...						
17	<input type="checkbox"/>	System Test	To check no transition from FLB4 State to IBA state when variable "M_AEBv_BrakeAs..."	TestScript-NEW ITEM-15...						
18	<input type="checkbox"/>	System Test	To check the variable "M_AEBr_Autonomous_Braking_Magnitude" when system is in...	TestScript-NEW ITEM-15...						
19	<input type="checkbox"/>	System Test	To check the variable M_AEBr_LongDistance in m	TestScript-NEW ITEM-15...						
20	<input type="checkbox"/>	System Test	To check the variable M_AEBr_LongDistance in m	TestScript-NEW ITEM-15...						
21	<input type="checkbox"/>	System Test	To check the variable Maximum deceleration	TestScript-NEW ITEM-15...						
22	<input type="checkbox"/>	System Test	To check the variable Maximum deceleration	TestScript-NEW ITEM-15...						
23	<input type="checkbox"/>	System Test	To check transition from Hold to Inactive state when variable "M_AEBr_HoldOverride..."	TestScript-NEW ITEM-15...						
24	<input type="checkbox"/>	System Test	To check transition from Hold to Inactive state when variable "M_AEBr_OverrideAccel..."	TestScript-NEW ITEM-15...						
25	<input type="checkbox"/>	System Test	To check transition from Hold to Inactive state when variable "M_AEBr_HoldExpired =..."	TestScript-NEW ITEM-15...						
26	<input type="checkbox"/>	System Test	To check transition from IBA State to Inactive State when variable "M_AEBr_BrakeAssi..."	TestScript-NEW ITEM-15...						
27	<input type="checkbox"/>	System Test	To check no transition from FLB4 State to IBA State when variable "M_AEBr_Override..."	TestScript-NEW ITEM-15...						
28	<input type="checkbox"/>	System Test	To check no transition from FLB4 State to IBA state when variable "M_AEBr_BrakeAs..."	TestScript-NEW ITEM-15...						
29	<input type="checkbox"/>	System Test	To check variable "M_AEB_CommandType" when variable "M_AEB_CommandType" =	TestScript-NEW ITEM-15...						
30	<input type="checkbox"/>	System Test	To check variable "M_AEB_CommandType" when variable "M_AEB_CommandType" =	TestScript-NEW ITEM-15...						
31	<input type="checkbox"/>	System Test	To check variable "M_AEBv_AccelPedalHome" when transition form Inactive to Auto b...	TestScript-NEW ITEM-15...						
32	<input type="checkbox"/>	System Test	To check variable "M_AEBv_AccelPedalHome" when transition form Prefill to Auto br...	TestScript-NEW ITEM-15...						
33	<input type="checkbox"/>	System Test	To check variable "M_AEBv_AccelPedalHome" when transition form Inactive to Auto b...	TestScript-NEW ITEM-15...						
34	<input type="checkbox"/>	System Test	To check variable "M_AEBv_AccelPedalHome" when transition form Inactive to Auto ...	TestScript-NEW ITEM-15...						
35	<input type="checkbox"/>	System Test	To check variable "M_AEBr_AccelPedalHome" when transition form Inactive to Auto ...	TestScript-NEW ITEM-15...						

Figure: Child cases of selected test case are checked if we use option check all test script(s)

Uncheck all test scripts option functionality on test script summary table

If we choose **uncheck all test scripts** option from tree right click pop-up, then all children of selected test cases will be going to uncheck in test script summary table.

#	All	TestCase ID	Category	Description	Script Name	Automation Status	Shared Id	Shared TestSuite Id	Shared Test... Valid For Proj...	Priority
1	<input type="checkbox"/>	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Ena...	TestScript-NEW ITEM-15...						
2	<input type="checkbox"/>	System Test	To check variable "M_AEBr_AccelPedalHome" when transition form prefill to Auto br...	TestScript-NEW ITEM-15...						
3	<input checked="" type="checkbox"/>	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Ena...	TestScript-NEW ITEM-15...						
4	<input type="checkbox"/>	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_Valldip..."	TestScript-NEW ITEM-15...						
5	<input type="checkbox"/>	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Ena...	TestScript-NEW ITEM-15...						
6	<input type="checkbox"/>	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_Valldip..."	TestScript-NEW ITEM-15...						
7	<input type="checkbox"/>	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Ena...	TestScript-NEW ITEM-15...						
8	<input type="checkbox"/>	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_TorqueL..."	TestScript-NEW ITEM-15...						
9	<input type="checkbox"/>	System Test	To check the variable "M_AEBv_Autonomous_Braking_Magnitude" when system is E...	TestScript-NEW ITEM-15...						
10	<input type="checkbox"/>	Heading	oooooooooooooo							

Cases are unselect due to the selection of "Uncheck all test script(s)" option from tree right click option

Created with the Personal Edition of HelpNDoc: Write eBooks for the Kindle

19.8 Requirements for Specific Mapper Implementation in TBC

Overview:

1. Project Background and Description

Currently, in the Label Mapping Frame, even though the same Test Label, IO Label or Diagnostic Label is used in many test cases, it is fetched as a single label into the mapping frame and

will be allowed to be mapped with a single Read Path and Write Path. But, there may be times where its usage will be different and so it should be mapped with different Read Paths and Write Paths.

So, this implementation will give a breakthrough for Label Mapping frame by allowing the user to map the same label which is used in different test cases with the different Read Paths and Write Paths.

This implementation is not required for Test Labels as their usage will not vary. So, this should be implemented for IO Labels and Diagnostic Labels.

2. Project Scope

This implementation should be done in MITE -> TBC -> Label Mapping Frame

3. High-Level Requirements

User Requirements

1. In Label Mapping window -> User should be able to see another tab as "Specific Mapper" in the lower pane.
2. This different table should be visible only in the IO Labels and Diagnostic Labels tab.
3. The UI implementation should be almost similar to the respective tabs i.e., the IO Specific mapper table should look the IO Labels table and the Diagnostic Specific mapper table should look the Diagnostic Labels table.
4. Two extra fields which are to be added into this table are the Test Case Name and the Test Case step ID where this particular IO Label or Diagnostic Label is used. This helps the user to identify exact position or place (which Test Case and which Step Id) where the label is used.
5. If the usage of a particular label is not more than one time then, a message should be displayed in the specific mapper table pane as "No Specific Mapper details are found".
6. The Short Name validation should be implemented for both the specific mapper tables.
7. After filling the details and by clicking on save button, these details have to be stored as Specific Mapping Details under the corresponding IO Label or the Diagnostic Label in the LabelMapping.xml.
8. Now, while generating the script, the label should be processed in such a way that if specific mapping details are found then those details are to be considered and if specific mapping details are not found then the main label details are to be considered.

Interface Requirements

1. The specific mapper table should also have features like DND and Combo boxes.
2. The Add Dependency button and the dependency checkbox should be disabled in these specific mapper tabs.
3. The basic View of these specific mapper tables should be as shown below.

Test Labels **ECU IO Labels** **Diagnostic Labels**

#	IO Labels	Read Path	Access Type	Write Path	Access Type	Short Name
1	Calculate_Bias for ...					
2	Calculate_Bias for l...					
3	Calculate_Accuracy...					
4	Calulate_Bias for l...					
5	Calulate-Accuracy f...					
6	Calulate_Accuracy...					
7	Calulate_Average ...					
8	Calulate_Average ...					
9	Calulate_Average ...					
10	Calulate_Uncertai...					
11	Calulate_Average ...					
12	Calulate_Bias for ...					

Dependencies

Label Name : Calculate_Bias for vertical position

Add Dependency Dependencies

Before **After** **Specific Mapper**

#	TestCaseName	Step Id	Read Path	Access Type	Write Path	Access Type	Short Name
1	Testtoverifythatbiasoft...	15					

The screenshot shows the 'Diagnostic Labels' tab of a software interface. At the top, there are three tabs: 'Test Labels', 'ECU IO Labels', and 'Diagnostic Labels'. The 'Diagnostic Labels' tab is active. Below the tabs is a table with the following columns: #, Diagnostic Labels, Request Path, Access Type, Response P..., Access Type, and Short Name. The rows contain the following data:

#	Diagnostic Labels	Request Path	Access Type	Response P...	Access Type	Short Name
1	Req_\$624004-STATU...					
2	Req_\$3100-STEUER...					
3	Res_\$71-STEUERN_...					
4	Req_\$2E 4004 03					
5	Req_\$6E 4004 03					
6	Res_\$22 4004					
7	Res_\$62 4004					
8	Req_\$2e4004-STEU...					
9	Res_\$6e4004-STEU...					
10	Res_\$224004-STATU...					
11	Res_Number of kms					
12	Res_Number of kms ...					

Below the table is a section titled 'Dependencies' with the label name 'Res_\$71-STEUERN_ROUTINE'. There are buttons for 'Add Dependency' and 'Dependencies'. At the bottom of this section is a table with columns: #, TestCaseName, Step Id, Request Path, Access Type, Response Path, Access Type, and Short Name. It contains one row:

#	TestCaseName	Step Id	Request Path	Access Type	Response Path	Access Type	Short Name
1	ThisTestverifiesth...	5					

Diagnostic Specific Mapper

4. When no specific mapping details are found, the view should be shown as below:

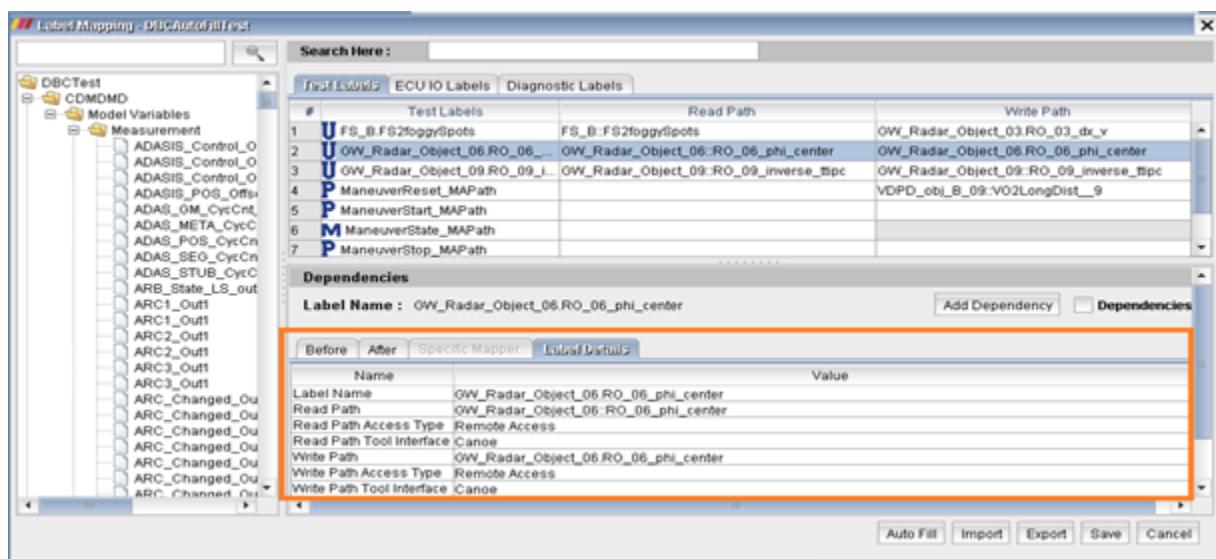
The screenshot shows the 'Specific Mapper' tab of the software interface. At the top, there are three tabs: 'Before', 'After', and 'Specific Mapper'. The 'Specific Mapper' tab is active. In the center of the screen, the text 'No specific mapper details are found.' is displayed.

Created with the Personal Edition of HelpNDoc: [What is a Help Authoring tool?](#)

19.9 Label Details Tab

Label Details Tab:-

User can view the selected label details like LabelName, Read Path, Read Path Access Type, Read Path Tool Interface, Write Path, Write Path Access Type, and Write Path Tool Interface in the tab.



Label Details Tab in Label Mapping Frame.

Created with the Personal Edition of HelpNDoc: [Easily create Qt Help files](#)

19.10 MITE and Automation Desk Integration

1. For successful Project Creation in Automation Desk through MITE , please follow below Pre-requisites:
 - i. Python – Python needs to be installed on system. Version should be 2.7 and above
 - ii. Environmental path settings : “C: \Python27;C:\Python27\Scripts”
 2. For all other data base files successfully import Python is not mandatory

19.4.2 Project Creation

Here AD license dongle is compulsory to create project. Using this option a MITE user can generate or create an AD (Automation Desk) Project. This project includes all the test scripts generated for the selected test suite in MITE. Project once created can be executed directly in AD.

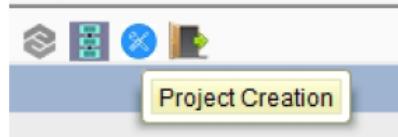
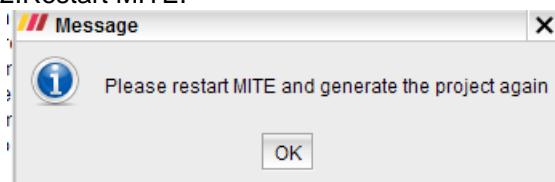


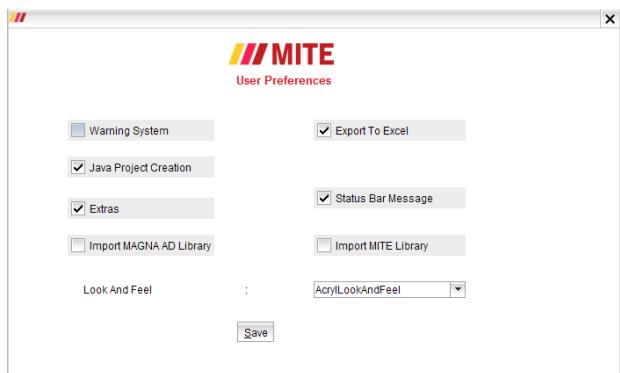
Figure 93: Project Creation icon

First time of Project creation

1. When user click on Project Creation button it will ask for administrator permission click YES.
 2. Restart MITE.



NOTE: If any Control Desk configuration is there user need to uncheck "JAVA PROJECT CREATION" in user preferences .CTRLR+SHIFT+F.and create project.



Created with the Personal Edition of HelpNDoc: Single source CHM, PDF, DOC and HTML Help creation

19.11 Multiple Suite test script generation and project creation

Description:

Earlier we can do test script generation for single suite. But using this enhancement at a time multiple suites we can do test script generation.

1. Test script eye selection

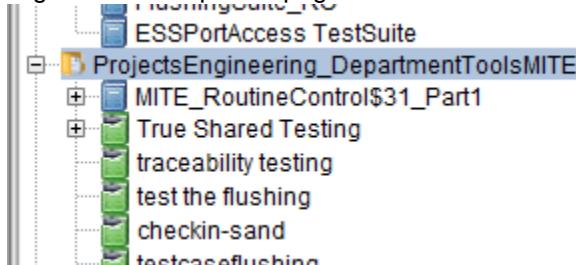
Select Test Script eye option to generate test script as like below



IMG: Select test script eye option

2. Select a project from Tree

To generate multiple script generation we need to select Project node from tree.



IMG: Select a project from tree

3. Project summary table with check boxes

While we select a project node in test script level, the summary table will update with check boxes to select test suites which are going to be generate test script.

#	ID	Type	Summary	Status	MITE Compatibility	All
1	7596670	Suite	testSuite	Open	True	<input type="checkbox"/>
2	7370654	Suite	test the flushing	Open	True	<input type="checkbox"/>
3	6712401	Suite	True Shared Testing	Open	True	<input type="checkbox"/>
4	6452308	Suite	shareSuite	Open	True	<input type="checkbox"/>
5		Suite	MITE_RoutineControl\$31_Part1	Open	True	<input type="checkbox"/>
6	7329245	Suite	traceability testing	Open	True	<input type="checkbox"/>
7	7370641	Suite	test the flushing	Open	True	<input type="checkbox"/>

IMG: Project summary table with check boxes

4. Select all check boxes

To select all check boxes from project summary table we should select column header with check box.

#	ID	Type	Summary	State	MITE Compatibility	<input checked="" type="checkbox"/>
1	7596670	Suite	testsuite	Open	True	<input checked="" type="checkbox"/>
2	7370564	Suite	test the bushing	Open	True	<input checked="" type="checkbox"/>
3	6712401	Suite	True Shared Testing	Open	True	<input checked="" type="checkbox"/>
4	6462308	Suite	sharedfalse	Open	True	<input checked="" type="checkbox"/>
5		Suite	MITE_RoutineControl\$31_Part1	Open	True	<input checked="" type="checkbox"/>
6	7329245	Suite	traceability testing	Open	True	<input checked="" type="checkbox"/>
7	7370541	Suite	test the bushing	Open	True	<input checked="" type="checkbox"/>
8	7383564	Suite	checkin-sand	Open	True	<input checked="" type="checkbox"/>
9	7550512	Suite	testcaseflushing	Open	True	<input checked="" type="checkbox"/>
10	7623412	Suite	testsuitefortoc	Open	True	<input checked="" type="checkbox"/>
11	7623085	Suite	testsuite	Open	True	<input checked="" type="checkbox"/>
12	7564410	Suite	test script flush	Open	True	<input checked="" type="checkbox"/>

IMG: Select all check boxes

5. Select individual check boxes

To select check box individually we should select a check box which is placed at last index of project summary table in test script level.

#	ID	Type	Summary	State	MITE Compatibility	<input checked="" type="checkbox"/>
5	7329245	Suite	MITE_RoutineControl\$31_Part1	Open	True	<input checked="" type="checkbox"/>
6		Suite	traceability testing	Open	True	<input checked="" type="checkbox"/>
7	7370541	Suite	test the bushing	Open	True	<input checked="" type="checkbox"/>
8	7383564	Suite	checkin-sand	Open	True	<input checked="" type="checkbox"/>
9	7550512	Suite	testcaseflushing	Open	True	<input checked="" type="checkbox"/>
10	7623412	Suite	testsuitefortoc	Open	True	<input checked="" type="checkbox"/>
11	7623085	Suite	testsuite	Open	True	<input checked="" type="checkbox"/>
12	7564410	Suite	test script flush	Open	True	<input checked="" type="checkbox"/>
13		Suite	NewTestCase@#%&\"_+o-`1\ @ +	Open	True	<input checked="" type="checkbox"/>
14	7604162	Suite	FlushingSuite_RC	Open	True	<input checked="" type="checkbox"/>
15		Suite	Suite From 8680_1	Open	True	<input checked="" type="checkbox"/>

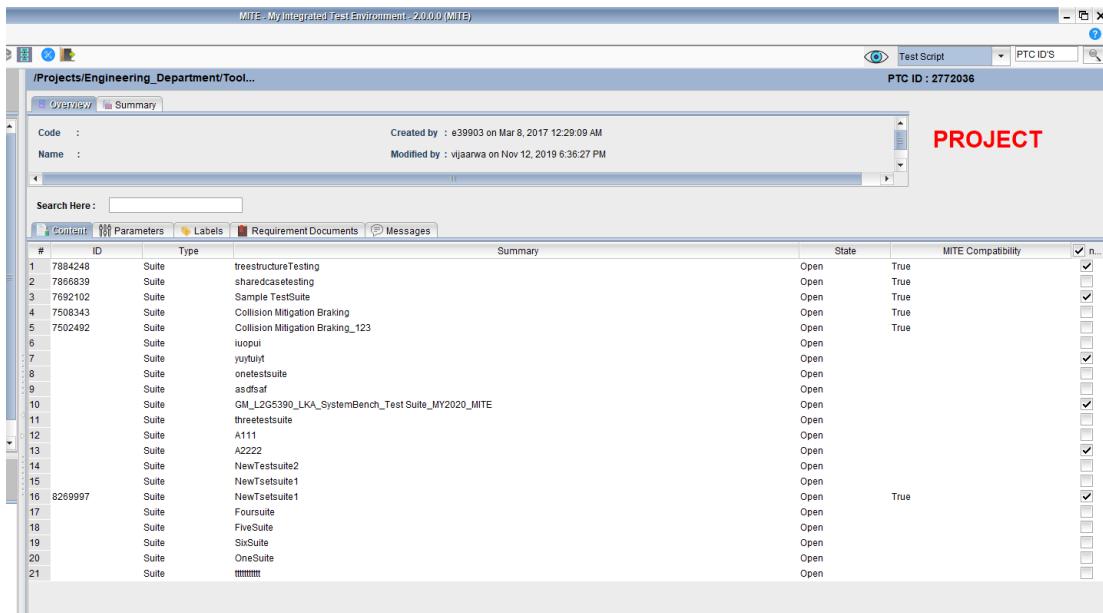
IMG: Select individual check boxes.

The screenshot shows the MITE software interface with the following details:

- Project Navigator:** Shows a tree structure with 'MITE Project' expanded, containing 'DemoProject' and several test cases like 'Collision Mitigation Braking', 'L2G8680_BMW_ADCAM_CMB_MV20', etc.
- Test Case Details:** For the test case 'L2G8680', the 'Overview' tab is selected. It shows the code 'L2G8680', created by 'MEE_MSCHRO' on Jan 26, 2017, and modified by 'zainbedr' on May 29, 2019, at 3:43:55 PM. The PTC ID is 2452288.
- Summary Table:** A table with columns: #, ID, Type, Summary, State, MITE Compatibility, and a final column with checkboxes. The 16th row (last row) has its checkbox checked.

6. Multiple script generation

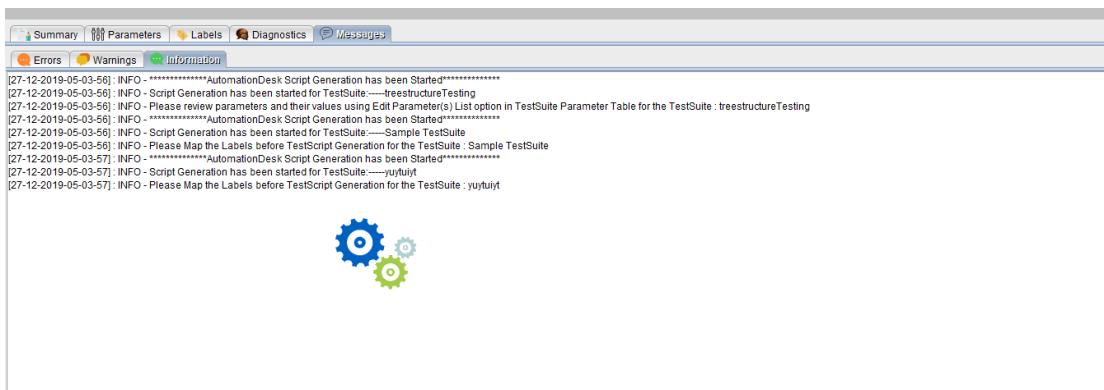
We can do multiple script generation for checked suite from project summary table by clicking script generation button in test script level.



The screenshot shows the 'Project Summary' table in the MITE interface. The table lists 21 test suites, each with an ID, Type, Summary, State, and MITE Compatibility column. The 'PROJECT' label is visible at the top right of the window.

#	ID	Type	Summary	State	MITE Compatibility
1	7894248	Suite	treestructureTesting	Open	True
2	786639	Suite	sharedcasetesting	Open	True
3	7692102	Suite	Sample TestSuite	Open	True
4	7508343	Suite	Collision Mitigation Braking	Open	True
5	7502492	Suite	Collision Mitigation Braking_123	Open	True
6		Suite	iuopui	Open	
7		Suite	yuyuif	Open	
8		Suite	onetestsuite	Open	
9		Suite	asdfsaf	Open	
10		Suite	GM_L2G5390_LKA_SystemBench_Test Suite_MY2020_MITE	Open	
11		Suite	threeTestsuite	Open	
12		Suite	A111	Open	
13		Suite	A222	Open	
14		Suite	NewTestsuite2	Open	
15		Suite	NewTestsuite1	Open	
16	8269997	Suite	NewTestsuite1	Open	True
17		Suite	Foursuite	Open	
18		Suite	FiveSuite	Open	
19		Suite	SixSuite	Open	
20		Suite	OneSuite	Open	
21		Suite	tttttttt	Open	

IMG: Select test suites from project summary table



IMG: Multiple script generation happened while click action performed on script button

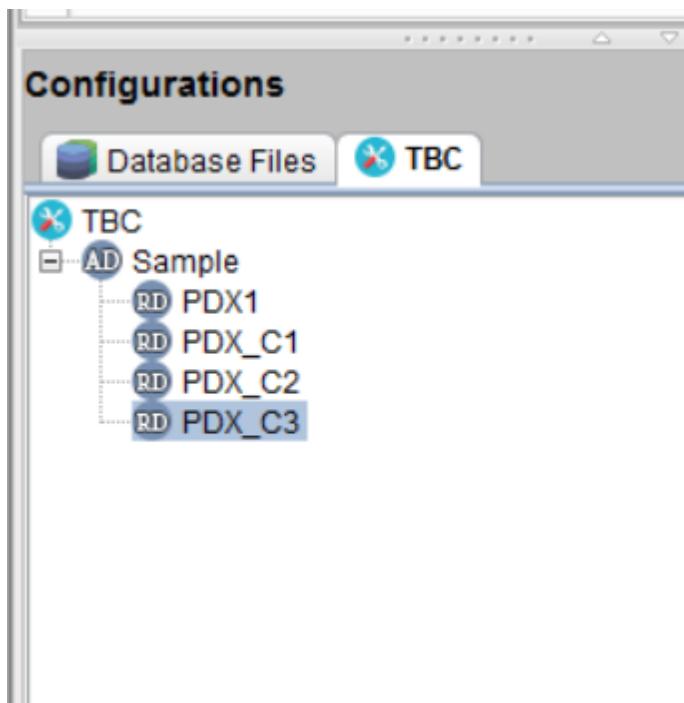
Created with the Personal Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

19.11.1 Multiple Allowances of Remote Diagnostic Access Files

Multiple Allowances of Remote Diagnostic Access Files

Introduction:

In TBC, user can add only one Remote Diagnostic file for one Tool Interface until MITE v 2.6.0.0. But, after that MITE started supporting different Diag Devices by allowing users to add multiple Remote Diagnostics files in one TBC. This is allowed only for Canoe Tool Interface only.



Setting Default Device Name:

For Diagnostic Labels, if there is no mapping found also, user can select the default device for that particular label. This can be done by right clicking on a particular Diag Label and clicking on "Select Diag Device".

- Select the Device Name from the list shown in the combo box..
- If no mapping is found only, then user chooses diag device will be considered.
- This selection of Diag device can be done for multiple labels by selecting them at a time in the diagnostics table.

#	Diagnostic Labels	Request Path
1	Req_\$221005-車速 Read	
2	Req_\$7002_Pressured(油圧情報切替え) Read	
	Set Diagnostic Device	

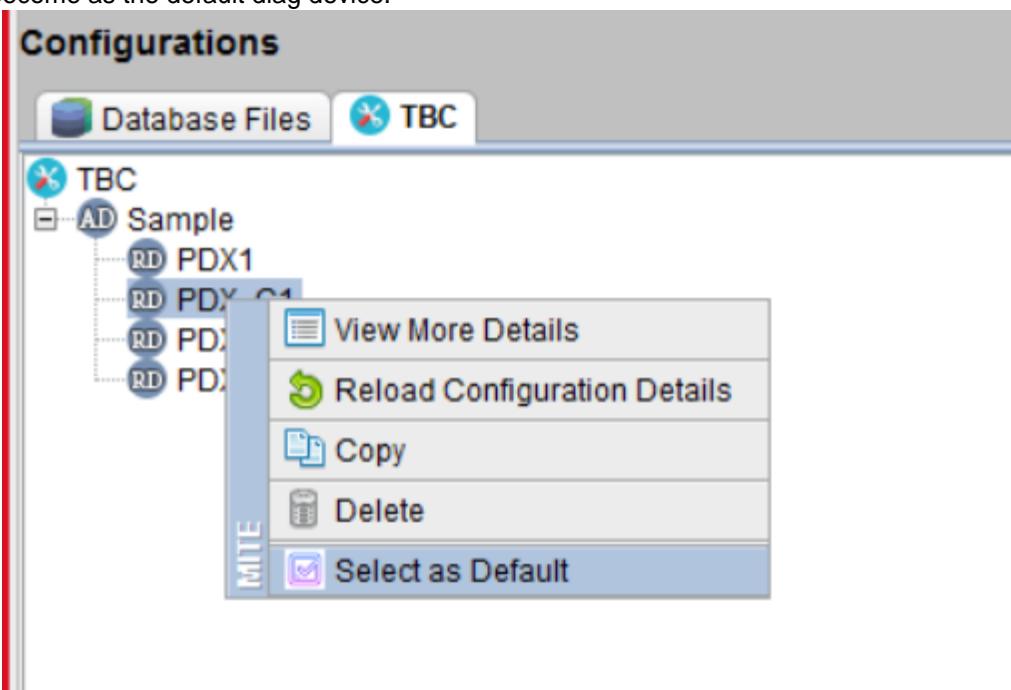
Diagnostic Device

Diag Device :

Note: This selection of Diag Device is applicable to the Diagnostic labels for which the mapping is not done. If mapping is done in the future, then the Diag device of the corresponding request path/response path will overwrite the current diag device.

- If user wants to give one Remote Diagnostics file as Default Diag Device for multiple labels, even MITE allows that also.

- Select the access file that you want to select as default in the configuration panel.
- Right click on that and select the option “Select as Default”. With this, this diag device which is selected will become as the default diag device.



Priority for Diag Device Selection from MITE:

- 1) If label is mapped in the label mapping frame, the diag device from which the label is coming will be considered. **Please use Drag and Drop in the Label Mapping Frame to get the needed Diag Device. From combo box, we cannot know from which file the label is coming from.**
- 2) If no mapping is done then the selected Diag Device from the Label Details tab will be given preference.
- 3) If no Diag device is selected from the Label Mapping Frame(mentioned in the 2nd point) then the default selected access from the tree will be given priority.
- 4) If any of the above 3 points are not followed, then MITE will automatically consider the first device from the tree as the default device.

Created with the Personal Edition of HelpNDoc: [Easily create Help documents](#)

19.12 SDF Extraction of Arrayed Labels

SDF Extraction of Arrayed Labels

From version MITE – V2.1.5.0, arrayed labels are also available in the mapping frame. Up to now only the main parent label is available whereas user cannot map the arrayed labels. The following are the rules on how user can give arrayed SDF labels.

- 1) These SDF arrayed labels are not available in the mapping frame tree or the option list as there are large number of SDF arrayed labels (in lakhs) depending up on the SDF file given as input.
- 2) The User can give the arrayed SDF label as free text. If that is valid label, then that will not be vanished from the table.
- 3) The user has to enter the arrayed label with in the correct range. For example, if the array size of a label “X” is 2, then user can give the labels X, X[0], X[1] in the mapping frame. All the other labels are not allowed and will be vanished immediately after giving in the mapping frame.
- 4) The same rules apply for the double arrayed labels as well. For example, if the allowed size of a label is 2x2. The labels that are allowed are: X, X[0][0], X[0][1], X[1][0], X[1][1].
- 5) No other special characters are allowed except square brackets (“[“) while defining array size of the arrayed labels.

Read Path	Access Type
Model Root\ADCAM_Features\Automatic Emergency Braking\1-D Lookup\InTable\table>Data[13]	Model Access
Model Root\ADCAM_Features\Automatic Emergency Braking\1-D Lookup\InTable\table>Data[12]	Model Access
Model Root\ADCAM_Features\Automatic Emergency Braking\1-D Lookup\InTable\table>Data[1]	Model Access
Model Root\MDU\Drivetrain\Final_Drive_Assembly\STABILIZATION\DRIVE\TRAIN_STABILIZATION\AND_FWD\RWD\StabilizationMatrix\Trq_dOmega_Matrix\Matrix\Concatenation1\Out1[1][1]	Model Access

From the Version v2.1.8.0, all the SDF labels will have the prefix "**(:)//"**. To see this change, all the older SDF files that are imported before the version v2.1.8.0 should be reloaded. With this reload, the already used labels in the label mapping frame will be auto updated except the labels in the specific mapper. The SDF labels from the specific mapper should be remapped manually.

#	Test Labels	Read Path	Access Type
1	M ACCAxlTrqCmd.ACCL...	(:)//BusSystems/CAN/ControllerSetup_Blocks/HECAN/Monitoring_Enable	Model Access
2	M ACCAxlTrqCmd.ACCL...	(:)//BusSystems/CAN/ControllerSetup_Blocks/HECAN/Monitoring_TimeStamp	Model Access
3	U CIB_Autonomous_Bra...	(:)//BusSystems/CAN/ControllerSetup_Blocks/HECAN/Variation	Model Access
4	U CIB_Autonomous_Bra...	(:)//BusSystems/CAN/ControllerSetup_Blocks/HECAN/Variation_Switch	Model Access
5	U CIB_Autonomous_Bra...	(:)//BusSystems/CAN/ControllerSetup_Blocks/LSCAN/Monitoring_Enable	Model Access

Created with the Personal Edition of HelpNDoc: [Write EPub books for the iPad](#)

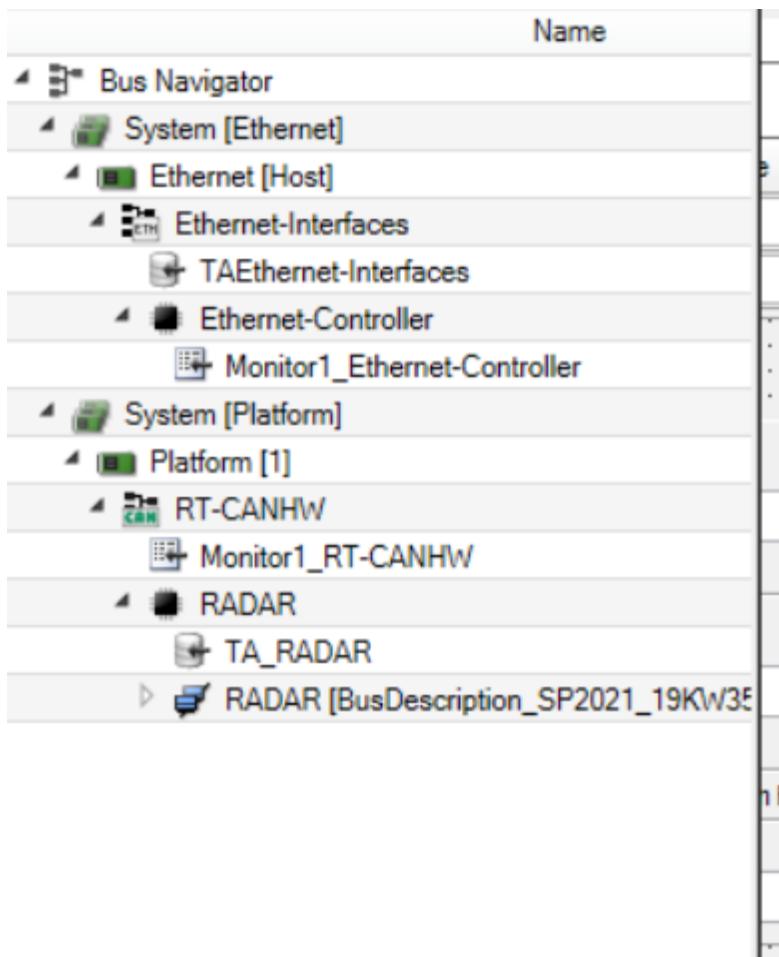
19.13 LOGGERS FROM CONTROL DESK

LOGGERS FROM CONTROL DESK

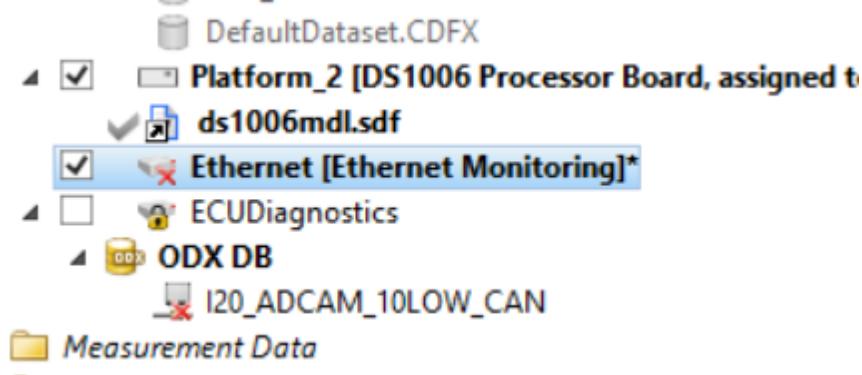
Access Type : Model Access

Tool-Interface : Control Desk

1. Configured Ethernet and CAN bus type (eg : RT –CANHW) in Control Desk.



2. Enabled Ethernet Monitoring at project level.



3. Activate Monitors for both Ethernet-Interface and RT-CANHW

Decoding View						
Name	Source MAC	Destination MAC	EtherType	Frame Length	802.1Q Tag	Ethernet Payload
Bus Navigator	02:84:CF:3B:BE:1A	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 106)		Priority:.. 00 49 08 00 45 00 00 54 08 18 00 00..	
System [Ethernet]	02:84:CF:3B:BE:26	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 106)		Priority:.. 00 49 08 00 45 00 00 54 08 19 00 00..	
Ethernet [Host]	02:84:CF:3B:BE:24	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 978)		Priority:.. 00 49 08 00 45 00 03 BC 07 97 00 00..	
Ethernet-Interfaces	02:84:CF:3B:BE:2A	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 214)		Priority:.. 00 49 08 00 45 00 00 C0 08 1A 00 00..	
Monitor1_Ethernet-Interfaces	02:84:CF:3B:BE:08	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 134)		Priority:.. 00 49 08 00 45 00 00 70 08 1C 00 00..	
Ethernet-Controller	02:84:CF:3B:BE:1C	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 106)		Priority:.. 00 49 08 00 45 00 00 54 08 1D 00 00..	
Monitor1_Ethernet-Controller	02:84:CF:3B:BE:1E	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 106)		Priority:.. 00 49 08 00 45 00 00 54 08 1E 00 00..	
System [Platform]	02:84:CF:3B:BE:1A	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 106)		Priority:.. 00 49 08 00 45 00 00 54 08 1B 00 00..	
Platform [1]	02:84:CF:3B:BE:26	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 106)		Priority:.. 00 49 08 00 45 00 00 54 08 1C 00 00..	
RT-CANHW	02:84:CF:3B:BE:2A	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 214)		Priority:.. 00 49 08 00 45 00 00 C0 08 1D 00 00..	
Monitor1_RT-CANHW	02:84:CF:3B:BE:24	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 978)		Priority:.. 00 49 08 00 45 00 03 BC 07 98 00 00..	
RADAR	02:84:CF:3B:BE:08	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 134)		Priority:.. 00 49 08 00 45 00 00 70 08 1F 00 00..	
TA_RADAR	02:84:CF:3B:BE:1C	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 106)		Priority:.. 00 49 08 00 45 00 00 54 08 20 00 00..	
RADAR	02:84:CF:3B:BE:1E	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 106)		Priority:.. 00 49 08 00 45 00 00 54 08 21 00 00..	
TA_RADAR	02:84:CF:3B:BE:26	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 106)		Priority:.. 00 49 08 00 45 00 00 54 08 1F 00 00..	
RADAR	02:84:CF:3B:BE:2A	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 214)		Priority:.. 00 49 08 00 45 00 00 C0 08 20 00 00..	
TA_RADAR	02:84:CF:3B:BE:24	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 978)		Priority:.. 00 49 08 00 45 00 03 BC 07 99 00 00..	
RADAR	02:84:CF:3B:BE:08	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 134)		Priority:.. 00 49 08 00 45 00 00 70 08 22 00 00..	
TA_RADAR	02:84:CF:3B:BE:1C	01:00:5E:40:FF:FB	IEEE 802.1Q (.. 106)		Priority:.. 00 49 08 00 45 00 00 54 08 23 00 00..	

4. Start execution from Automation Desk.

NOTE: User needs to Import MITE libraries.

Created with the Personal Edition of HelpNDoc: [Free Kindle producer](#)

19.14 CDD at TestScript Level

CDD at TestScript Level

User can now import CDD into TBC and it should be accessed through Remote Diagnostics as shown in Fig1.

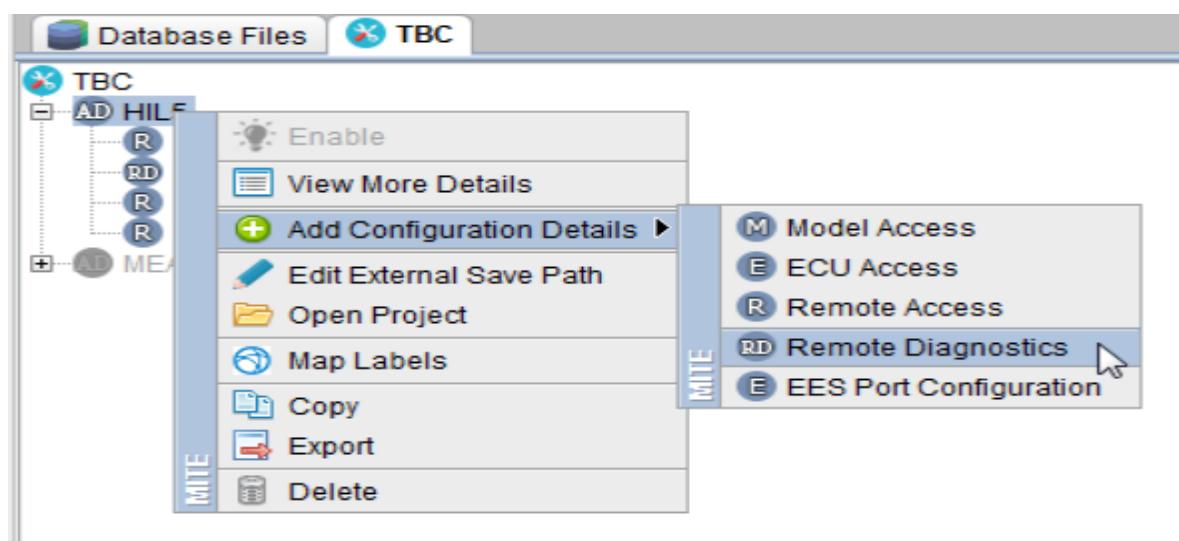


Fig1. Accessing CDD through Remote Diagnostics

After selecting Remote Diagnostics option a Remote Diagnostic Dialog will be displayed as shown in Fig2. In that user need to enter shortname, select required tool interface, select required File Type from combo box (in our case it is CDD Files) and enter File Path.

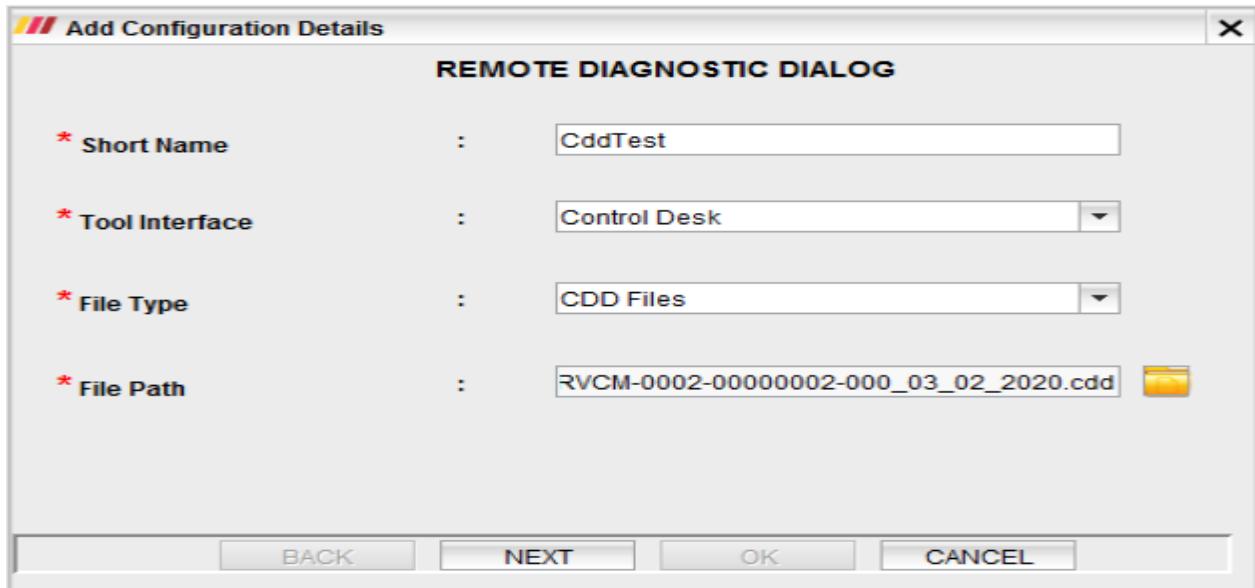


Fig2. Importing of Cdd File

After proper filling, click on NEXT button. Here we need to enter Service CSV file and Data CSV file in their appropriate fields, select Base Variant Name from combo box and click OK as shown in Fig3.

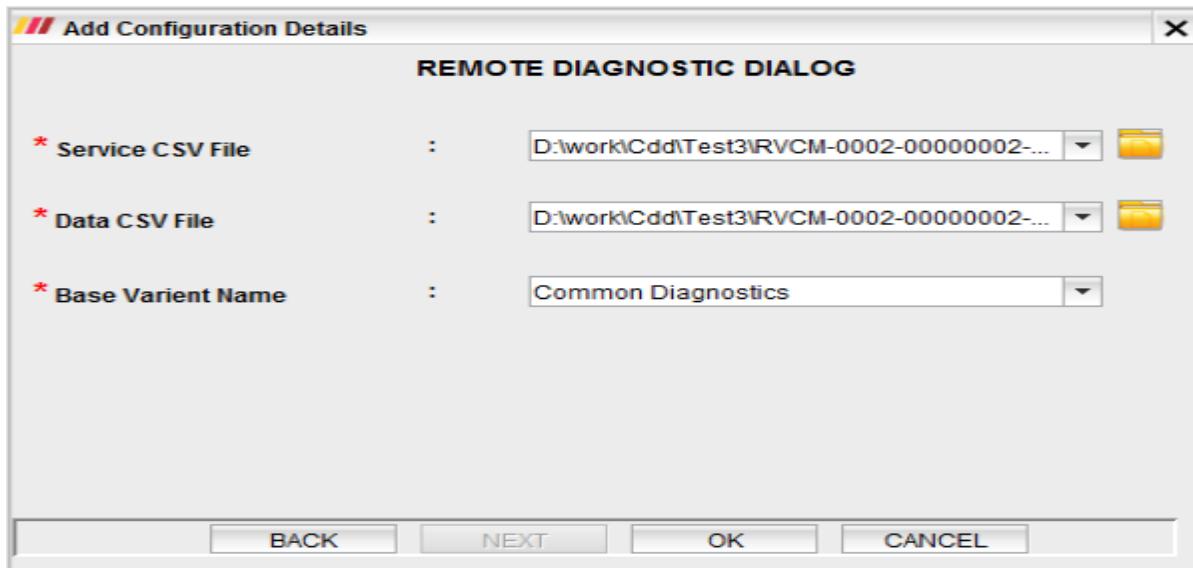


Fig3. Importing of Service and Data CSV files

What are Service and Data CSV Files:

These are CSV files exported from CANdela studio containing information about Services and Data Objects. When opened cdd files in Vector CANdelaStudio we can find Service Overview and Data Overview options under Data Exchange tab as shown in Fig4. Upon clicking them we will get Service CSV file and Data CSV file for particular cdd file. These

files are required for importing Cdd into Label Database and TBC.



Fig4. Exporting of Service and Data CSV files

This is the process of importing CDD into TBC for the tool interface Control Desk. But for Canoe tool interface the process is slightly different. For Canoe Tool interface after filling the details an additional dialog box will be displayed as shown in Fig5.

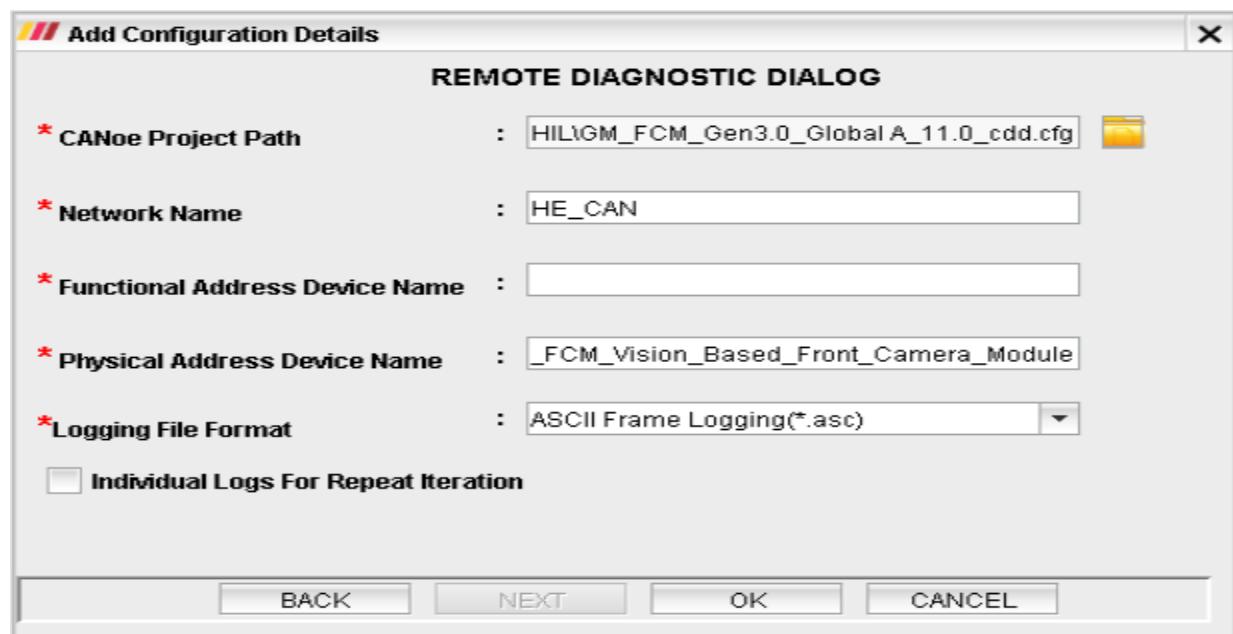


Fig5. Importing Canoe Configuration File.

We need to enter Canoe configuration file, select the Logging File format and click ok.

After this we can map the cdd diagnostic labels with diagnostic labels written in test case editor.

Created with the Personal Edition of HelpNDoc: [Easy EPub and documentation editor](#)

19.15 Hyperlink In Warning Tab

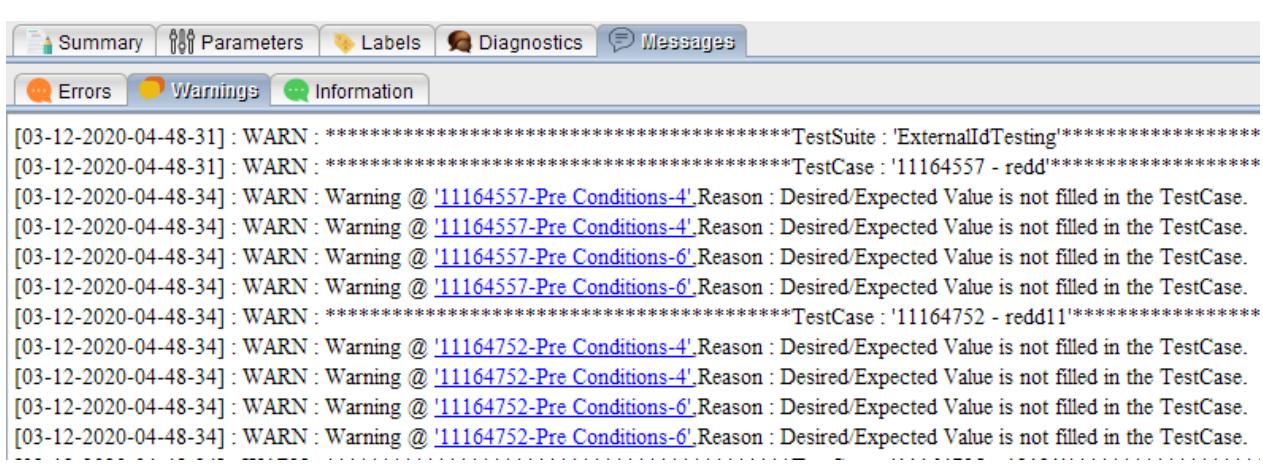
With the new version of MITE user will be able to get Hyperlinks in Warning tab after script generation as follows:-

- 1) User need to generate the scripts by using Test Script generation option.
- 2) After scripts gets generated user can switch to warning tabs to get to know which test steps have

warnings.

3)In each warnings hyperlink will be generated so user can click on it and it will traverse to the particular test step.

4)User can modify the changes according to the requirement and generate scripts again.



```
[03-12-2020-04-48-31] : WARN : *****TestSuite : 'ExternalIdTesting'*****
[03-12-2020-04-48-31] : WARN : *****TestCase : '11164557 - redd'*****
[03-12-2020-04-48-34] : WARN : Warning @ '11164557-Pre Conditions-4' Reason : Desired/Expected Value is not filled in the TestCase.
[03-12-2020-04-48-34] : WARN : Warning @ '11164557-Pre Conditions-4' Reason : Desired/Expected Value is not filled in the TestCase.
[03-12-2020-04-48-34] : WARN : Warning @ '11164557-Pre Conditions-6' Reason : Desired/Expected Value is not filled in the TestCase.
[03-12-2020-04-48-34] : WARN : Warning @ '11164557-Pre Conditions-6' Reason : Desired/Expected Value is not filled in the TestCase.
[03-12-2020-04-48-34] : WARN : *****TestCase : '11164752 - redd11'*****
[03-12-2020-04-48-34] : WARN : Warning @ '11164752-Pre Conditions-4' Reason : Desired/Expected Value is not filled in the TestCase.
[03-12-2020-04-48-34] : WARN : Warning @ '11164752-Pre Conditions-4' Reason : Desired/Expected Value is not filled in the TestCase.
[03-12-2020-04-48-34] : WARN : Warning @ '11164752-Pre Conditions-6' Reason : Desired/Expected Value is not filled in the TestCase.
[03-12-2020-04-48-34] : WARN : Warning @ '11164752-Pre Conditions-6' Reason : Desired/Expected Value is not filled in the TestCase.
-----
```

Created with the Personal Edition of HelpNDoc: [Free EPub and documentation generator](#)

19.16 Multiselection of Test Cases in Test Script Viewset

Multiselection of Test Cases in Test Script Viewset

Description:-

Previously when User used to execute the test cases, he/she will check individual test cases or select all test cases based on the test case requirement. In addition to that from MITE -v ,We have enhanced a new feature i.e. Multiselection of test cases either in random wise or order wise for smooth operation while execution of test cases. This can be done in following ways:-

1)Using right click action for multiselection of test cases where we can check the selected test case as well as vice versa.:-

Test Suite : 13110905								
Created By : gurdsaho on May 26, 2021 4:13:56 PM								
Modified By : gurdsaho on Sep 30, 2021 1:42:51 PM								
Summary Parameters Labels Diagnostics Messages								
#	All	TestCase ID	Category	Description	Script Name	Automation Sta...	Shared Id	Share
0	<input checked="" type="checkbox"/>	13110907	Heading	AEB_PFGS				
1	<input checked="" type="checkbox"/>	13110909	Heading	Place Holders				
2	<input checked="" type="checkbox"/>	13110901	System Test	PreCondition_DTC	Open			
3	<input checked="" type="checkbox"/>	13110903	System Test	PostCondition_	View More Details	TC_13110903.sqx	Semi Automated	
4	<input checked="" type="checkbox"/>	13110905	System Test	6829025_Rem	Check Selected TestCases			
5	<input checked="" type="checkbox"/>	13110907	System Test	6829027_Degr	UnCheck Selected TestCases	TC_13110907.sqx	Semi Automated	
6	<input checked="" type="checkbox"/>	13110901	System Test	6829031_Degradation_High				
7	<input checked="" type="checkbox"/>	13110903	System Test	8760180_Degradation_Critical	TC_13110903.sqx	Semi Automated		
8	<input checked="" type="checkbox"/>	13110905	System Test	6759610_Radar_Blindness				
9	<input checked="" type="checkbox"/>	13110907	System Test	6759612_Camera_blind				
10	<input checked="" type="checkbox"/>	13110909	System Test	6829021_Remove Fault Radar Blindness	TC_13110909.sqx	Semi Automated		
11	<input checked="" type="checkbox"/>	13110911	System Test	8776804_Remove DTC for Degradation_More Criti...	TC_13110911.sqx	Semi Automated		
12	<input checked="" type="checkbox"/>	13110913	System Test	8879185_Remove DTC for Active degraded state D...				
13	<input checked="" type="checkbox"/>	13110915	System Test	8879187_Remove DTC for Error state Degradation ...				
14	<input checked="" type="checkbox"/>	13110917	System Test	9006147_Remove DTC for Error state Degradation ...	TC_13110917.sqx	Semi Automated		
15	<input checked="" type="checkbox"/>	13110919	System Test	9820275_Degradation_More Critical	TC_13110919.sqx	Semi Automated		
16	<input checked="" type="checkbox"/>	13110921	System Test	PFGS_CLH_Precondition	TC_13110921.sqx	Semi Automated		
17	<input checked="" type="checkbox"/>	13110923	System Test	PPA_PPA_PPA_PPA_PPA_PPA_PPA_PPA_PPA_PPA_PPA_P...				

2) Using Control key from the keypad we can select multiple test case in order wise or random wise test case and we can right click and use the check/uncheck selected test cases:-

#	All	TestCase ID	Category	Description	Script Name	Automation Sta...	Shared Id	Share
0	<input type="checkbox"/>	13110904	Comment	Comments other than the requirement development in...				
1	<input checked="" type="checkbox"/>	13110907	System Test	To verify the function shall implement a diagnostics ...		Medium	draft	
2	<input type="checkbox"/>	13110908	Heading	Degradation and DTC		Medium	draft	
3	<input checked="" type="checkbox"/>	13110909	System Test	The DTC 0xBA0AC9 (Function PFGS not available)...		Low	draft	
4	<input checked="" type="checkbox"/>	13110905	System Test	The DTC 0xBA0AC9 (Function PFGS temporary not ...		Low	draft	
5	<input checked="" type="checkbox"/>	13110907	System Test	The DTC 0xBA0AC8 (Function PFGS not available)...		Low	draft	
6	<input checked="" type="checkbox"/>	13110909	System Test	The DTC 0xBA0AC8 (Function PFGS temporary not ...		Low	draft	
7	<input checked="" type="checkbox"/>	13110910	System Test	The DTC 0xBA0AC9 (Function PFGS not available)...		Low	draft	
8	<input checked="" type="checkbox"/>	13110910	System Test	The DTC 0xBA0AC9 (Function PFGS temporary not ...		Low	draft	
9	<input type="checkbox"/>	13110905	System Test	The DTC 0xBA0AC8 (Function PFGS not available)...		Low	draft	
10	<input checked="" type="checkbox"/>	13110910	System Test	The DTC 0xBA0AC8 (Function PFGS temporary not ...		Low	draft	
11	<input type="checkbox"/>	13110910	Heading	FZDs combination		Low	draft	
12	<input type="checkbox"/>	13110911	System Test	To check for the DTC combination when first Degr...		Low	draft	

3)By using Shift key we can we can keep a selection of test cases from the required point to the other point of test cases and then we can click right click and use the option check the selected test cases and vice versa:-

#	All	TestCase ID	Category	Description	Script Name	Automation Sta...	Shared Id	Share
1	<input checked="" type="checkbox"/>	131109121	System Test	To check for the DTC combination when first Degr...		Low	draft	
2	<input checked="" type="checkbox"/>	131109123	System Test	To check for the DTC combination when first Degr...		Low	draft	
3	<input checked="" type="checkbox"/>	131109126	System Test	To check for the DTC combination when first Degr...		Low	draft	
4	<input checked="" type="checkbox"/>	131109128	System Test	To check for the DTC combination when first Degr...		Low	draft	
5	<input checked="" type="checkbox"/>	131109128	Heading	States	Open			
6	<input checked="" type="checkbox"/>	131109140	System Test	Unavail	View More Details			
7	<input checked="" type="checkbox"/>	131109142	System Test	Run_S	Check Selected TestCases	Critical	draft	
8	<input checked="" type="checkbox"/>	131109144	System Test	Run_E	UnCheck Selected TestCases	Critical	draft	
9	<input checked="" type="checkbox"/>	131109146	System Test	Run_Disable_State_behavior		Critical	draft	
10	<input checked="" type="checkbox"/>	131109148	System Test	Run_Error_State_behavior_critical		Critical	draft	
11	<input checked="" type="checkbox"/>	131109150	System Test	Run_Active_Degraded_State_behavior_Low		Critical	draft	
12	<input checked="" type="checkbox"/>	131109152	System Test	Run_Active_Degraded_State_behavior_Medium		Critical	draft	
13	<input checked="" type="checkbox"/>	131109154	System Test	Disable_Degraded_State_behavior		Critical	draft	
14	<input type="checkbox"/>	131109156	Heading	Active State				
15	<input checked="" type="checkbox"/>	131109158	System Test	To verify the function PFGS shall stay in active state ...		Critical	draft	
16	<input checked="" type="checkbox"/>	131109160	System Test	To verify the function PFGS shall stay in active state ...		Critical	draft	
17	<input checked="" type="checkbox"/>	131109162	System Test	To verify the function logic PFGS shall be in the stat...		Critical	draft	
18	<input checked="" type="checkbox"/>	131109164	System Test	Negative TC To verify the function PFGS shall not st...		Critical	draft	

4) Using of Shift + Control we can able to select multiple number of test cases in Random wise order.

Test Suite - L2G8810_BMW-LIDAR_SysTe...

PTC ID : 9050215

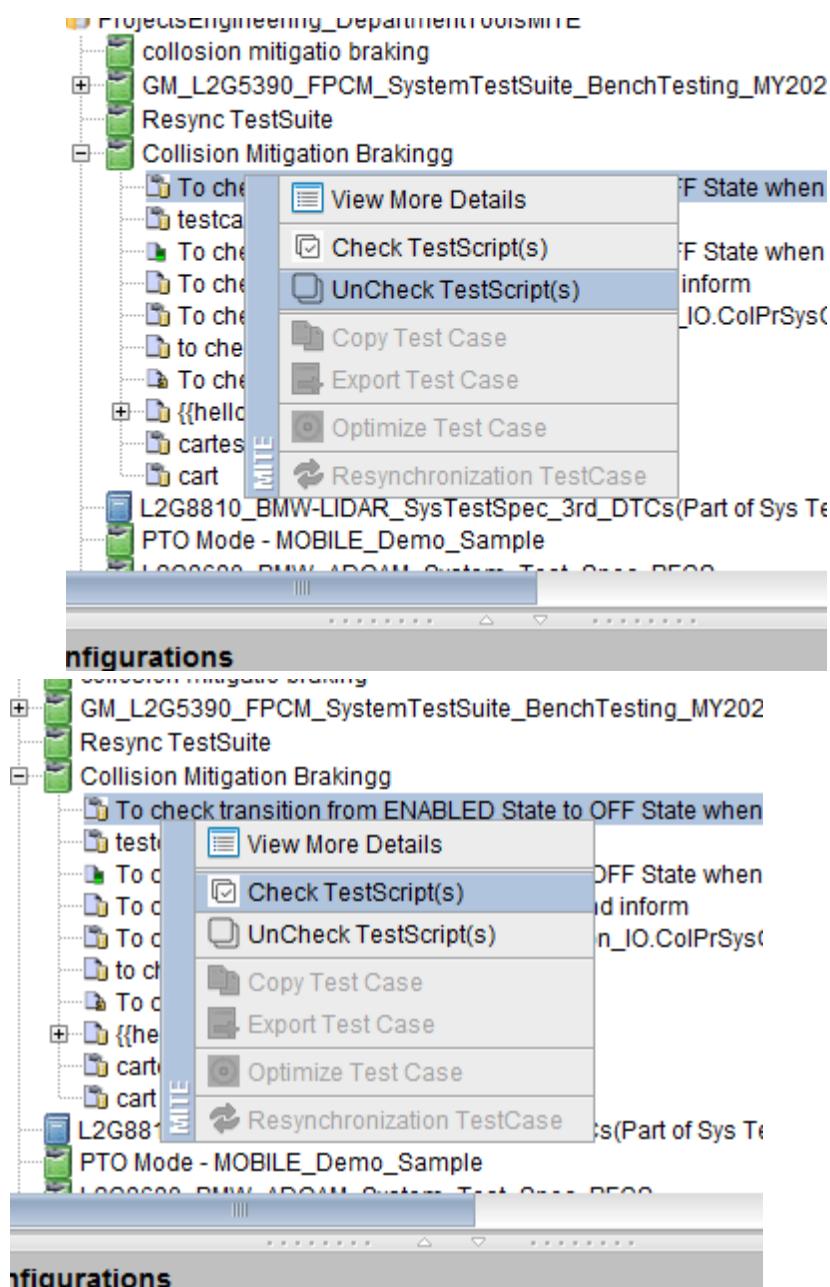
TESTSUITE

#	All	TestCase ID	Category	Description	Script Name	Automation Sta...	Shared Id	Shared TestSuite Id	Shared Test...	Valid For Pro...	Priority	Status	variant
1		5608164	Heading	Signal (0x5531-0x0001-BMW BODY-Wiper) invalid...				L2G8810	High	reviewed		draft	
2		5608185	System Test	a) Create Fault-Signal (0x5531-0x0001-BMW BODY...				L2G8810	High	reviewed			
3		5608187	System Test	b) Read Active DTC- Signal (0x5531-0x0001-BMW...				L2G8810	High	reviewed			
4		5608189	System Test	c) Remove Fault-Signal (0x5531-0x0001-BMW BO...				L2G8810	High	reviewed			
5		5608191	System Test	d) Read Inactive DTC- Signal (0x5531-0x0001-BMW...				L2G8810	High	reviewed			
6		5608193	System Test	e) Read DTC not in NVM- Signal (0x5531-0x0001-B...				L2G8810	High	reviewed			
7		5608176	System Test	1. Normal: Signal (0x5531-0x0001-BMW BODY-Wip...				L2G8810	High	reviewed			
8		5608168	System Test	2. Below Min voltage: Signal (0x5531-0x0001-BMW...				L2G8810	Medium	reviewed			
9		5608170	System Test	3. Above Max voltage: Signal (0x5531-0x0001-BMW...				L2G8810	Medium	reviewed			
10		5608172	System Test	4. ECU Power ON and Diag Job Reset: Signal (0x5...				L2G8810	Medium	reviewed			
11		5608166	System Test	5. Healing cycle (Warmup cycle): Signal (0x5531-0x...				L2G8810	Medium	reviewed			
12		5608178	System Test	7. Fe Mode : Signal (0x5531-0x0001-BMW BODY-Wi...				L2G8810	Medium	reviewed			
13		5608181	System Test	8.Tra Mode : Signal (0x5531-0x0001-BMW BODY-Wi...				L2G8810	Medium	reviewed			
14		5756973	System Test	9.Fla Mode : Signal (0x5531-0x0001-BMW BODY-Wi...				L2G8810	Medium	reviewed			
15		5756975	System Test	10. Transition from Fe to Tra mode : Signal (0x5531...				L2G8810	Medium	reviewed			
16		5756977	System Test	11. Transition from Fla to Fe mode : Signal (0x5531...				L2G8810	Medium	reviewed			
17		5756979	System Test	12.Transition from Tra to Fla mode: Signal (0x5531...				L2G8810	Medium	reviewed			
18		5756981	System Test	13.Transition Fla to Tra mode: Signal (0x5531-0x0...				L2G8810	Medium	reviewed			
19		7142321	System Test	14.Transition from Fe to Fla mode: Signal (0x5531...				L2G8810	Medium	reviewed			
20		7142323	System Test	15. Transition from Tra to Fe mode: Signal (0x5531...				L2G8810	Medium	reviewed			
21		8270749	System Test	16.Fahrberbeitschaft_beenden mode: Signal (0x553...				L2G8810	High	reviewed			
22		8270751	System Test	17.Fahrberbeitschaft_herstellen mode: Signal (0x55...				L2G8810	High	reviewed			
23		8270753	System Test	18.Prufen_Analyse_Diagnose mode: Signal (0x55...				L2G8810	High	reviewed			
24		8277191	Comment	28.Undervoltage 2 - DTC 0xEF5576 - Signal(0x3000...				L2G8810		reviewed			
25		8277193	Comment	29.Undervoltage 3 - DTC 0xEF5576 - Signal(0x3000...				L2G8810		proposed			
26		8277195	System Test	30.Undervoltage 4 - DTC 0xEF5576 - Signal(0x3000...				L2G8810		reviewed			
27		8277197	System Test	31.Undervoltage 5 - DTC 0xEF5576 - Signal(0x3000...				L2G8810		reviewed			

5)Using right click action for multiselection of test cases where we can check the selected test case as well as vice versa.

#	All	TestCase ID	Category	Description	Script Name	Automation Sta...	Shared Id	Shared TestSuite Id	Shared Test...	Valid For Pro...	Priority	Status	variant
1	<input type="checkbox"/>		System T...	Send Dia...	TC_NEW...	Fully Auto...						draft	
2	<input checked="" type="checkbox"/>		System T...	Send Dia...	TC_NEW...	Semi Aut...						draft	
3	<input checked="" type="checkbox"/>		System T...	Send Dia...	TC_NEW...	Fully Auto...						draft	
4	<input checked="" type="checkbox"/>		System T...	Send Dia...	TC_NEW...	Fully Auto...						draft	
5	<input checked="" type="checkbox"/>		System T...	Send Dia...	TC_NEW...	Fully Auto...						draft	
6	<input checked="" type="checkbox"/>		System T...	Send Dia...	TC_NEW...	Fully Auto...						draft	
7	<input checked="" type="checkbox"/>		System T...	Send Dia...	TC_NEW...	Fully Auto...						draft	
8	<input type="checkbox"/>		System T...	Send Dia...								draft	
9	<input type="checkbox"/>		System T...	Send Dia...								proposed	

TS-Rightclick option Multi Selection Scenarios Images



Created with the Personal Edition of HelpNDoc: [Free PDF documentation generator](#)

20 MITE LabVIEW

MITE_LabVIEW is used to generate LabVIEW scripts Vis based on Magna TAF architecture.

Created with the Personal Edition of HelpNDoc: [Easily create Help documents](#)

20.1 Pre-requisites

1. LabVIEW 2012 or above should be installed.
2. Create mks sandbox using PTC for respective Project files within c: drive only.
3. Using TAF, Respective project file should be generated within _mks sandbox.

Computer > Local Disk (C:) > _mks > Ford_360_L21301 > Systems > Test >			
	Name	Date modified	Type
nt Places			
ViewTestScript	10_SysTestPlan	10/7/2016 11:37 A...	File folder
uments	20_SysTestProcess	10/7/2016 11:37 A...	File folder
plate	30_SysTestSpecification	10/7/2016 11:38 A...	File folder
LabVIEW	40_SysTestRun	10/7/2016 11:38 A...	File folder
_LabVIEW_v1.0	50_SysTestObject	10/7/2016 11:38 A...	File folder
es	60_SysTestReport	10/7/2016 11:39 A...	File folder
uments	70_SysTestCase	11/29/2016 7:01 A...	File folder
c	80_SysTestTool	10/7/2016 11:43 A...	File folder
res	project.pj	7/6/2016 9:40 AM	PJ File

Figure 106: Ford360 project sandbox files

4. Through an executable file, by just double-clicking on the “MITE.exe” icon, MITE application can be installed on System successfully.
5. Assuming Test Cases are authored in MITE only.

Created with the Personal Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

20.2 Test Case Editor

Test case authoring
is same as MITE
Please refer
chapter 4 to 18.

Created with the Personal Edition of HelpNDoc: [Create HTML Help, DOC, PDF and print manuals from 1 single source](#)

20.3 Test Script

To start with Test script generation – User should select “**Test Script**” option from the list. LabVIEW Test Script Vis generation can be done automatically in MITE and the input it takes is MITE format Test cases only.

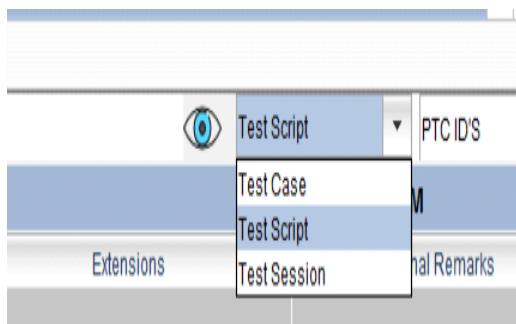


Figure 107: Select Test Script View Set

MITE script generation process mainly depends on the following elements:

- MITE Test cases
- Test bench configuration
- Related TAF library elements and
- Test data (Parameters, their values).

20.4 Test Bench Configuration

To start with Test Bench Configuration – User should select “**Test Script**” option.

In order to generate the test script which can be executed on a specific test bench, MITE needs to have test bench details. MITE has a separate window to add test bench configuration details.

Steps to add configuration details:

- Activate TBC window by Selecting Test Script from Eye icon on top right corner
- Then Right click on TBC node to add configuration as shown below

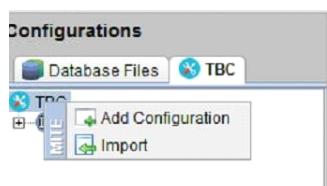


Figure 108: Add TBC Configuration

- Provide configuration details as bellow.
- Set Script Format – “LabVIEW” and Click OK

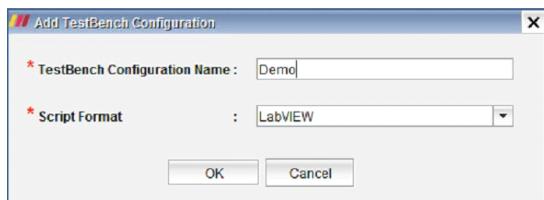


Figure 108 Add TBC Configuration

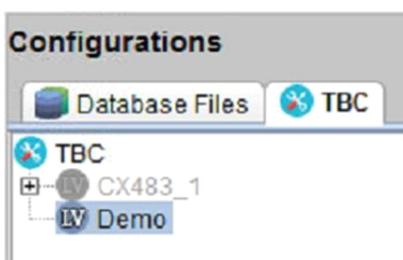


Figure 109: Configuration View after selecting script format

- For selecting Access type (like Remote Access or Remote diagnostics Access) and further Configuration details by Right click on TBC LV node and select “Add Configuration Details”

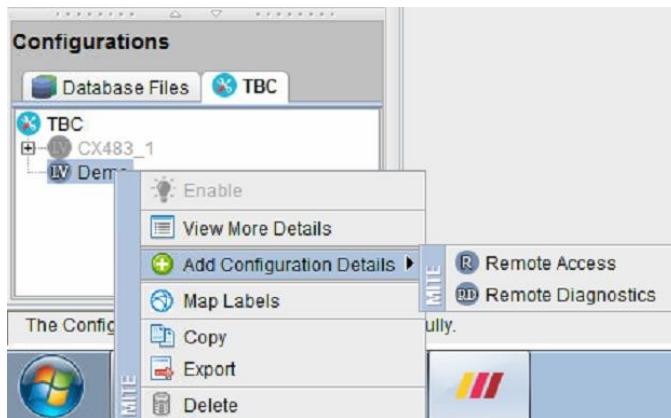


Figure 110: Add access details

Remote Access and Remote Diagnostics:

- Select Access type
- Provide file paths as shown below.
- Click OK

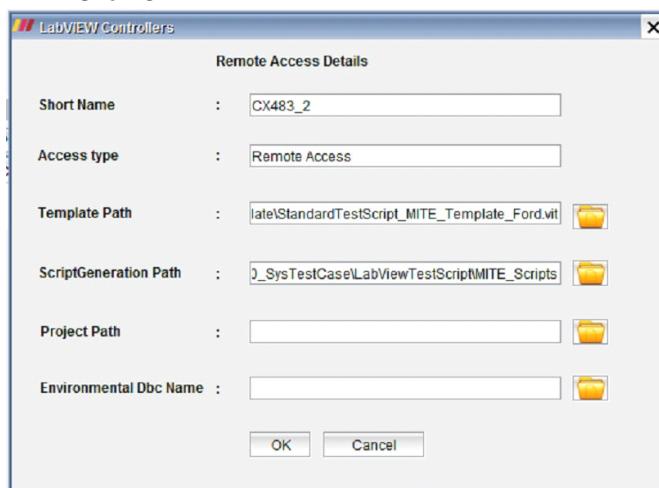


Figure 111: Remote Access MITE LabVIEW Configuration

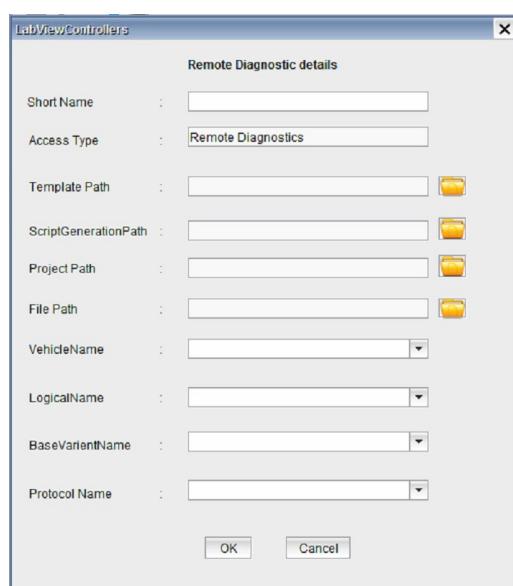


Figure 112: Remote Diagnostics MITE LabVIEW Configuration

20.5 Configuration Enable – Disable

User can choose between the added configurations using “Enable – Disable” on right click options as shown below

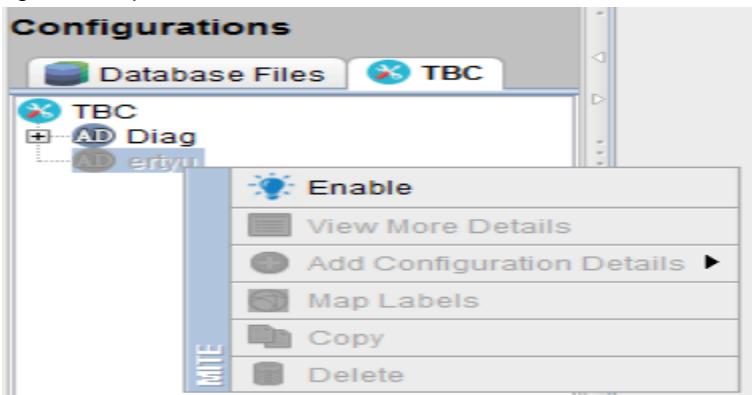


Figure 113: configuration Enable-Disable

20.6 Label Mapping

MITE shall import those files and create tool label database which later can be mapped to test labels used in test cases and create a configuration file.

- To view available labels in test suite , Right click on Configuration node and Select “Map Labels”

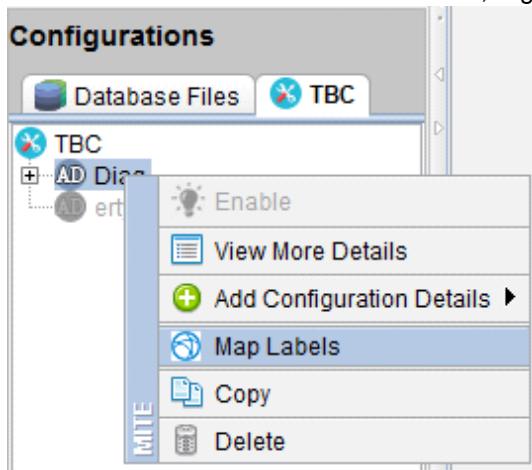


Figure 114: To Map a Label

- Label Mapping window appears as below :
 - Left side tree defines all the Label data base files imported in test suite
 - Test Labels Tab defines list of labels used in the test suite
 - Dependencies section allows user to add dependency labels if any otherwise can uncheck the option

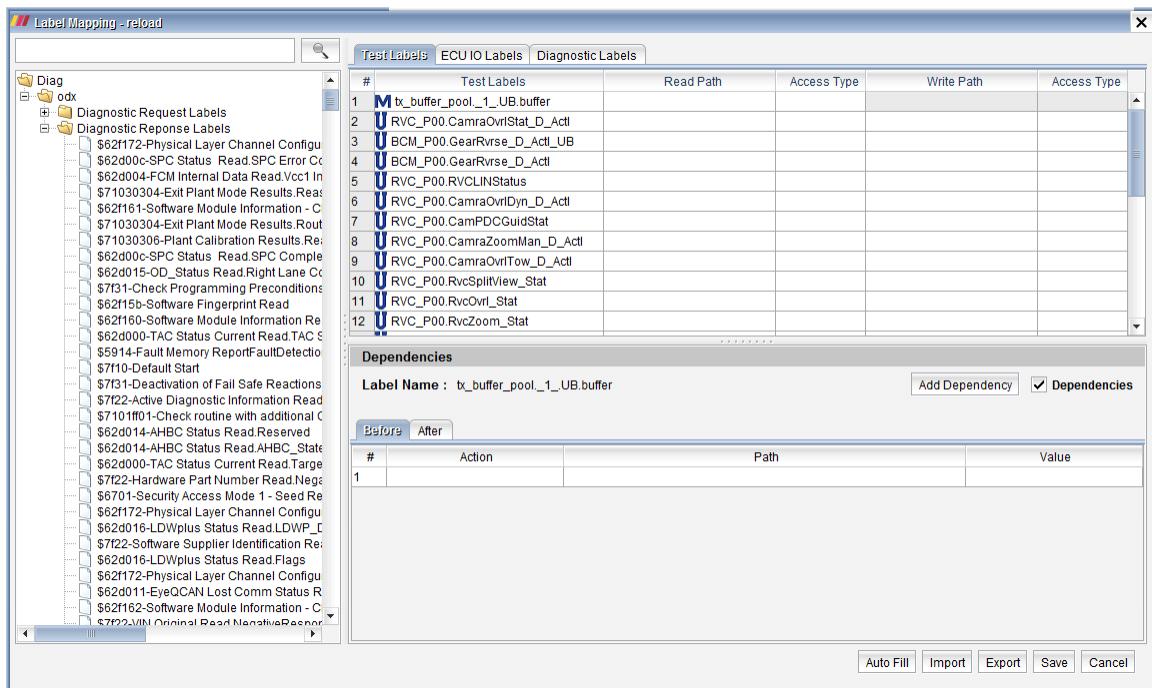


Figure 115: Label Mapping Window

- Mapping Labels can be done using “Drag and Drop” action
 - On drag and drop from tree to table gives “write path and read path” as shown below

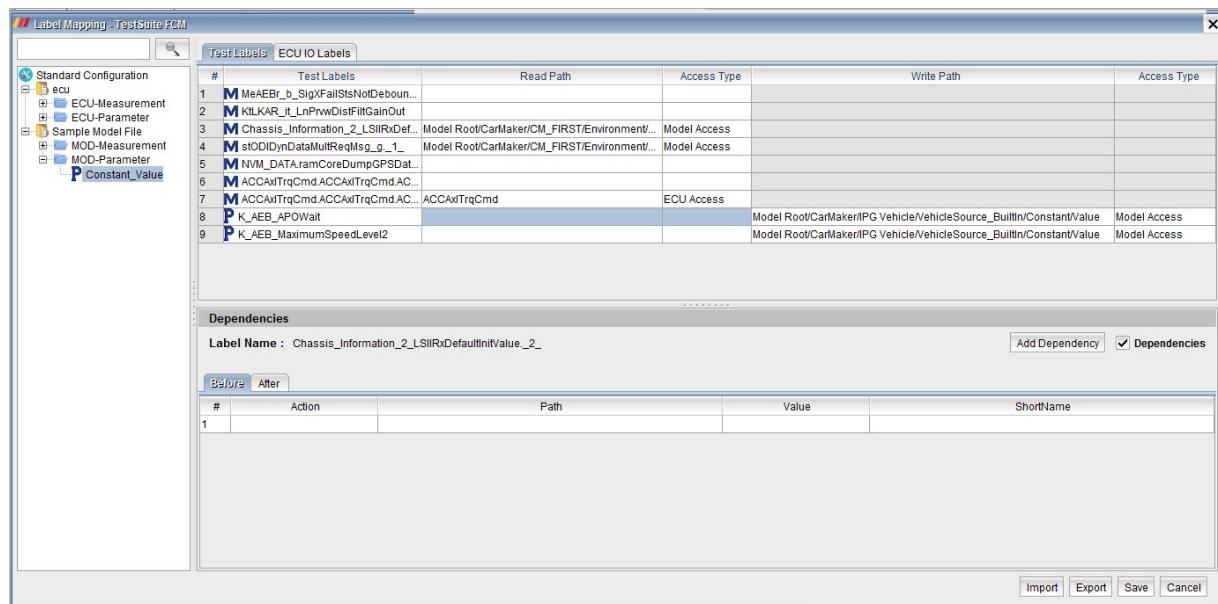


Figure 116: Read/Write path in Label Mapping

- Dependency Label Mapping

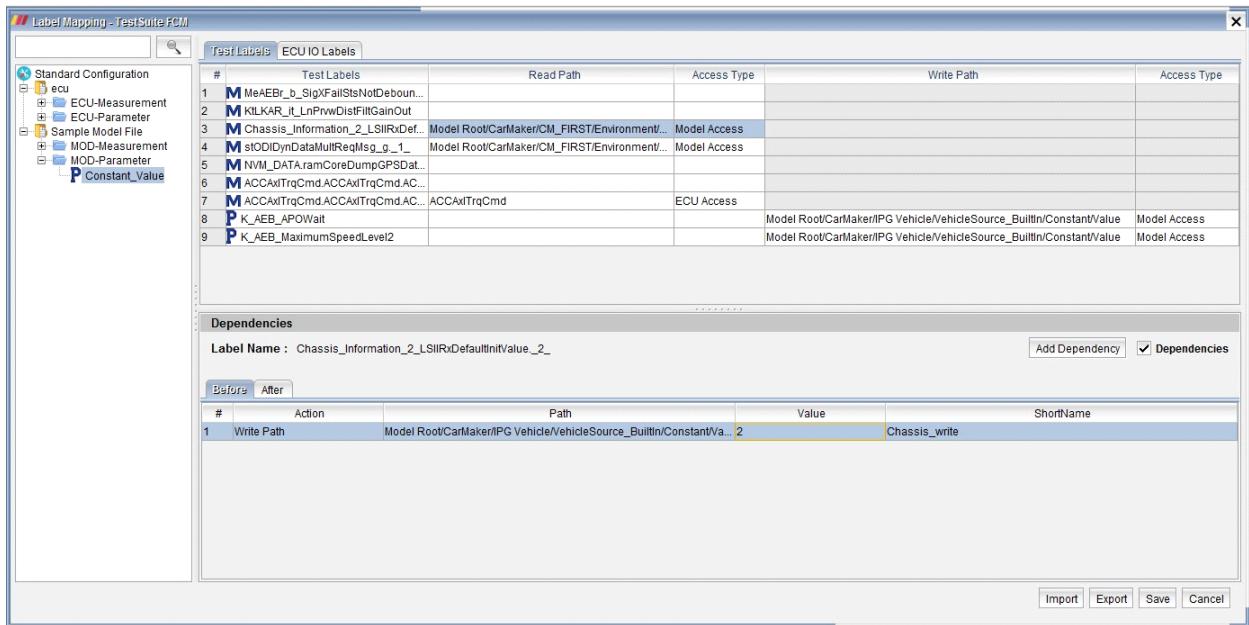


Figure 117: Dependency Label Mapping

- Label Filtering

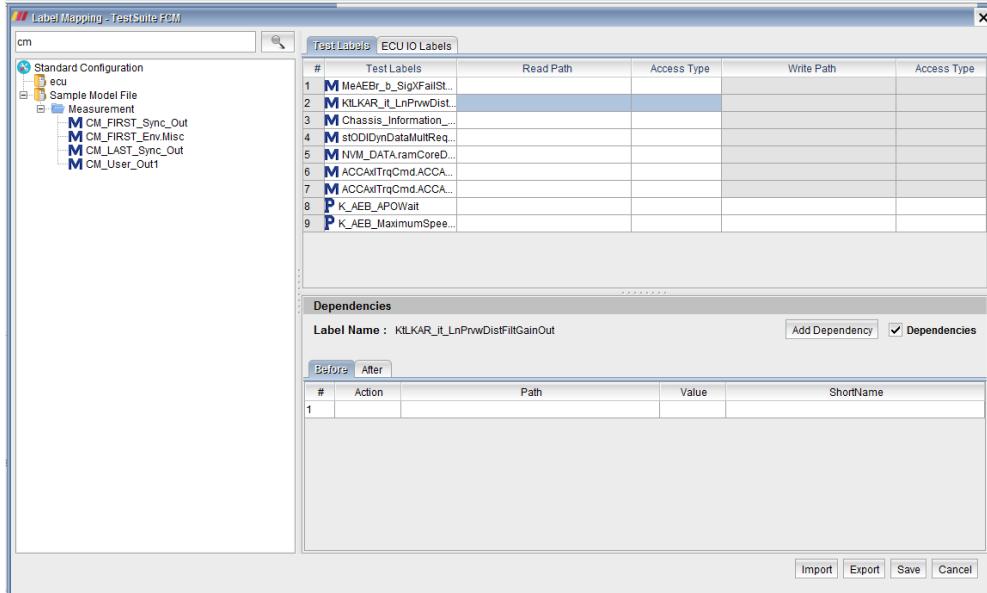


Figure 118: Filtering Label

- Label Mapping file Import/Export Feature

This feature is to import or export a Label mapping file which is already mapped or reuse of mapper files. After mapping respective read paths and write paths to all the labels, the data can be exported and imported in another configuration.

The below is the screenshot of the frame before mapping:

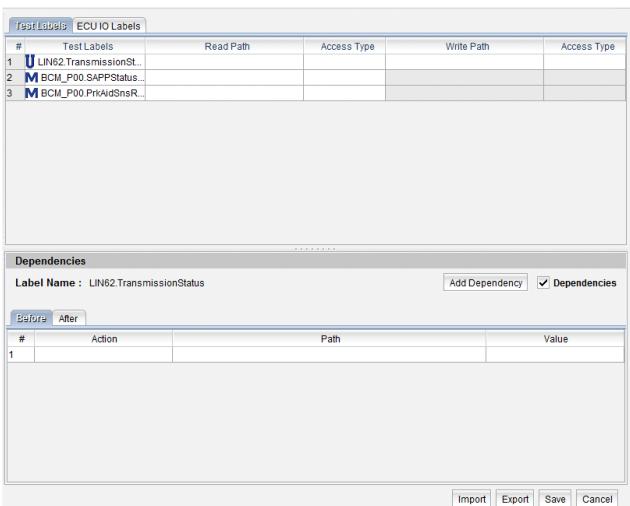


Figure 119: Before Mapping Label

Now, Click on import and give the exported label mapping file as zip. After importing the zip, the data of the current mapping tables gets overwritten by the label mapping from the zip file for the matched labels.

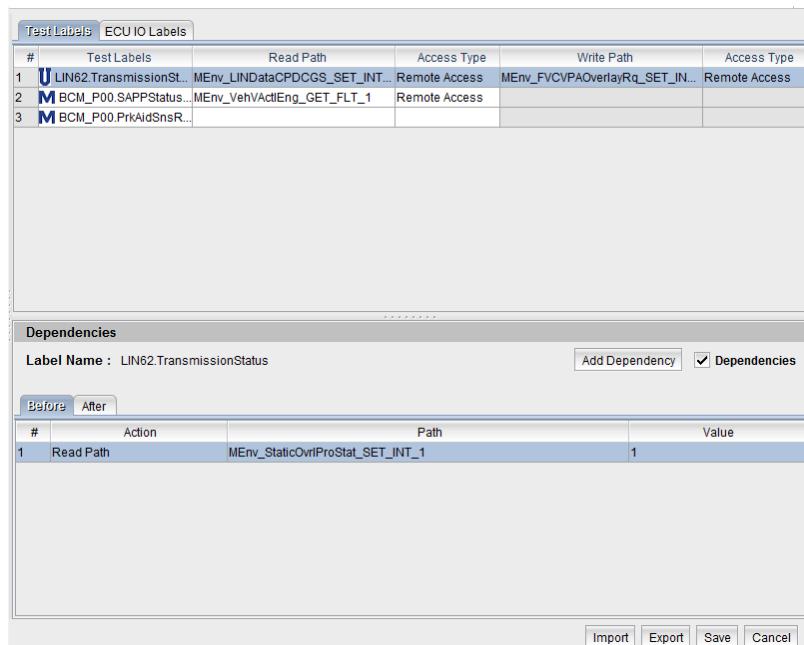


Figure 120: After Mapping Label

Created with the Personal Edition of HelpNDoc: [Free Web Help generator](#)

20.7 Test Script (LabVIEW Vis) Generation

To generate test script(s) in MITE, follow below steps:

- Make sure to change the View set to “Test Script” using eye icon
- Select a Test Suite under a Project
- Select require Configuration from “TBC window”
- Select required test cases to generate scripts
- Now click on “Test Script(s) Generation” icon  from tool bar to open MITE_LabVIEW Front Panel.

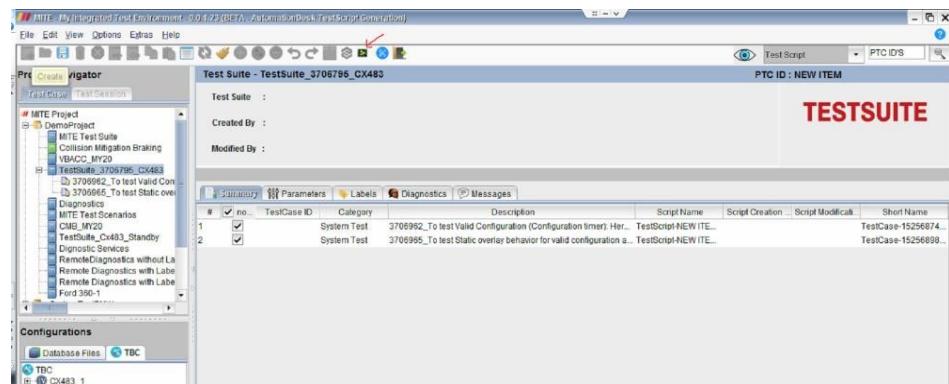


Figure 121 MITE Use Interface



Figure 122 MITE LabVIEW user interface

- You can verify selected test cases at MITE_LabVIEW interface.
- Click on “Generate script” button to generate LabVIEW scripts.
- Check the generation process in Script Generation Log
- LabVIEW script will be placed at script generation folder along with logs.
- We can use log files to identify errors.

Created with the Personal Edition of HelpNDoc: [Easily create HTML Help documents](#)

21 Other Additional Features

Created with the Personal Edition of HelpNDoc: [Easily create CHM Help documents](#)

21.1 MITE Demo Project :- Demo project is added to the list for the first time MITE users

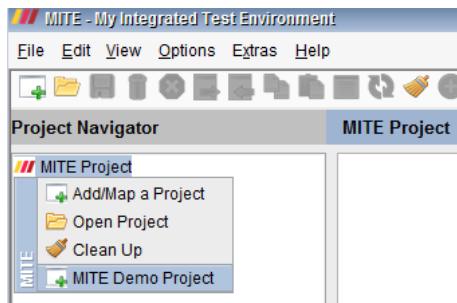


Figure 123: Right-Click on MITE Project for Demo Project

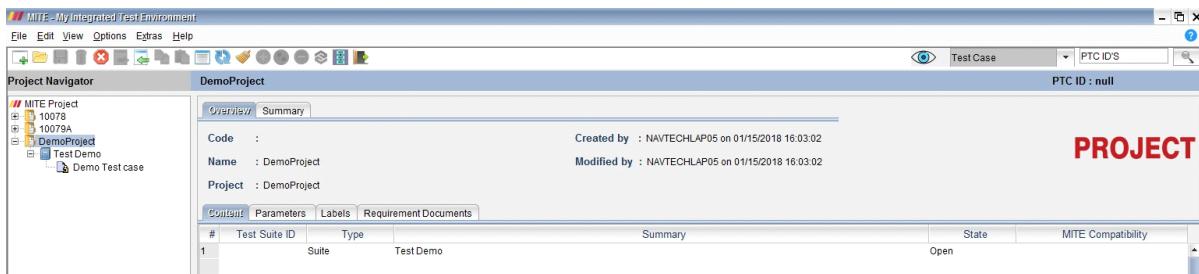


Figure 124: Demo Project

Above figure demonstrates the view upon addition of “Demo Project”, this is enabled with all the features available in MITE same as actual project(s).

This demo-project provides an opportunity for the first time user(s) of MITE to explore all the operations/features available in MITE application without any restriction except for “PTC flush”.

Note: - Demo Project cannot be Flushed or Submitted to PTC.

Created with the Personal Edition of HelpNDoc: [Easy EPub and documentation editor](#)

21.2 Spell Check:- A check for misspellings by using a ‘spell check’ option on ‘right-click’ for Description fields only at Test suite , at Test case levels and for Additional Remarks.

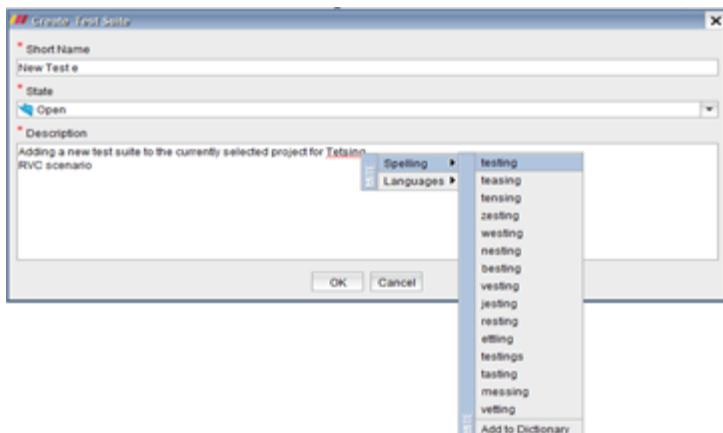


Figure 125: Spell Check – Description

Created with the Personal Edition of HelpNDoc: [Easy EPub and documentation editor](#)

21.3 Optimize Test Case:- Appears on Test Case right click options.

It eliminates the empty test steps, un-used parameters, un-used Labels as shown below:

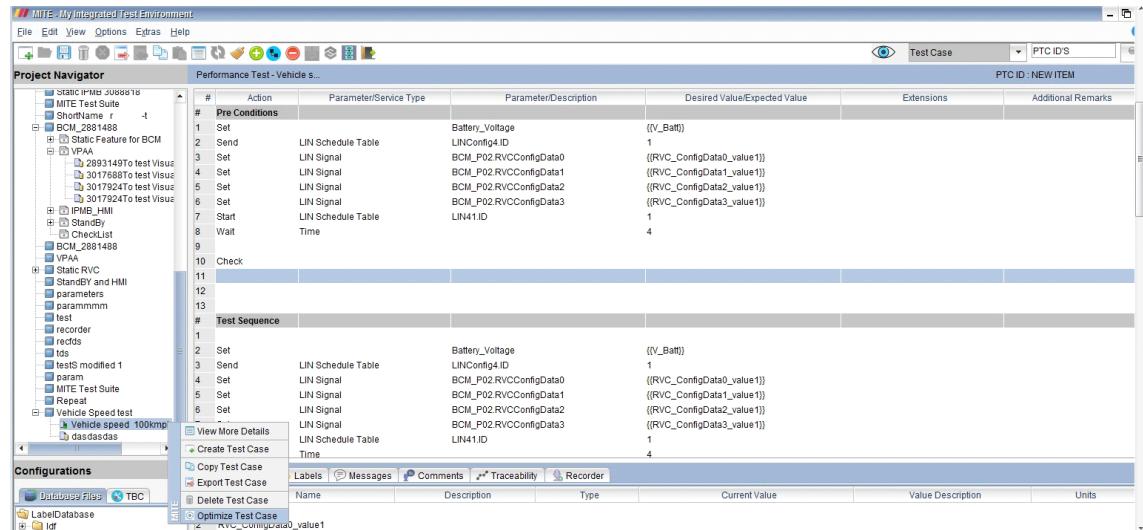


Figure 126: Optimize Test Case

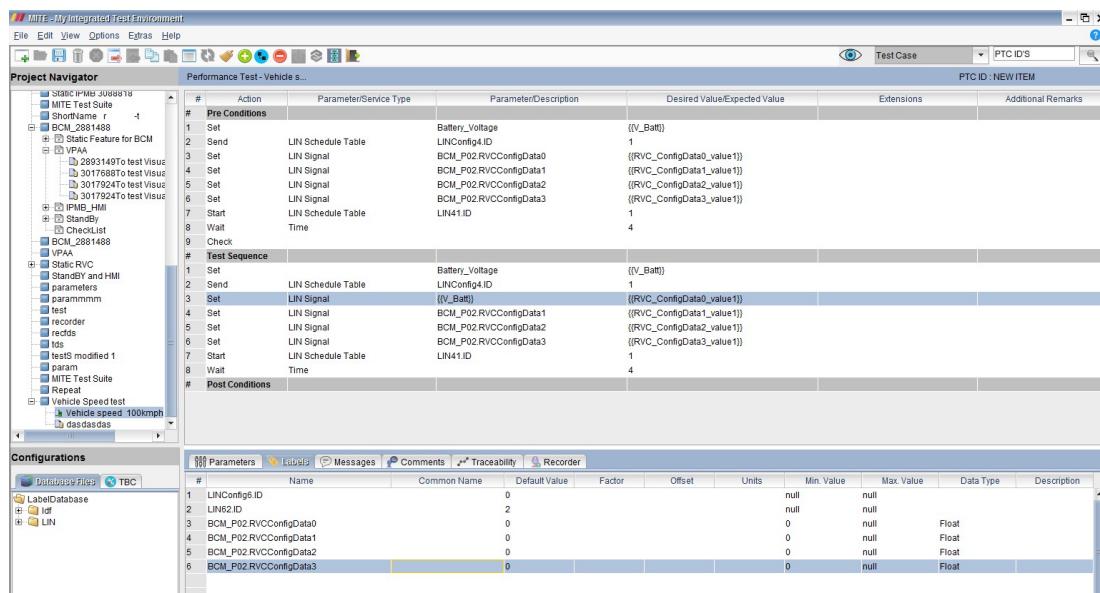


Figure 127: After Optimizing Test Cases

Created with the Personal Edition of HelpNDoc: [Easily create iPhone documentation](#)

21.4 Un-Do operation (Ctrl+Z) :- Un-do operation takes place on CTRL+Z

Scope of un-do: – Currently enabled on the test case editor and text areas only

Created with the Personal Edition of HelpNDoc: [Free Web Help generator](#)

21.5 Re-Do operation (Ctrl+Y) :- Re-do operation takes place on CTRL+Y

Scope of re-do: – Currently enabled on the test case editor and text areas only

Created with the Personal Edition of HelpNDoc: [Easy EBook and documentation generator](#)

21.6 MITE Statistics

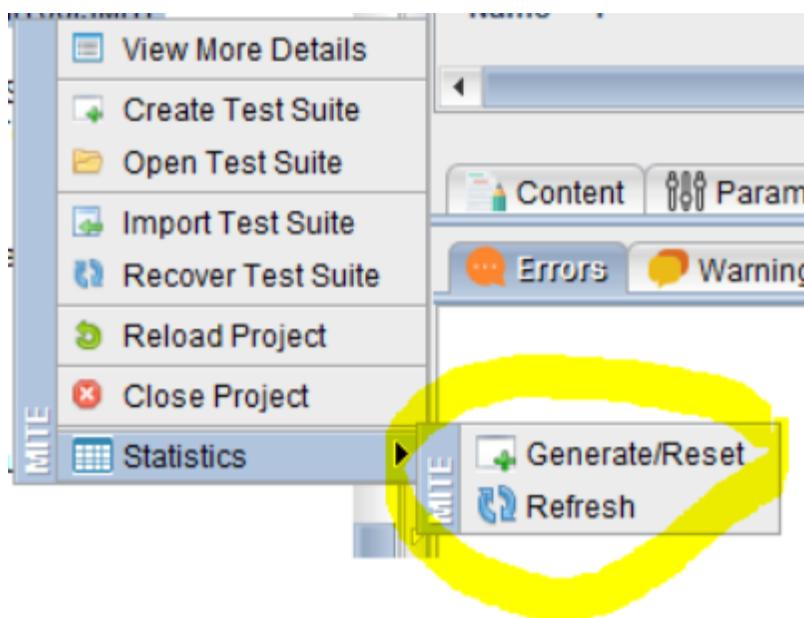
Introduction:

Basically, writing a Test Case in MITE starts from going through its particular requirement that is created in the PTC. One requirement will be analyzed and divided into multiple numbers of Test Cases. Currently, Traceability tab in Test Case level helps us to get some information about it.

But, it only shows the requirements that are linked to a particular Test Case. But, sometimes this information alone is not enough for deep analysis. With this new implementation, the important information starting from the requirements till the Test Case will be shown in the tabular format.

Functionality Details:

In MITE, statics are provided at project level and Test Suite level. At project level, user can get requirement statistics and Test suite wise statistics.



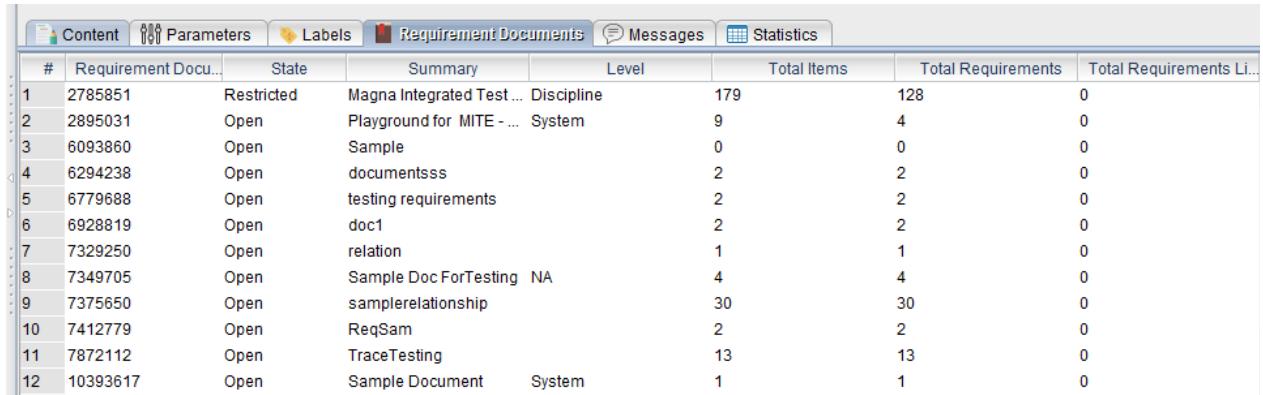
Requirement Statistics:

As a part of Requirement statistics, following columns are provided in the Requirement Documents tab.

Total Number of Items: Total Number of items that is present under that Requirement Document.

Total Number of Requirements: Total Number of items that is present under that Requirement Document other than headings and comments.

Total Number of Requirements covered with Valid Test Cases: Total Number of Requirements who's validated by is having Test Cases which are not headings, comments and unspecified items.



The screenshot shows the MITE software interface with the 'Requirements Documents' tab selected. The main area displays a table of requirements with the following columns: #, Requirement Docu..., State, Summary, Level, Total Items, Total Requirements, and Total Requirements Li... . The data includes:

#	Requirement Docu...	State	Summary	Level	Total Items	Total Requirements	Total Requirements Li...
1	2785851	Restricted	Magna Integrated Test ... Discipline		179	128	0
2	2895031	Open	Playground for MITE - ... System		9	4	0
3	6093860	Open	Sample		0	0	0
4	6294238	Open	documentss		2	2	0
5	6779688	Open	testing requirements		2	2	0
6	6928819	Open	doc1		2	2	0
7	7329250	Open	relation		1	1	0
8	7349705	Open	Sample Doc ForTesting NA		4	4	0
9	7375650	Open	samplerelationship		30	30	0
10	7412779	Open	ReqSam		2	2	0
11	7872112	Open	TraceTesting		13	13	0
12	10393617	Open	Sample Document System		1	1	0

Test Suite Statistics:

As a part of Test Suite statistics, following columns are provided in the Test Suite Statistics tab.

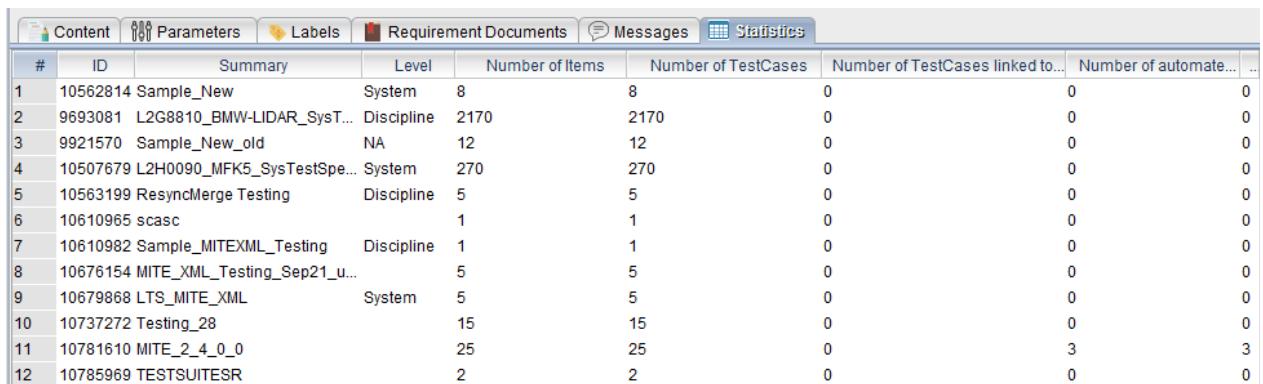
Total Number of Items: Total number of Test Cases under that Test Suite.

Total Number of Test Cases: Total number of Test Cases under that Test Suite whose category is not Heading, Comment and Unspecified.

Total Number of Test Cases linked to requirements : Total Number of Test Cases whose validates is having these requirements and the category of Test Case should not be heading or Comment or Unspecified.

Total Number of Automated Test Cases: The test case whose external script name is having the word "Automated" in it.

Total Number of MITE Automated Test Cases: The automated test case whose external script name is having the word "_MITE" in it.



The screenshot shows the MITE software interface with the 'Test Case Statistics' tab selected. The main area displays a table of test cases with the following columns: #, ID, Summary, Level, Number of Items, Number of TestCases, Number of TestCases linked to..., Number of automate..., and The data includes:

#	ID	Summary	Level	Number of Items	Number of TestCases	Number of TestCases linked to...	Number of automate...	...
1	10562814	Sample_New	System	8	8	0	0	0
2	9693081	L2G8810_BMW-LIDAR_SysT...	Discipline	2170	2170	0	0	0
3	9921570	Sample_New_old	NA	12	12	0	0	0
4	10507679	L2H0090_MFK5_SysTestSpe...	System	270	270	0	0	0
5	10563199	ResyncMerge Testing	Discipline	5	5	0	0	0
6	10610965	scasc		1	1	0	0	0
7	10610982	Sample_MITEXML_Testing	Discipline	1	1	0	0	0
8	10676154	MITE_XML_Testing_Sep21_u...		5	5	0	0	0
9	10679868	LTS_MITE_XML	System	5	5	0	0	0
10	10737272	Testing_28		15	15	0	0	0
11	10781610	MITE_2_4_0_0		25	25	0	3	3
12	10785969	TESTSUITESR		2	2	0	0	0

These Test Suite statistics can be seen in Test Suite level also in a separate tab called Test Case Statistics:

TestCases Information		
Number of Items	:	8
Number of TestCases	:	7
Number of TestCases Linked to Requirements	:	0
Number of Automated TestCases	:	5
Number of MITE_Automated TestCases	:	5

User have an option of customization also we these queries. User can change the queries from query details in the PTC and can click on “RESET” to get the statistics according to the changes that are done manually in the query. If user want the statistics as per MITE query definitions, then there is an option called “REFRESH” which can be used to get the original statistics as per MITE.

Query Details		
Query Name_Requirements	:	MITE_Requirements_Universal_ProjectsEngineering_DepartmentToolsMITE
Query Name_TestCases	:	MITE_TestCases_Universal_ProjectsEngineering_DepartmentToolsMITE

Level Set Up:

In the statistics tables, level is to be set to get the statistics regarding the linkage of requirements and Test Cases. If the Level is equal then only the relation related statistics are counted.

Export To Excel:

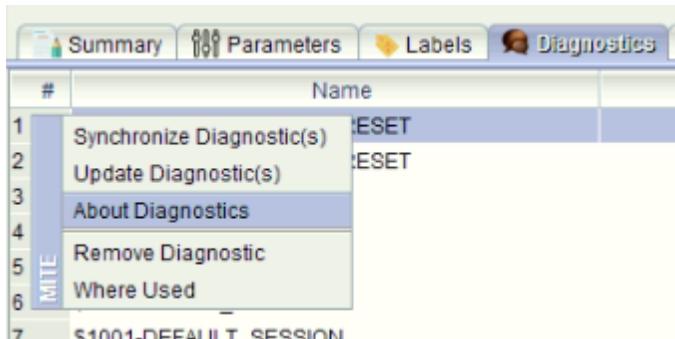
User can export the statistics data in to an excel file by clicking on Export Statistics which is available on header.

Created with the Personal Edition of HelpNDoc: [Free EBook and documentation generator](#)

21.7 Q & A Links

Whenever user faces any difficulty about a particular module of TestCase Authoring he/she can simply select help option or a right click option. Help option is available in Extensions dialog box when clicked on it, it displays the functionality of Extensions in default browser. To know about Parameters user can do a right

click on a column under Parameters tab and select About Parameters. Similarly to know about TestCase Editor or Labels he can select a right click option at TestCase Editor or Labels Tab.



7.11 Update and Synchronize options for Diagnostics and Labels

Update and Synchronize options for Diagnostics and Labels

(1)Update Labels:

- Whenever User wants to change/modify any labels then he/she can use update labels as option in Test Suite level. If User edits any labels at Test Case editor those labels can be automatically used in Test Suite levels.. For example in recorder,labels,diagnostics etc.
- It can be performed in various scenarios for example while copy and paste of test cases or while reloading any label database file. The following scenarios will help a User on how to update the labels while performing the action:-

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

21.8 Downloading User Manual & FAQ

In the previous versions of MITE (2.4.0.0 or below), User is able to view the manual by selecting Help Contents option under Help Menu item. From now User can download User Manual by selecting “Download UserManual” option under Help Menu item. User can also view the manual by selecting View Content or by clicking Short cut key “F1”.

In the earlier versions of MITE(2.6.0 or below) when user selects FAQ option in menu bar it used to display in PDF format but from now it is going to display in html view. And also User is able to download FAQ which will be in PDF format.



Created with the Personal Edition of HelpNDoc: [Full-featured Documentation generator](#)

21.9 Find & Replace, Rename Functionallities

RENAME

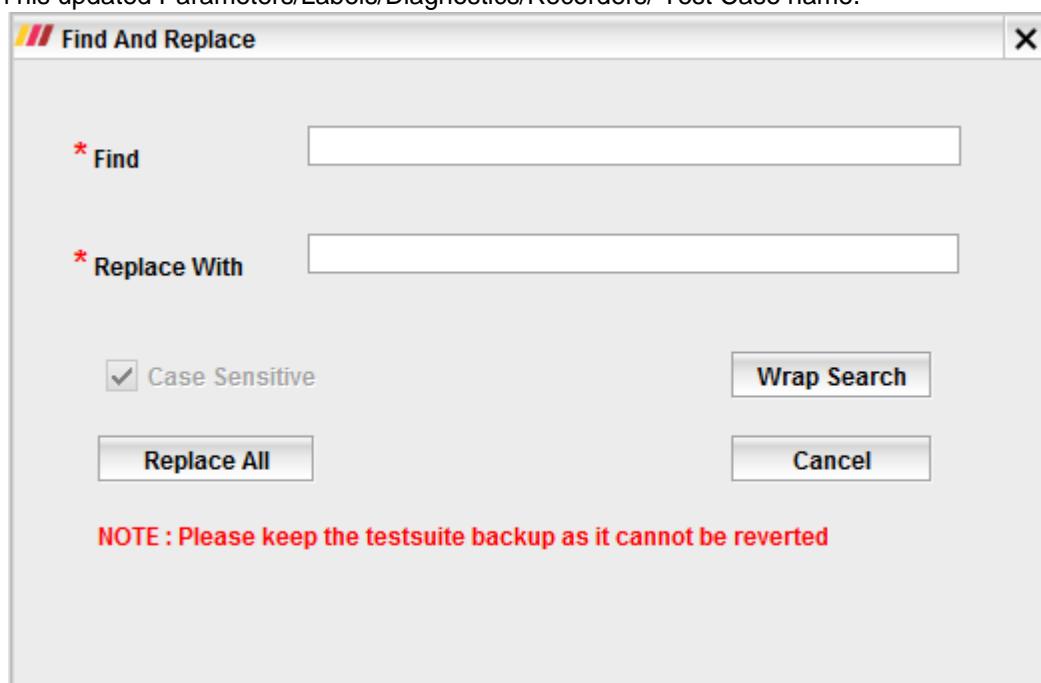
Rename feature allows users to rename the TestCase or TestSuite name in TestCase role .This Renaming is only applicable for TestCase and TestSuite only and not for Project. This is also not applicable for False TestCase or False TestSuite.

There are 3 ways for renaming.

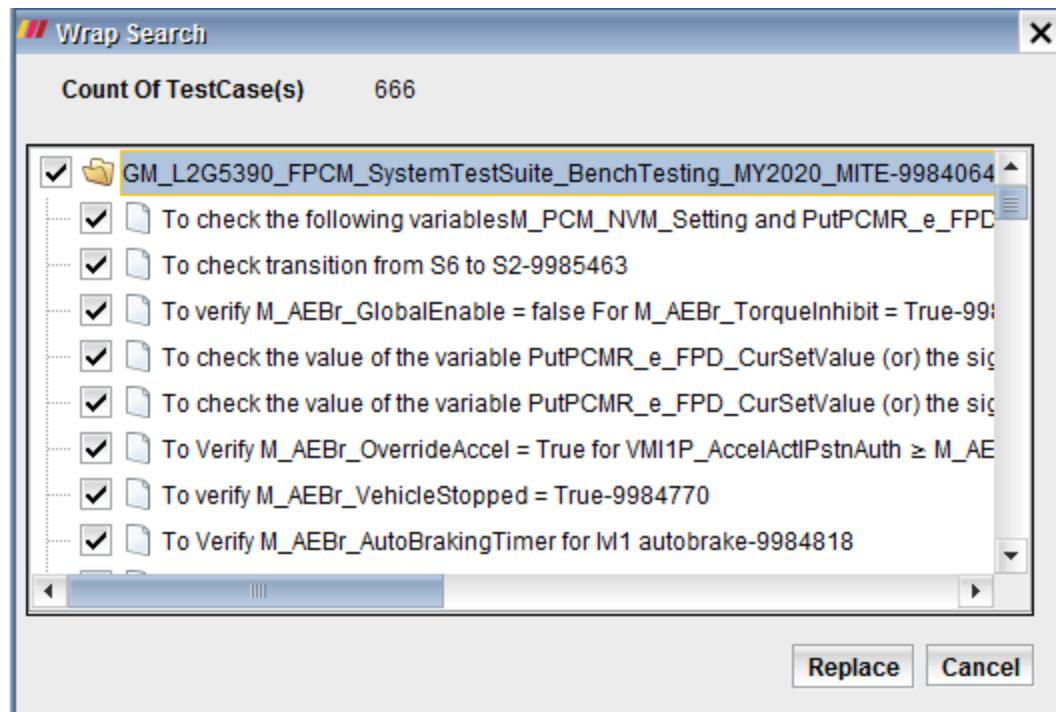
1. Using Rename TestSuite/TestCase, a right click option.
2. By clicking F2, a shortcut key.
3. By Selecting Rename option under View Menu item in Menu bar.

Find And Replace

User can replace the text using ctrl+f short cut key which will be applicable only for Test Suite and Test Cases at Test Case role .Using this user can replace used Parameters/Labels/Diagnostics/Recorders/Test Case name with his updated Parameters/Labels/Diagnostics/Recorders/ Test Case name.



In the field Find, user needs to enter the text he/she is searching for.
 In the field Replace With, user needs to enter the updated text.
 By default this operation is Case Sensitive.
 For Parameter Search, the value should be in curly braces like {{sample}}.
 Replace All option replaces all the matched text with new text.
 Cancel option simply closes the window.
 Wrap Search option will be enabled only when selection is on Test Suite, this will display all the Test Cases containing the Find text.
 User needs to select the Test Cases where he/she requires replacement.
 User needs to select at least one Test Case to do replace.
 All check box is for multi selection purpose, user can select multiple cases using ctrl and select All checkbox then selected cases will be checked.
 All cases can be selected at once by clicking at all checkbox which is beside of testsuite



Created with the Personal Edition of HelpNDoc: [Easily create Help documents](#)

21.10 Email Attachment & Videos

With MITE-v3.0.0.0 User can directly send the email attachment through Help section. In help section we have support where user will be sending the respective query along with the attachment directly.

Also for better understanding User can checkout the videos in the particular section incase of any additional information.

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

22 Exercise – Test Case Authoring in MITE

Created with the Personal Edition of HelpNDoc: [Free Kindle producer](#)

22.1 CAN/LIN tests case(s)

1. LIN Examples

#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	Additional Remarks
#	Pre Conditions					
1	Set	LIN Signal	BCM_P00.PrkAidSnsRrCnrr_D_Stat_UB	1 - Refreshed		
2	Set	LIN Message	BCM_P91.ID	33		
3	Set	LIN Bus	LIN4.ID	2		
4	Set	LIN Schedule Table	LINConfig4.ID	0		

2. Set Lin signal - PrkAidSnsRrCnrr_D_Stat_UB To Not Refreshed

#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	Additional Remarks
#	Pre Conditions					
1	Set	LIN Signal	BCM_P00.PrkAidSnsRrCnrr_D_Stat_UB	0 - Not Refreshed		

3. Set ECU in Reverse Gear in Active_confirmed state and its respective UB bit active and confirm using Transmission Status LIN signal. Activate the parking sensor PrkAidSnsRrCnrr_D_Stat to 3- zone 3 along with its UB bit and verify RVC Zoom level is 3

System Test - To Set a ...							PTC ID : NEW ITEM
#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	Additional Remarks	
#	Pre Conditions						
1	Set	LIN Signal	BCM_P00.GearRvrse_D_AdL	3 - Active_confirmed			
2	Set	LIN Signal	BCM_P00.GearRvrse_D_AdL_UB	1 - Refreshed			
3	Set	LIN Signal	BCM_P90.PrkAidSnsRrCnrr_D_Stat	3 - zone 3			
4	Set	LIN Signal	BCM_P00.PrkAidSnsRrCnrr_D_Stat_UB	1 - Refreshed			
5	Check	LIN Signal	BCM_P93.TransmissionStatus	1 - Enable			
6			RVC_P20.RVCZoomLevel_Stat	3 - Level 3 Zoom			

4. Example : Parameters

Pre-Condition -

- a. Set Battery supply voltage to {{V_Batt}}V
- Test Sequence -

- b. Set Veh_V_ActlEng = {{Veh_V_ActlEng_value3}} kmph
- c. Wait for {{DelayTime}} sec while observing Video and LIN : Video Output :

{{\Vide
oOutput_value1}}

}} Post-

Condition –

- d. Stop recording Video and LIN.

#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	Additional Remarks
#	Pre Conditions					
1	Set	ECU Input	Battery supply voltage to	{{V_Batt}}		
#	Test Sequence					
1	Set	LIN Signal	BCM_P01.Veh_V_AdLEng	{{Veh_V_AdLEng_value3}}		
2	Wait	Time		{{DelayTime}}		
3	While	Test Step (s)			Test Sequence Step 2	
3.1	Check	Video	Video Output	{{VideoOutput_value1}}		
#	Post Conditions					
1	Stop	LIN Bus		0		

5. Example : Loops Repeat

Test Sequence -

- Set {{CamraOvrlStat_D_Rq_param}} = ON
 - After every test combination Wait for {{DelayTime}}sec while observing the Video and LIN S.No {{DrStatTgate_B_Actl_param}}
- {{TrlrLampCnct_B_Actl_param}}

- 1 Closed Disconnected
- 2 Closed Connected
- 3 Ajar Disconnected
- 4 Ajar Connected

#	Test Sequence					
1	Start Recorder	Video	{}(Debut)	{}(Debut_ON)		
2	Start	Record	Rec_LIN			Camera is ON Start recording LIN.
3	Check	Environment	Video signal quality should not change for the duration of th...	NO CHNAGE		
4	Set	LIN Signal	BCM_P01.DistanceBarSetting	1 - On (Command from HMI to enable Visual Park Aid Alert)		
5	Set	LIN Signal	BCM_P00.GearRvrse_D_Actl	1 - Refreshed		
6	Set	LIN Signal	BCM_P00.GearRvrse_D_Actl	1 - Inactive_confirmed		
7	Repeat	Test Step				
7.1	Set	LIN Signal	BCM_P01.CamraOvrlStat_D_Rq	0,1,1,1,0,1,1,1,1		
7.2	Set	LIN Signal	BCM_P01.TrlrLampCnct_B_Actl	0 - No (No Trailer Connected)		
7.3	Set	LIN Signal	BCM_P01.DistanceBarSetting	1,0,0,0,1,0,0,0,0		
7.4	Set	LIN Signal	BCM_P00.SAPPStatusCoding	0,0,1,0,0,1,1,0,0		
7.5	Set	LIN Signal	BCM_P01.DrStatTgate_B_Actl	0,1,0,1,0,1,0,1,0,1		
7.6	Set	LIN Signal	BCM_P01.DrStatInnTgate_B_Actl	0,1,0,1,0,1,0,1,0,1		
7.7	Run	Function	Sapp_Status			Function "Sapp_Status" call
8	Wait	Time	{}(DelayTime)	3		

Created with the Personal Edition of HelpNDoc: [Easily create Qt Help files](#)

22.2 Diagnostic Test Case(s)

- Read DID – F1A6 in default session and expect positive response

#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	Additional Remarks
# Pre Conditions						
1	Send	DiagRequest	\$1001-Default Session Start			
2	Check	DiagResponse	\$5001-Default Session Start P2Can	00 01		
# Test Sequence						
1	Send	DiagRequest	\$22f1a6-Message Matrix Read			
2	Check	DiagResponse	\$62f1a6-Message Matrix Read Vehicle Line	51 - KL		
# Post Conditions						
1	Send	DiagRequest	\$1101-HardReset Reset			
2	Check	DiagResponse	\$5101-HardReset Reset			

Parameters Labels Messages Comments Traceability Recorder Diagnostics

#	Name	Service Name	Service Id	DID Id	SubFunction Id	Signal Name	Bit Length	Byte Position
1	\$1001-Default Session Start	Start	(\$10)	NA	(\$01)	NA	8	1
2	\$5001-Default Session Start P2Can	Start	(\$50)	NA	(\$01)	P2Can	16	2
3	\$22f1a6-Message Matrix Read	Read	(\$22)	(\$f1a6)	NA	NA	16	1
4	\$62f1a6-Message Matrix Read Vehicle Line	Read	(\$62)	(\$f1a6)	NA	Vehicle Line	8	5
5	\$1101-HardReset Reset	Reset	(\$11)	NA	(\$01)	NA	8	1
6	\$5101-HardReset Reset	Reset	(\$51)	NA	(\$01)	NA	8	1

Note – Above test case authored using labels from odx/pdx files [no RAW values are used]

- Read DID – F186 in programming session and expect NRC-31

#	Action	Parameter/Service Type	Parameter/Description	Desired Value/Expected Value	Extensions	Additional Remarks
1	Run	Test Case	Standard preconditions			
2	Send	DiagRequest:Physical	\$1001-DEFAULT_SESSION			
3	Check	DiagResponse:Physical	\$5001-DEFAULT_SESSION	0x50 01		Default session active
#	Test Sequence					
1	Send	DiagRequest:Functional	Read DID - F186	0x22 F1 86		
2	Check	DiagResponse:Functional	Positive Response	0x62 F1 86 01		
3	Send	DiagRequest:Physical	\$1002-PROGRAMMING_SESSION			
4	Check	DiagResponse:Physical	\$5002-PROGRAMMING_SESSION	0x50 02		programming request is present
5	Send	DiagRequest:Physical	Read DID - F186 in Programming Session	0x22 F1 86		
6	Check	DiagResponse:Physical	Negative Response NRC31	0x7F 22 31		NRC S31 - Request Out of Range
#	Post Conditions					
1	Send	DiagRequest:Physical	Hard Rest	0x11 01		
2	Check	DiagResponse:Physical	PositiveResponse	0x51		

Diagnostics								
#	Name	Service Name	Service Id	DID Id	SubFunction Id	Signal Name	Bit Length	Byte Position
1	\$1001-DEFAULT_SESSION		(\$10)	NA	(\$01)	NA	8	1
2	\$5001-DEFAULT_SESSION		(\$50)	NA	(\$01)	NA	8	1
3	\$5002-PROGRAMMING_SESSION		(\$50)	NA	(\$02)	NA	8	1

Table 129: Combinations allowed in authoring Test Case in MITE

Note – Above test case authored using raw values [no pdx/odx files are used]

Created with the Personal Edition of HelpNDoc: [Free HTML Help documentation generator](#)

22.3 Authoring rules for value entry

The following are the allowed numeric value entry possible in MITE test case editor

Desired Value/Expected Value	Description	Examples
Logical values	Drop down list appears from label database files	0 – Disable 1 – Refreshed
Raw/physical Values	To assign value to a specific byte or bit using format " :ByteXX.BitXX"	:Byte05.Bit03 :Byte45.Bit00
Decimal Values	Decimal Values including negative values	0 ,1,256,0052,89,-127
Hex Values	Hex decimal values ranging from 0-9 , A-F , a-f	0xA1 , 0x12FF
Binary Values	Binary values using "b()" range 0-1 Only allowed when :byteXX.bitXX format is used in Parameter/Description column	b(00001110)
Float Values	Float values	0.1 ,10.25,2560.0
Don't care	Represent using 'X'	0xX1 , 0x1XFX b(000011X0)

Table 130: Number system format allowed in MITE

Created with the Personal Edition of HelpNDoc: [Free EPub producer](#)

23 MULTIPLE NODE SELECTION IN TREE FOR COPY/PASTE

- Only test cases are allowed to copy/paste in multiple node selection.
- Test Cases selection should happen from **only one single suite** and paste them in **same test suite/test case (or) different test suite/test case**.
- “**CTRL**” button must use to select multiple nodes from tree.

Created with the Personal Edition of HelpNDoc: [Free CHM Help documentation generator](#)

23.1 Multiple test case nodes selection with right click pop-up

If selected test case nodes count more than one then except copy event remaining all events will be disable.

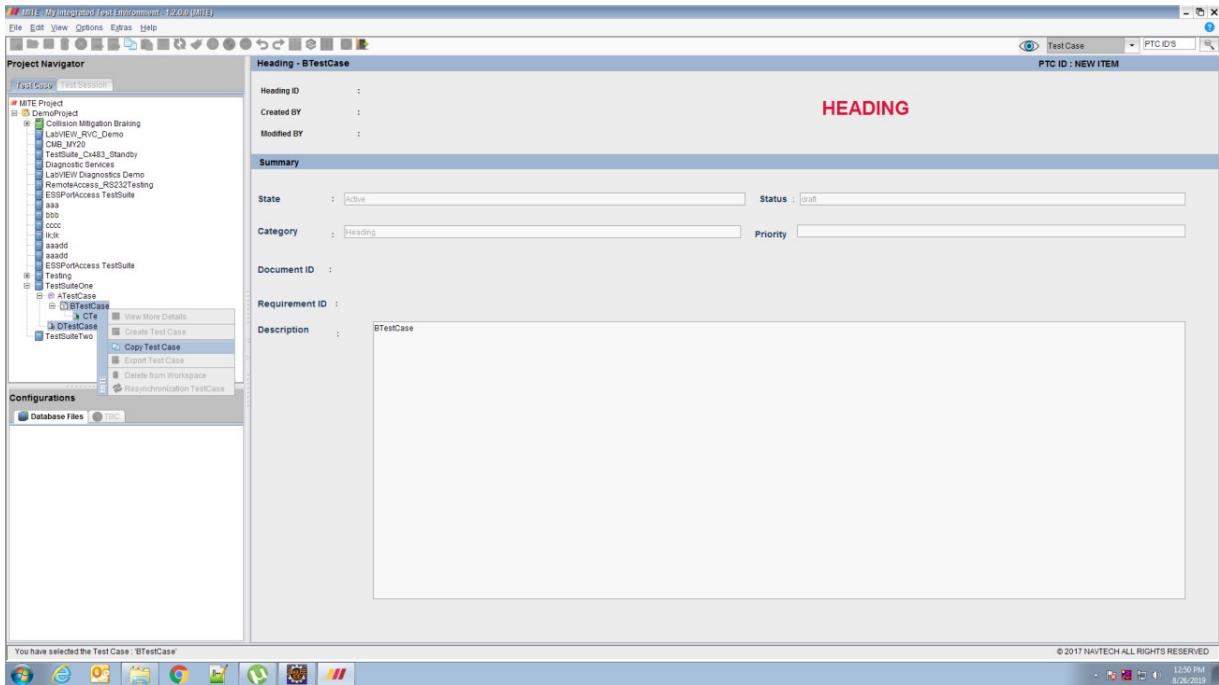


Figure 131: Multiple test case node selection with tree right clicks pop-up.

Created with the Personal Edition of HelpNDoc: [Benefits of a Help Authoring Tool](#)

23.2 Label database hiding

If node selection count is more than one then label database will be hidden.

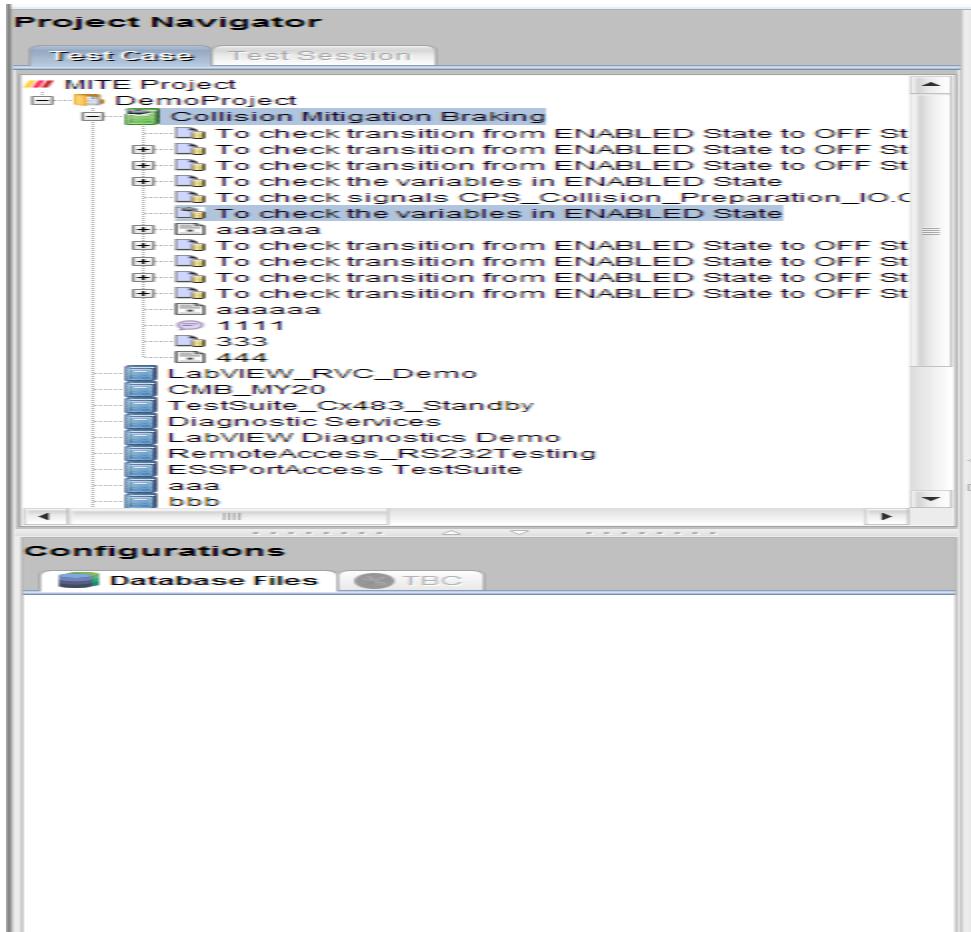


Figure 132: Label database hiding if node selection counts more than one.

Created with the Personal Edition of HelpNDoc: Produce electronic books easily

23.3 Right side page update while clicking node in multiple node selection

Right side panel will never update in multiple node selection from the first selected page.

For example: If we select Test suite without using "ctrl" button then right side panel will be update and if we select any other node using ctrl button then right side panel will never update.

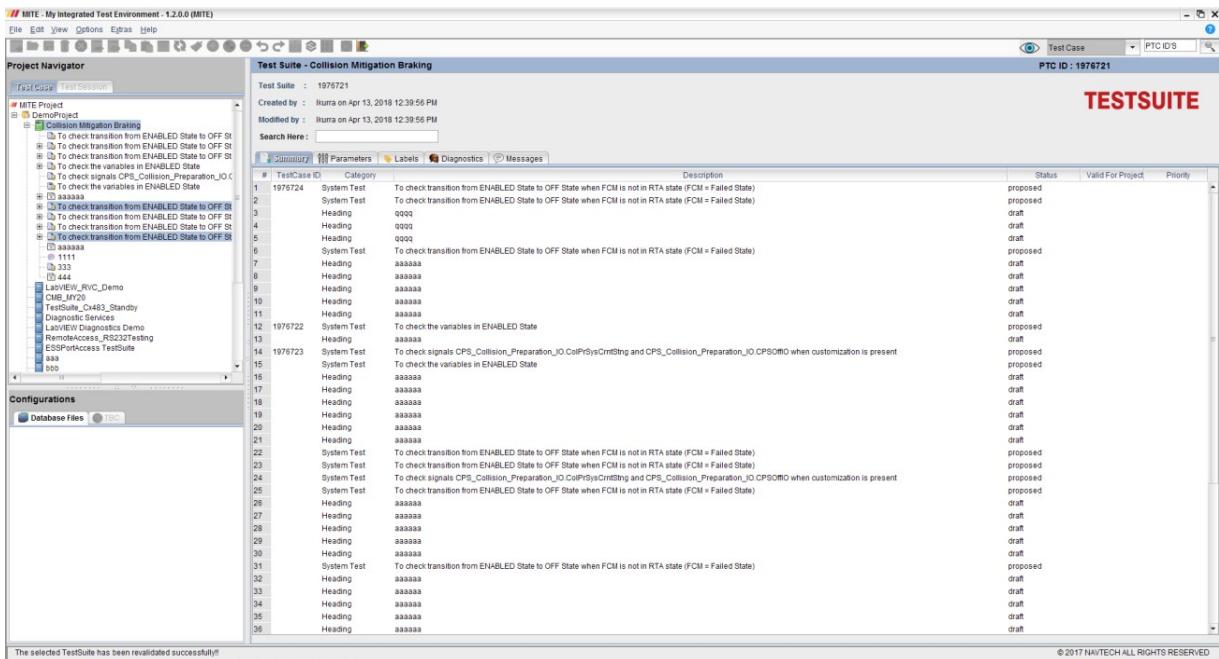


Figure 133: Right side panel updating status based on multiple node selection

Icons will be disabling if more than one node selected.

If more than one test case selected from a suite then except “copy & exit” icon remaining icons will be disable. If more than one test case including test suite selected then except “exit” icon remaining icons will be disable

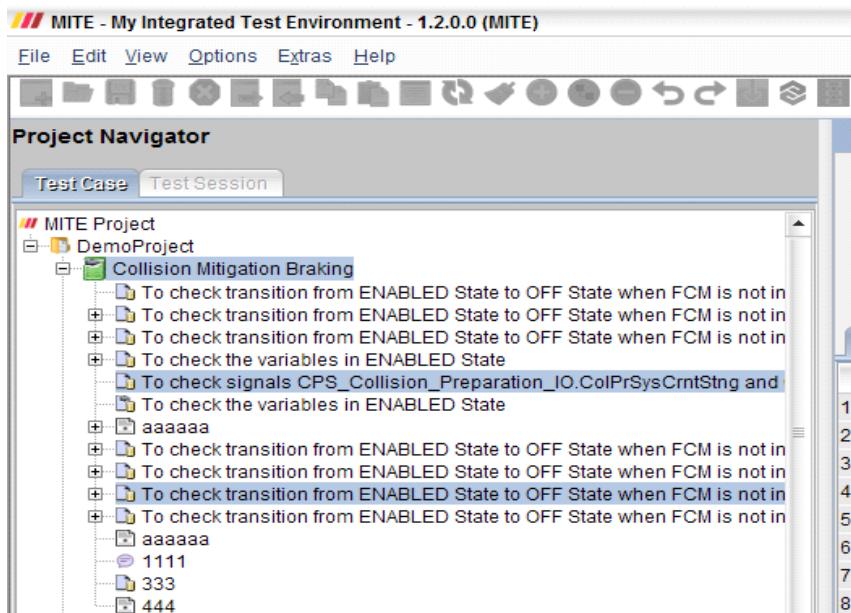


Figure 134: Icons disabled due to the multiple node selection

Events are not allowed due to violation of the multiple test case selection copy/paste.
Multiple node selection copy/paste rules already defined above in the documentation. If the rules are violated then validation will happen with proper message as like below

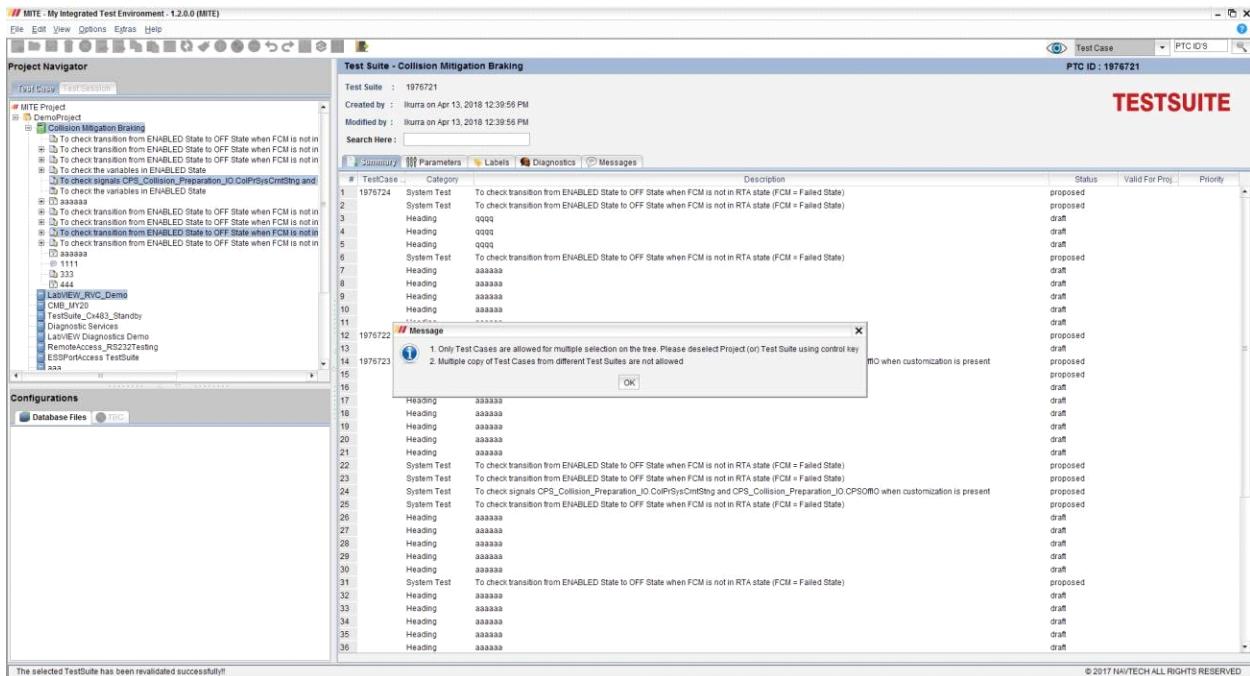
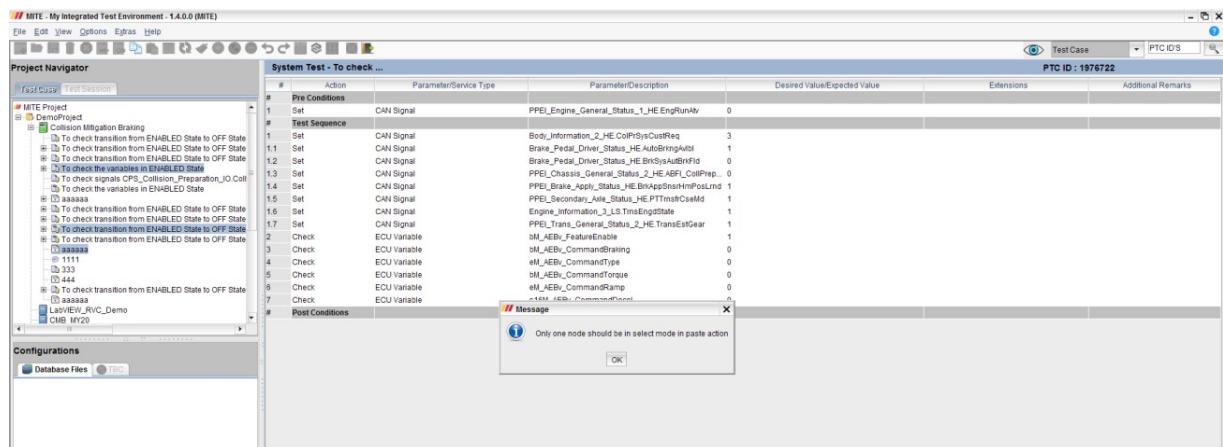


Figure 135: Validation message for violate the rules of multiple node selection

Created with the Personal Edition of HelpNDoc: Full-featured EPub generator

23.4 Validation for paste event after copy one or more test cases

If selected nodes are more than one and try to paste copied nodes then validation will happen. Because at a time we can paste copied test cases in single. So in that case we should not select more than one node from tree.



Created with the Personal Edition of HelpNDoc: Produce online help for Qt applications

23.5 Select and deselect multiple nodes from tree

If we select multiple nodes from tree and deselect each one by using **ctrl button**, in that case the right side panel will update while deselect a node and updated with next selection node in the order of selection.

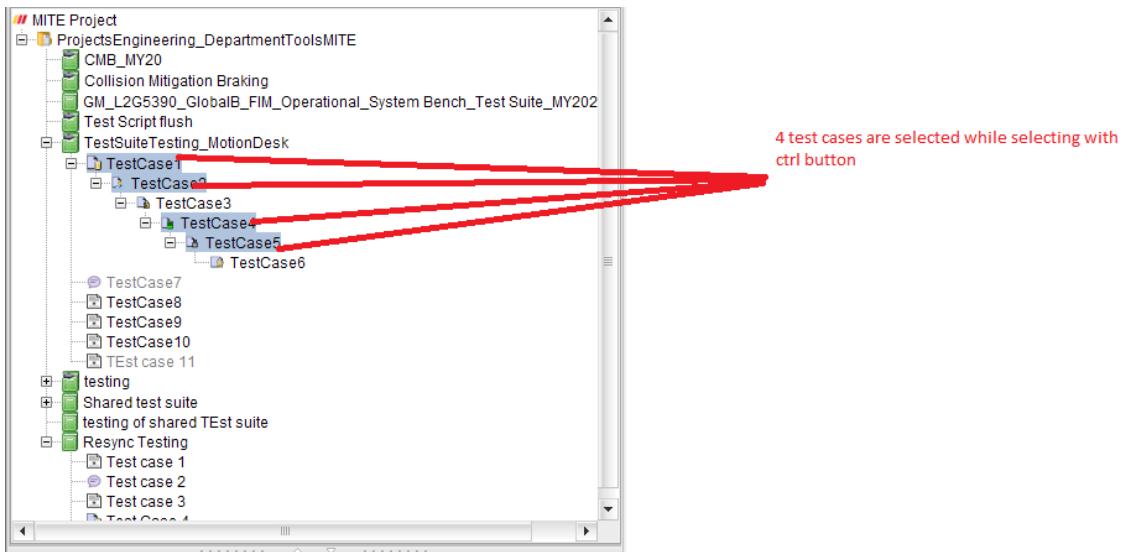


FIG : Multiple test cases are selected using ctrl button from key board.

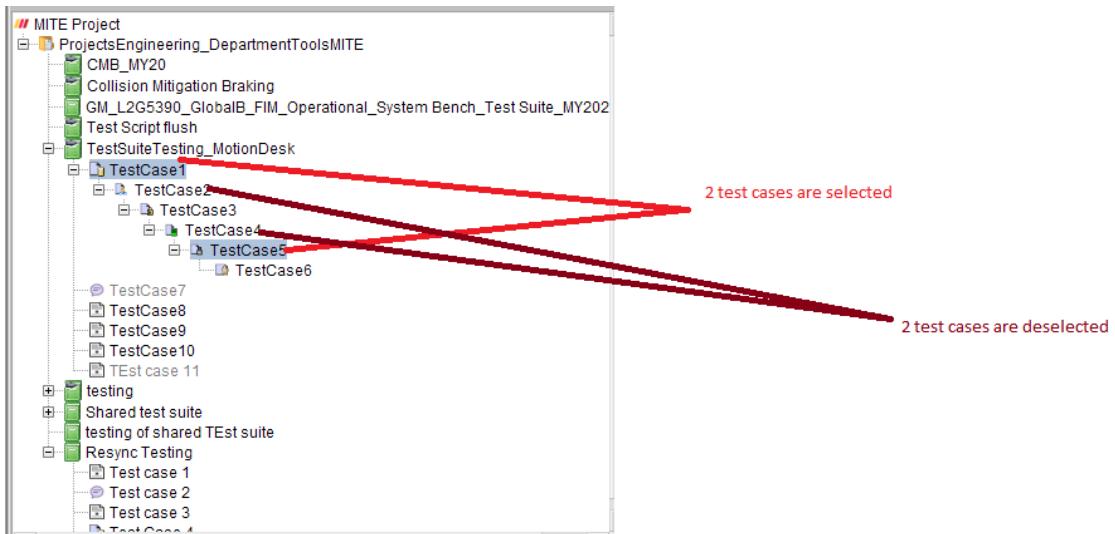


FIG : Two test cases are deselected using ctrl button from key board.

Created with the Personal Edition of HelpNDoc: [Free help authoring environment](#)

23.5.1 Test Case level “Project summary” table

Project summary table contains a text filed on the top of the table. To find information from table we can use this text field to search that. While we type required information in text filed, table automatically sort and show that information to us.



Search Here :

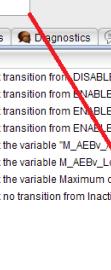
#	ID	Type	Summary	Status	MITE Compatibility
1	7662608	Suite	testsuitesuite flush	Open	True
2	7662583	Suite	testsuite flush	Open	True
3	7662560	Suite	testcase	Open	True
4	7660994	Suite	testcaseflushtesting	Open	True
5	7660944	Suite	TestSuiteTesting_MotionDesk	Open	True

Search field to search and sort data from project summary table.

FIG : Search field to search and sort data from project summary tableCreated with the Personal Edition of HelpNDoc: [Full-featured EBook editor](#)

23.5.2 Test Case level “Test suite summary” table

Test suite summary table contains a text filed on the top of the table. To find information from table we can use this text field to search that. While we type required information in text filed, table automatically sort and show that information to us.



Search Here :

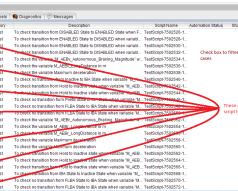
#	TestCase	Category	Description	Status	Valid For Proj...	Priority
1	7502526	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Enabled and variable "b_ME_FDI_DiagnosticsFailed = False" (Updated according to the requirement)	proposed		
2	7502528	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_ValidInputs = False" due to setting of DTC B356A	proposed		
3	7502530	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_ValidInputs = False" due to variable "ME_FDI_DiagnosticsFailed = True"	proposed		
4	7502532	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_TorqueInhibit = True"	proposed		
5	7502534	System Test	To check the variable "M_AEBv_Autonomous_Braking_Magnitude" when system is in ENABLED State	proposed		
6	7502536	System Test	To check the variable M_AEBv_LongDistance in m	proposed		
7	7502538	System Test	To check the variable Maximum deceleration	proposed		
8	7502540	System Test	To check no transition from Inactive to BRA State when variable "M_AEBv_BrakeAssistBrakeRate = False"	reviewed		

Search field to search and sort the data from test suite summary table

FIG : Search filed to search and sort the data from test suite summary tableCreated with the Personal Edition of HelpNDoc: [Easily create Web Help sites](#)

23.5.3 Test Script level “Test Suite Summary” table

Test Script level **Test suite summary table** contains a text filed on the top of the table. To find information from table we can use this text field to search that. While we type required information in text filed, table automatically sort and show that information to us. In this summary table there is a check box beside search field. The main purpose of that check box is filtering checked test cases in test script summary table.



Search Here :

Check Test Suite ID :

Check box to filter checked test cases

Search field to search and sort the data from test script level summary table

The user has the last case as selected in test suite summary table in test script level

#	TestID	Category	Description	Actual Status	Brake Test	Search TestID
1	7620001	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Enabled and variable "b_ME_FDI_DiagnosticsFailed = False" (Updated according to the requirement)	proposed		
2	7620002	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_ValidInputs = False" due to setting of DTC B356A	proposed		
3	7620003	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_ValidInputs = False" due to variable "ME_FDI_DiagnosticsFailed = True"	proposed		
4	7620004	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_TorqueInhibit = True"	proposed		
5	7620005	System Test	To check the variable "M_AEBv_Autonomous_Braking_Magnitude" when system is in ENABLED State	proposed		
6	7620006	System Test	To check the variable M_AEBv_LongDistance in m	proposed		
7	7620007	System Test	To check the variable Maximum deceleration	proposed		
8	7620008	System Test	To check no transition from Inactive to BRA State when variable "M_AEBv_BrakeAssistBrakeRate = False"	reviewed		
9	7620009	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Enabled and variable "b_ME_FDI_DiagnosticsFailed = False" (Updated according to the requirement)	proposed		
10	7620010	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_ValidInputs = False" due to setting of DTC B356A	proposed		
11	7620011	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_ValidInputs = False" due to variable "ME_FDI_DiagnosticsFailed = True"	proposed		
12	7620012	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_TorqueInhibit = True"	proposed		
13	7620013	System Test	To check the variable "M_AEBv_Autonomous_Braking_Magnitude" when system is in ENABLED State	proposed		
14	7620014	System Test	To check the variable M_AEBv_LongDistance in m	proposed		
15	7620015	System Test	To check the variable Maximum deceleration	proposed		
16	7620016	System Test	To check no transition from Inactive to BRA State when variable "M_AEBv_BrakeAssistBrakeRate = False"	reviewed		
17	7620017	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Enabled and variable "b_ME_FDI_DiagnosticsFailed = False" (Updated according to the requirement)	proposed		
18	7620018	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_ValidInputs = False" due to setting of DTC B356A	proposed		
19	7620019	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_ValidInputs = False" due to variable "ME_FDI_DiagnosticsFailed = True"	proposed		
20	7620020	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_TorqueInhibit = True"	proposed		
21	7620021	System Test	To check the variable "M_AEBv_Autonomous_Braking_Magnitude" when system is in ENABLED State	proposed		
22	7620022	System Test	To check the variable M_AEBv_LongDistance in m	proposed		
23	7620023	System Test	To check the variable Maximum deceleration	proposed		
24	7620024	System Test	To check no transition from Inactive to BRA State when variable "M_AEBv_BrakeAssistBrakeRate = False"	reviewed		
25	7620025	System Test	To check transition from DISABLED State to ENABLED State when FDI feature is Enabled and variable "b_ME_FDI_DiagnosticsFailed = False" (Updated according to the requirement)	proposed		
26	7620026	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_ValidInputs = False" due to setting of DTC B356A	proposed		
27	7620027	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_ValidInputs = False" due to variable "ME_FDI_DiagnosticsFailed = True"	proposed		
28	7620028	System Test	To check transition from ENABLED State to DISABLED when variable "M_AEBv_TorqueInhibit = True"	proposed		
29	7620029	System Test	To check the variable "M_AEBv_Autonomous_Braking_Magnitude" when system is in ENABLED State	proposed		
30	7620030	System Test	To check the variable M_AEBv_LongDistance in m	proposed		
31	7620031	System Test	To check the variable Maximum deceleration	proposed		
32	7620032	System Test	To check no transition from Inactive to BRA State when variable "M_AEBv_BrakeAssistBrakeRate = False"	reviewed		

FIG : Test script test suite summary table with random checked cases



FIG : Checked test cases are filtered while check filtered check box

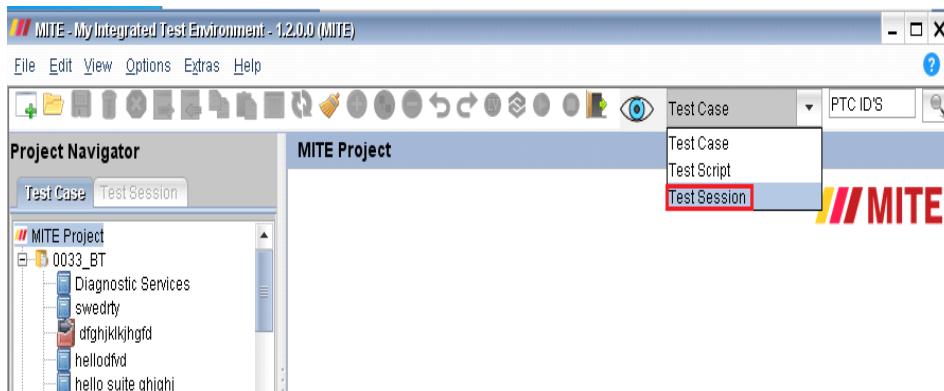
Created with the Personal Edition of HelpNDoc: [Free help authoring tool](#)

24 Test Session

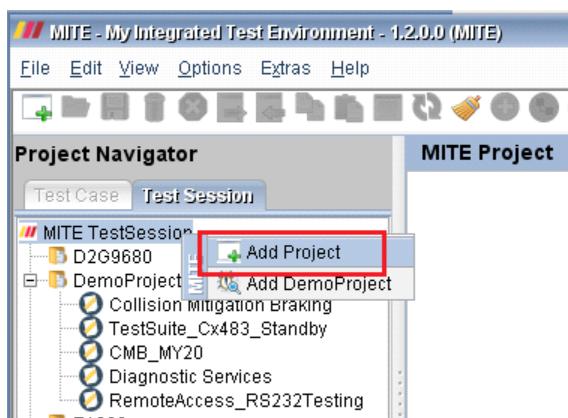
Created with the Personal Edition of HelpNDoc: [Full-featured EBook editor](#)

24.1 Add / Map a Project

Change to **TEST SESSION** role

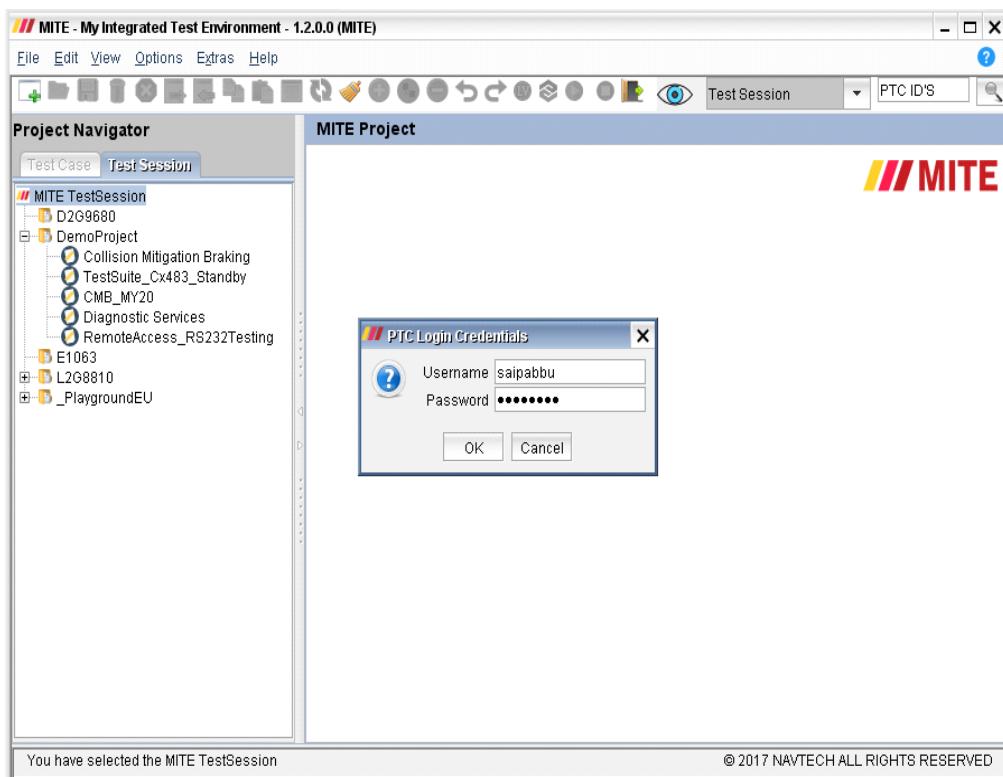


Right click on **MITE Test Session** and select **Add Project**

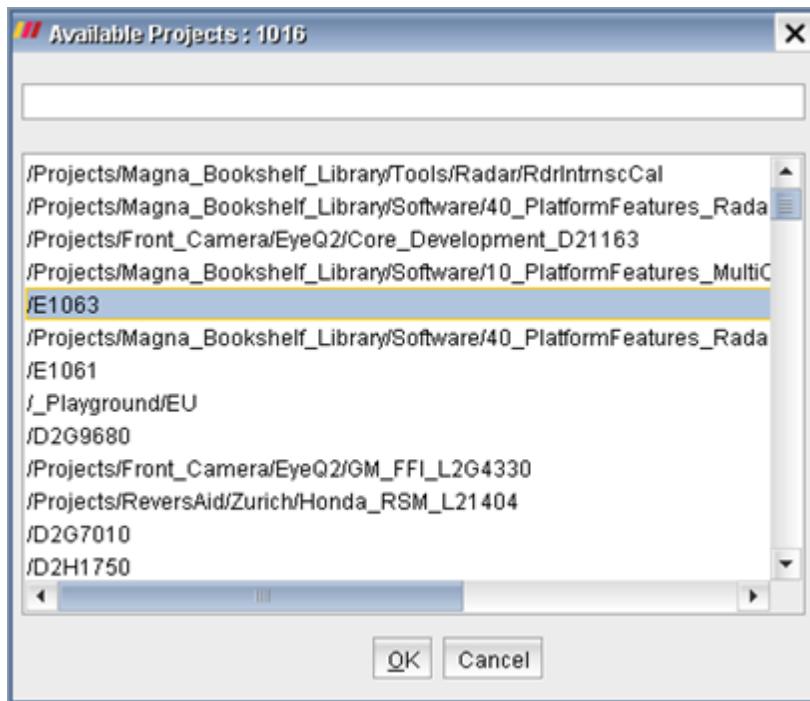


Before entering PTC credentials, make sure that your PTC is connected and it's connection is strong. Weak connection may interrupt your session.

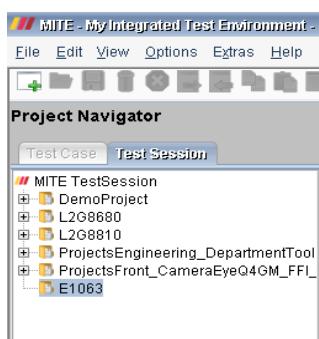
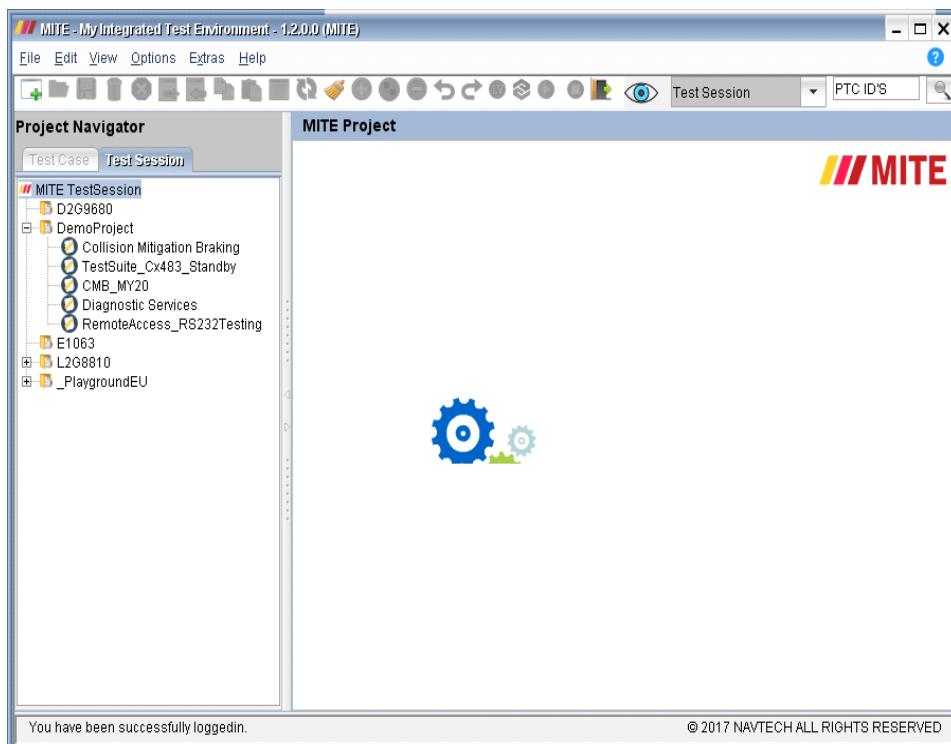
Enter valid **PTC credentials** and click on **OK**.



A dialog box will appear with list of projects in it. Choose a project which you want to add and click on ok.



After completion project will be added.



Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

24.2 Add Test session in project

To add test session in Demo project User should go with right click option on “Project Node” as like below.

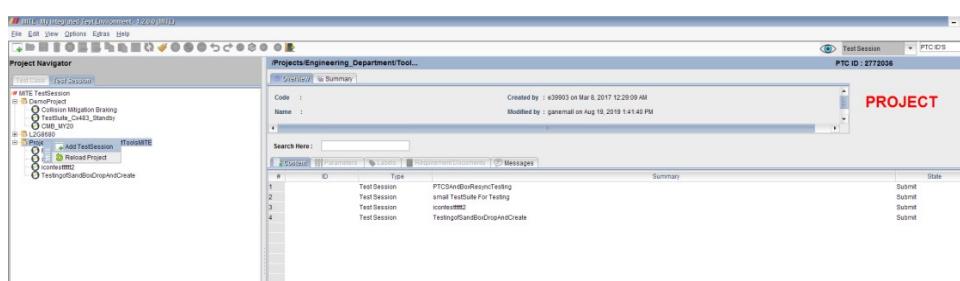


Figure 141: Right click option on project node to add test session.

Created with the Personal Edition of HelpNDoc: [What is a Help Authoring tool?](#)

24.2.1 Add Test Session Frame

If user click on “Add Test session” option from tree right click, a dialogue box will appear on screen i.e **Add Test Session** dialogue box.

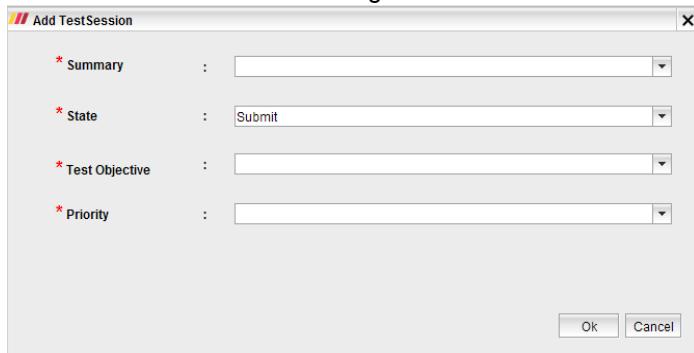


Figure 142: Add Test session dialogue box

In Add Test Session dialogue box, it consists of summary, state, Test Objective, Priority feeds. To submit details, user should select one option from existed list in drop-down from each field. Here summary & “test objective” fields are filtered drop-downs.

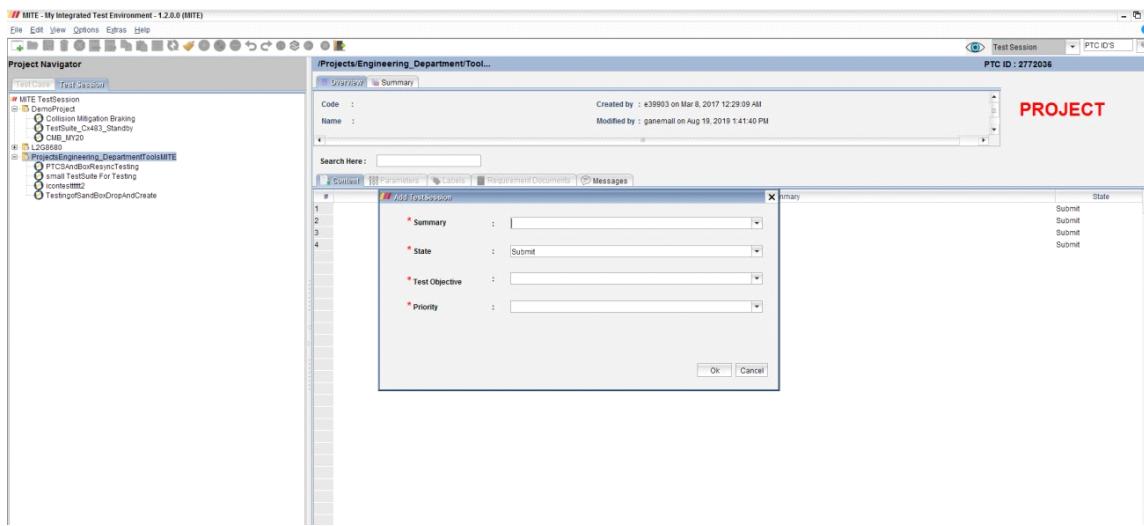


Figure 143: Add Test session dialogue window with summary details To add test session in project we need to fill complete details as like below image.

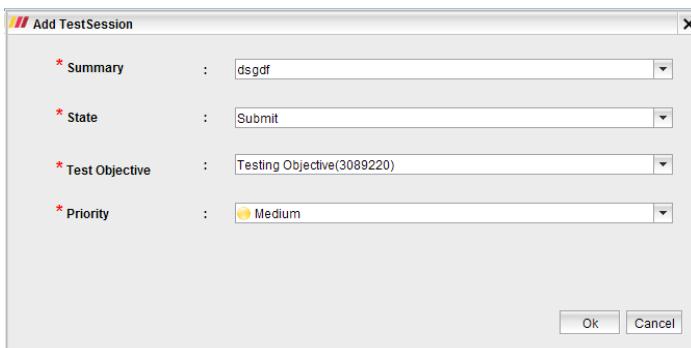


Figure 144: Add Test session window with complete details.

Newly added test session is added in the project like this.

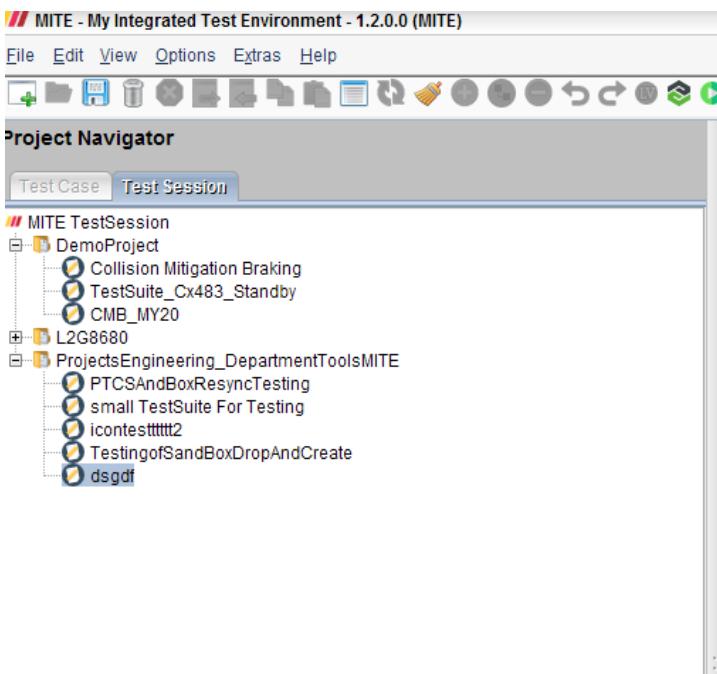


Figure 145: Test session added under project

Created with the Personal Edition of HelpNDoc: [Create HTML Help, DOC, PDF and print manuals from 1 single source](#)

24.3 Reload Project

In reload project, previously user does not had any Test Objective in the project, but now user can click on reload project option and get the test objectives from the PTC.

If any New test sessions are added in PTC while Reload project user will get that Test sessions also.

Reload project:

1. User can select the **Reload Project** in Project Right click action if any newly added Test Session in there in PTC as shown in below pop up:-

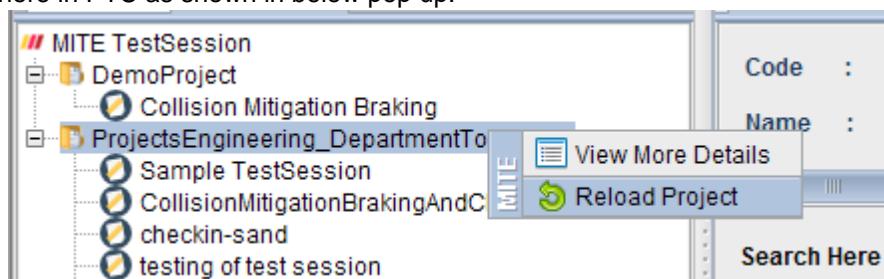


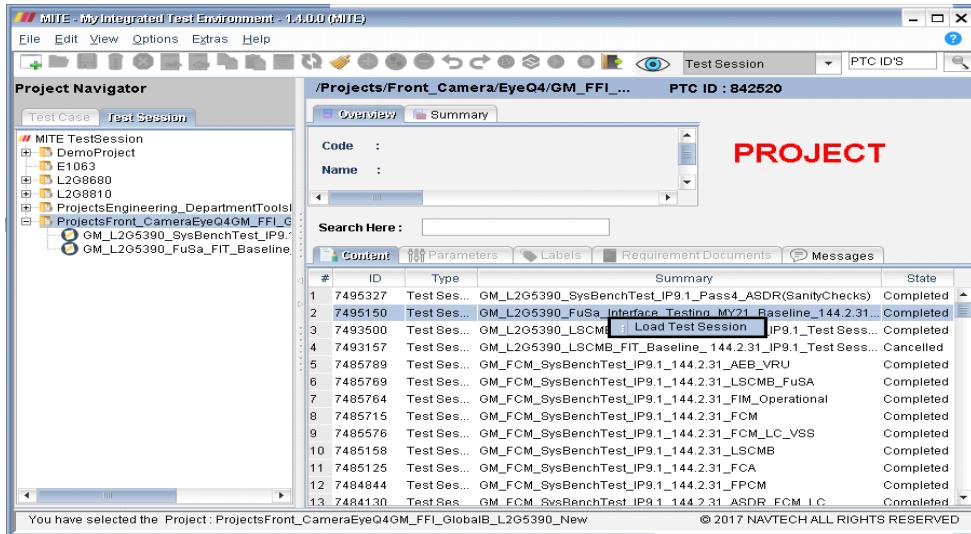
Figure 136: RELOAD PROJECT FROM TREE RIGHT CLICK

Created with the Personal Edition of HelpNDoc: [Free HTML Help documentation generator](#)

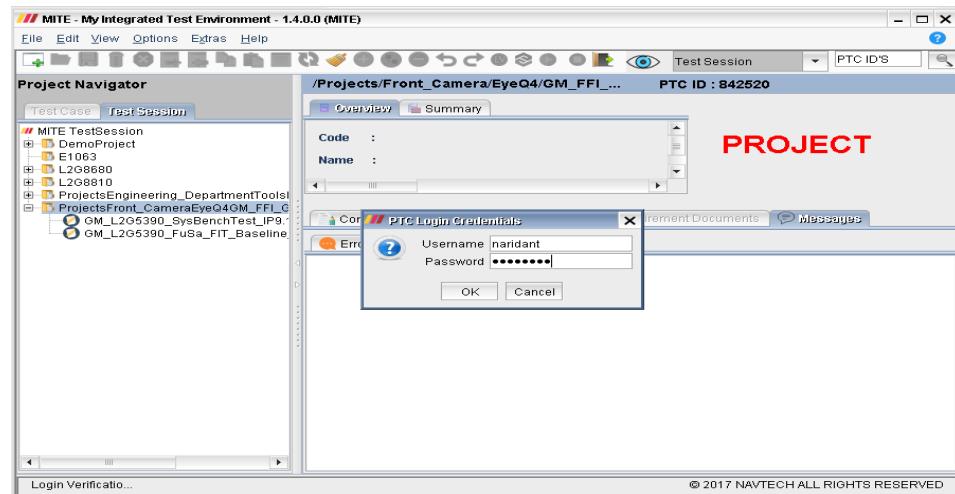
24.4 Load Test Session

Load Test Session from PTC: - Any Test-Session(s) from PTC can be loaded from PTC using “Load Test Session” option as shown below:

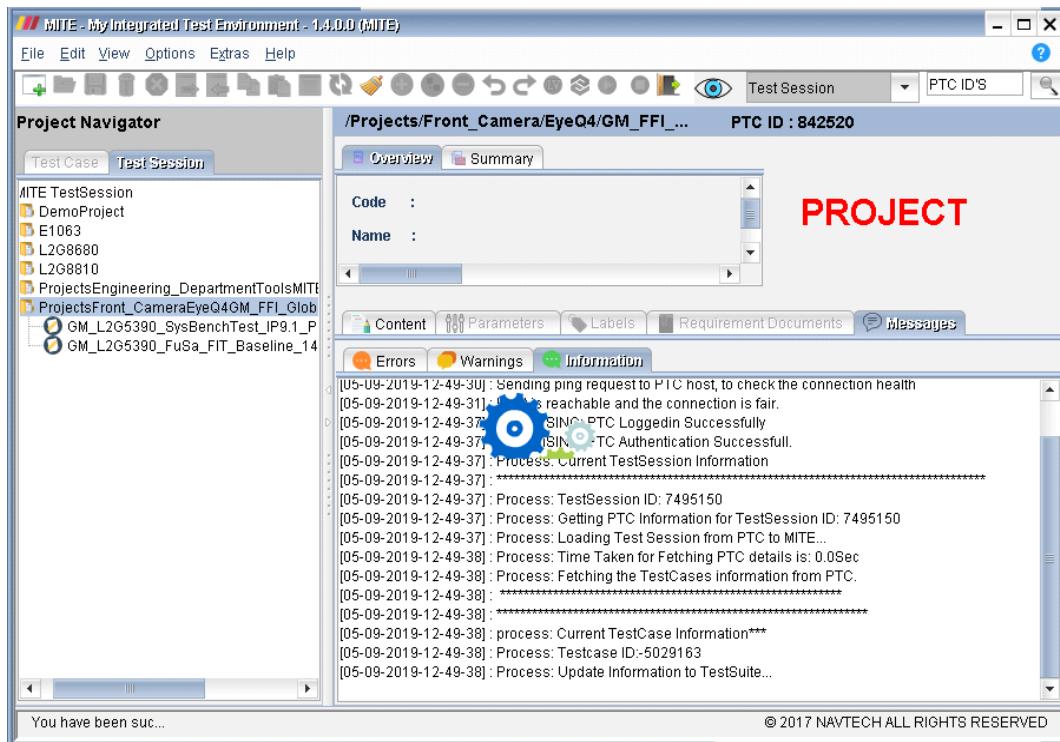
In project content table, select the test session you want to load from the list and right click on it. Now, Click on **Load Test Session** option.



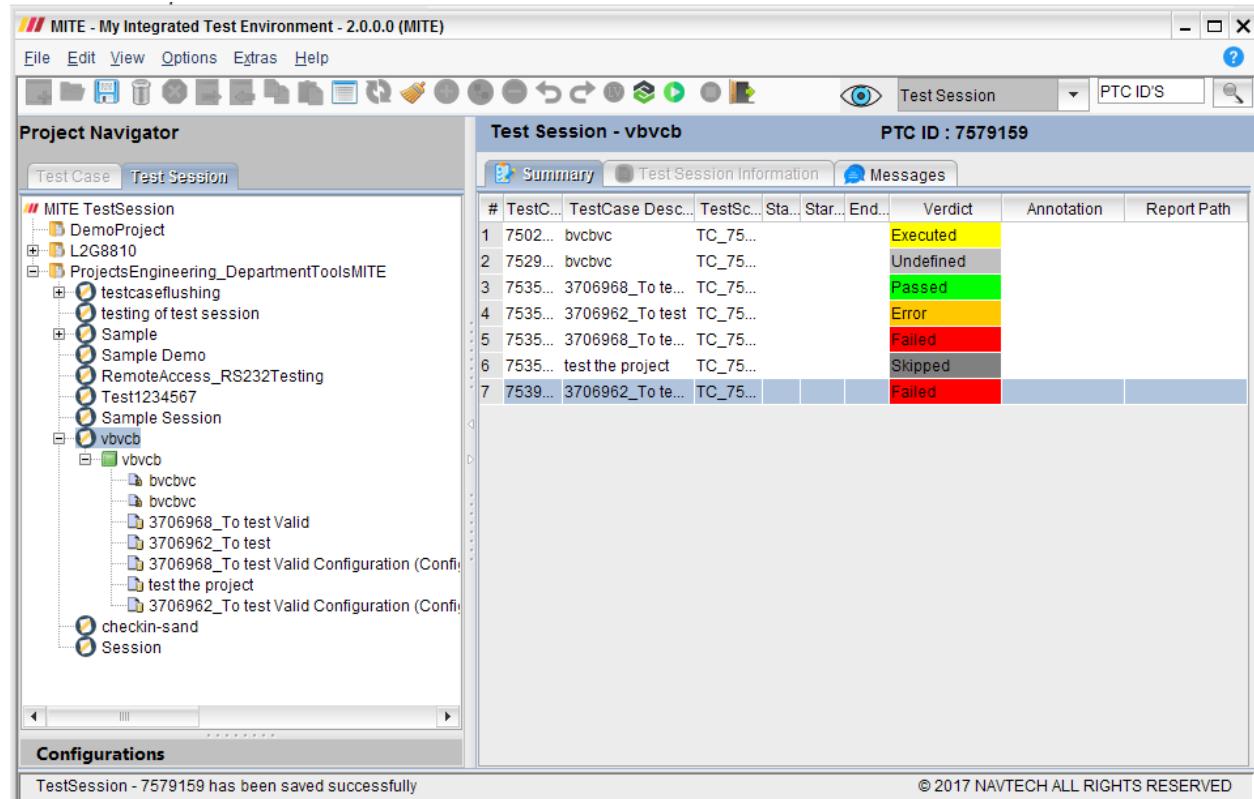
Before entering PTC credentials, make sure that your PTC is connected and its connection is strong. Enter valid **PTC credentials** and click on **OK**.



After login, load test session process starts and session will be added as shown below :



After completion of Load test session user will get the Test suites & Test cases of the session as shown below:



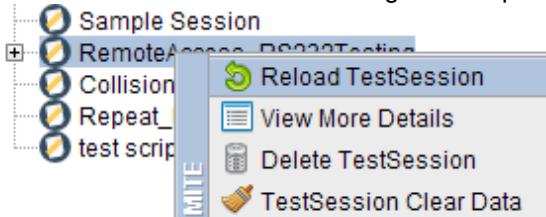
Created with the Personal Edition of HelpNDoc: [Easy to use tool to create HTML Help files and Help web sites](#)

24.5 Reload TestSession

Reload TestSession

1. User again adding any newly Test Case in PTC for the same suite and that newly added data he/she want it into MITE can choose the Reload Test session option.

2. It is available in Test session right click option:-



Reload Test session

Options available:-

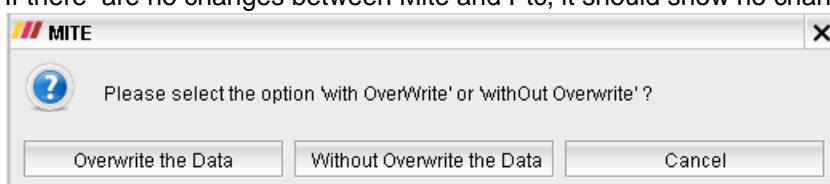
Newly add Test Cases, Verdict and annotation changed in PTC, it should show the pop up message to user will below options:-

Overwrite the data: When user clicked on this option, it should show the updated content in Mite Whatever is present in PTC.

Without over write the data: Whenever user clicked on this option, newly added test Cases should be added in to Mite at the same time PTC deleted test cases should be deleted in to Mite .

Cancel: If User does not need any of the options, he/she can click on this option and close the window.

If there are no changes between Mite and Ptc, it should show no changes has been found.



Created with the Personal Edition of HelpNDoc: Produce electronic books easily

24.6 Project creation and Load/Reload TBC

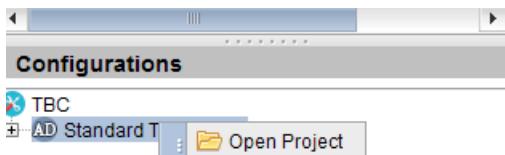
PROJECT CREATION

1) Before Project Creation in Test-Session, User needs to make COMInterfaceProjectCreation enable by enabling **ctrl+Shift+F**.



2) Before execution User should have project in Automation Desk.

If user already created project for respective Test Suite then he/she can right click on TBC and click on open project



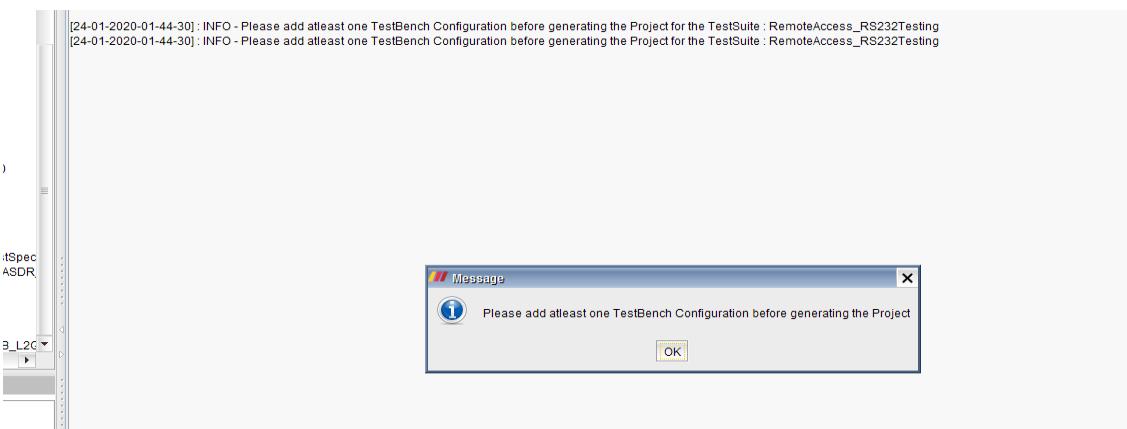
3) After successful opening of project and library in Automation Desk popup message will be shown:-

```

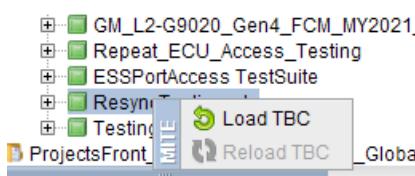
14 To check variable "M_AEBv_HoldExpired" TC_7502634 Complet... Thu Jan 23... Thu Jan 23... Error
12 To check variable "M_AEBv_OverrideBrakeDuring..." TC_7502632 Complet... Thu Jan 23... Thu Jan 23... Error
18 To check variable "M_AEB_AccelPedalPositionRate" TC_7502598 Complet... Thu Jan 23... Thu Jan 23... Error
18 To check variable "M_AEBr_HoldExpired" when cal... TC_7502638 Complet... Thu Jan 23... Thu Jan 23... Error
10 To check transition from ENABLED State to OFF St... TC_7761620 Complet... Thu Jan 23... Thu Jan 23... Error
16 To check variable "M_AEBr_HoldExpired" TC_7502636 Complet... Thu Jan 23... Thu Jan 23... Error
12 To check transition from ENABLED State to OFF St... TC_7761622 Complet... Thu Jan 23... Thu Jan 23... Error
14 To check transition from ENABLED State to OFF St... TC_7761624 Complet... Thu Jan 23... Thu Jan 23... Error
12 To check variable "M_AEBv_AccelPedalHome" wh... TC_7502582 Complet... Thu Jan 23... Thu Jan 23... Error
10 To check variable "M_AEBv_AccelPedalHome" wh... TC_7502580 Complet... Thu Jan 23... Thu Jan 23... Error
16 To check variable "M_AEBr_AccelPedalHome" Message
14 To check variable "M_AEBv_AccelPedalHome"
12 To check variable "M_AEBr_OVERRIDEACCEL"
18 To check variable "M_AEBr_AccelPedalHome"
10 To check variable "M_AEBv_OVERRIDEACCEL"
16 To check variables "M_AEBv_OVERRIDESTEER" a...
14 To check variable "M_AEBr_OVERRIDEACCEL" TC_7502624 Complet... Thu Jan 23... Thu Jan 23... Error
18 As per the defect 3572457 to check variables "M_A... TC_7502628 Complet... Thu Jan 23... Thu Jan 23... Error
13 To check transition from ENABLED State to OFF St... TC_7502493 Complet... Thu Jan 23... Thu Jan 23... Error
3910221 Description To set signal FwdCmMtgNbr... TC_7502652 Complet... Thu Jan 23... Thu Jan 23... Error
17 To check transition from ENABLED State to OFF St... TC_7502497 Complet... Thu Jan 23... Thu Jan 23... Error
10 To check transition from ENABLED State to DISAB... TC_7502530 Complet... Thu Jan 23... Thu Jan 23... Error

```

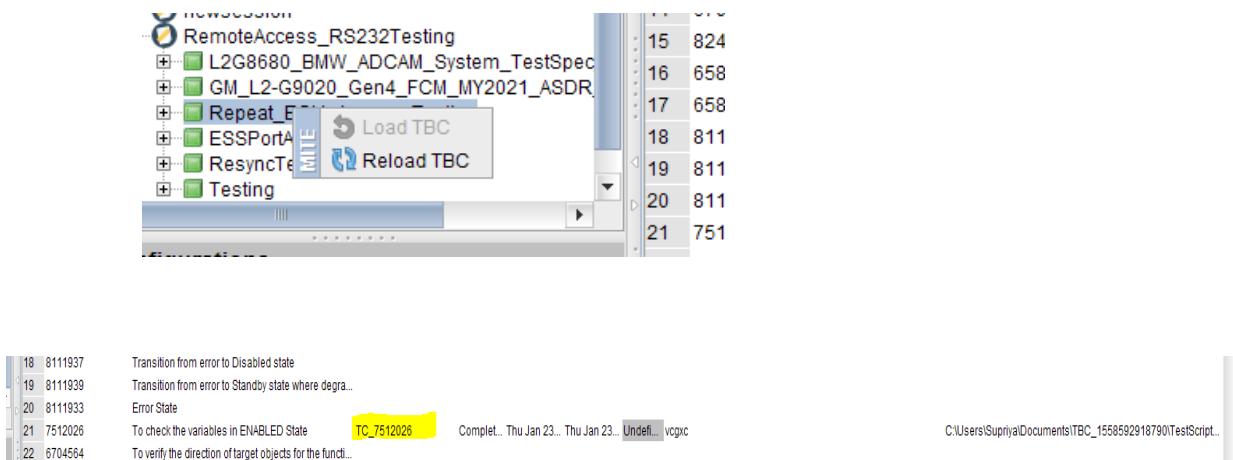
4) Else User can create Project by clicking Project Creation button, if its respective TBC is not present it will show a popup that TBC configuration is not added.



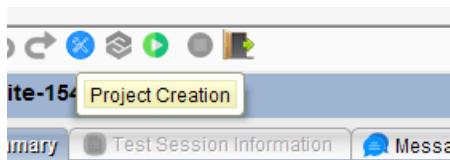
5) User need to load TBC, after loading TBC user can see for which Test Case ,Script is present in the Test Script Name column.



6) If any changes is there in TBC,User can reload TBC.



7) On clicking of Project Creation, Project will be created in Automation Desk.

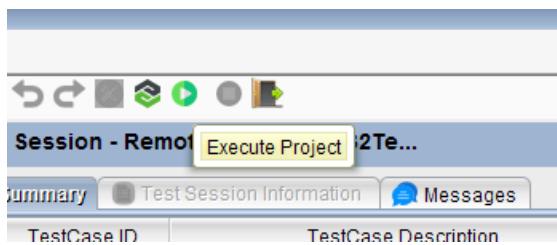


Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

24.7 Test Session START/STOP Execution

Test Session Start Execution

1. User can click on Test Session Project Execution button (Project Execute) as shown below:-



2. If those Scripts is present then in TestScriptName Column in Test Session Summary Panel those scripts only will be executed.

3. Respective Script present in Automation for project selected, Test Case will Start Executing from Automation Desk if the project and Script is present in Automation Desk.

4. The output will come in form of Verdict {Executed, Passed, Undefined, Failed, Error}.

Test Session - CollisionMitigationBraking...								PTC ID : 8281989		
	#	TestCase ID	TestCase Description	TestScript Name	Status	Start Time	End Time	Verdict	Annotation	Report Path
Sury	1	7502592	To check variable 'M_AEB_AccelPedalHome' wh...	TC_7502592	Complet...	Thu Jan 23...	Thu Jan 23...	Error		E:\AutomationDesk\Projects\AlternateProjects\TBC_153000629...
ig	2	7502590	To check variable 'M_AEB_AccelPedalHome' wh...	TC_7502590	Complet...	Thu Jan 23...	Thu Jan 23...	Passed		E:\AutomationDesk\Projects\AlternateProjects\TBC_153000629...
entToolsMITE	3	7502830	As per the Defect3572457 To check variables 'M...	TC_7502830	Complet...	Thu Jan 23...	Thu Jan 23...	Skipped		E:\AutomationDesk\Projects\AlternateProjects\TBC_153000629...
	4	7502596	To check no transition from Prell State to IBA Stat...	TC_7502596	Complet...	Thu Jan 23...	Thu Jan 23...	Undef		E:\AutomationDesk\Projects\AlternateProjects\TBC_153000629...
	5	7502594	To check variable 'M_AEB_BrakePressureRate'	TC_7502594	Complet...	Thu Jan 23...	Thu Jan 23...	Failed		E:\AutomationDesk\Projects\AlternateProjects\TBC_153000629...
	6	7502834	To check variable 'M_AEB_HoldExpiried'	TC_7502834	Complet...	Thu Jan 23...	Thu Jan 23...	Error		E:\AutomationDesk\Projects\AlternateProjects\TBC_153000629...
	7	7502832	To check variables 'M_AEBv_OverrideBrakeDuring...	TC_7502832	Complet...	Thu Jan 23...	Thu Jan 23...	Error		E:\AutomationDesk\Projects\AlternateProjects\TBC_153000629...

5.Based on this verdict, column named as Verdict will be updated in Test Session Table as executed, passed, undefined, and failed or Error.

6.The Report of this script execution will be in backend as html format and the report path will be updated in Report Path column in Test Session Table and on double clicking report will open:-

#	TestCase ID	TestCase Description	TestScript Name	Status	Start Time	End Time	Verdict	Annotation	Report Path
1	7502592	To check variable "M_AEB_AccelPedalHome" wh...	TC_7502592	Comple...	Thu Jan 23...	Thu Jan 23...	Error		E:\AutomationDesk\Projects\AlternateProjects\IBC_153000629...
2	7502590	To check variable "M_AEB_AccelPedalHome" wh...	TC_7502590	Comple...	Thu Jan 23...	Thu Jan 23...	Passed		E:\AutomationDesk\Projects\AlternateProjects\IBC_153000629...
3	7502630	As per the Defect 3572457 To check variables "M...	TC_7502630	Comple...	Thu Jan 23...	Thu Jan 23...	Skipped		E:\AutomationDesk\Projects\AlternateProjects\IBC_153000629...
4	7502598	To check no transition from Prefull State to IBA Stat...	TC_7502598	Comple...	Thu Jan 23...	Thu Jan 23...	Undef...		E:\AutomationDesk\Projects\AlternateProjects\IBC_153000629...
5	7502594	To check variable "M_AEB_BrakePressureRate"	TC_7502594	Comple...	Thu Jan 23...	Thu Jan 23...	Failed		E:\AutomationDesk\Projects\AlternateProjects\IBC_153000629...
6	7502634	To check variable "M_AEB_HoldExpired"	TC_7502634	Comple...	Thu Jan 23...	Thu Jan 23...	Error		E:\AutomationDesk\Projects\AlternateProjects\IBC_153000629...
7	7502632	To check variables "M_AEBv_OverrideBrakeDuring...	TC_7502632	Comple...	Thu Jan 23...	Thu Jan 23...	Error		E:\AutomationDesk\Projects\AlternateProjects\IBC_153000629...
8	7502598	To check variable "M_AEB_AccelPedalPositionRate"	TC_7502598	Comple...	Thu Jan 23...	Thu Jan 23...	Error		E:\AutomationDesk\Projects\AlternateProjects\IBC_153000629...

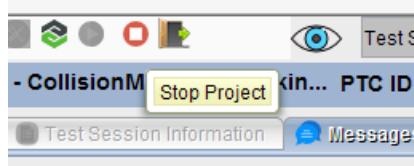
7.With this Status, Start Time of execution end time of execution will also be updated.

The screenshot shows the dSPACE AutomationDesk Report window. At the top, it displays the report title 'dSPACE' and 'TC_7502592'. Below this, it shows the result date and time: 'Result_2020/01/23 12:42:50'. A note indicates the report was generated by 'dSPACE report library'. The main content area is divided into several sections:

- Execution Information:** Shows start time (2020-01-23 12:42:50), duration (0.040 seconds), executed element (TC_7502592 (Sequence)), termination state (Finished), and library information (Online(11)).
- Sequence Statistics:** A progress bar indicating 100% completion with one error (1) highlighted.
- Overview of Sequences:** A table showing the sequence details for TC_7502592 (Sequence). It lists the name (TC_7502592), execution duration (0.040 seconds), and start time (2020-01-23 12:42:50).

Test Session Stop Execution

1.On clicking stop execution button (Stop Project) the execution running will be stopped



PTC SUBMIT:

1.After execution User can flush the results into PTC at Test-Session level as shown below:-

The screenshot shows the MITE application interface. At the top, there's a toolbar with various icons. Below it is a main window titled "Test Session - RS232Te...". The window has tabs: "Summary" (which is selected), "Test Session Information", and "Messages". The "Summary" tab displays a table with two rows:

#	TestCase ID	TestCase Description
1	8111927	Transition from Disable to Standby State where s
2	8111929	Transition from Disable to Unavailable state

Below this, there's a message bar indicating "Test Session: 8281989 - bharathi.kolla@magnalm-prd.ptcmscloud.com:7001". There are tabs for "Item", "Test" (which is selected), and "View". Under "Test", there are icons for "New", "Edit", "Delete", "Copy", "Paste", and "Print".

Further down, there's a section titled "Test Session: 8281989" with details: "Created by Bharathi Kolla (bharathi) on Jan 6, 2020 2:51:24 PM" and "Modified by Bharathi Kolla (bharathi) on Jan 6, 2020 3:41:01 PM". A navigation bar below includes "Properties", "Configuration", "Parameters", "Metrics", "Relationships", "Attachments", "Test Results" (which is checked), "Branches", "Time Entries", "Workflow", and "History".

A table titled "Show results that are" lists test cases and their verdicts:

Test Session	Test Case	Verdict	Annotation
8281989	7502493	Failed	
8281989	7502495	Failed	
8281989	7502497	Failed	
8281989	7502499	Failed	
8281989	7502526	Failed	
8281989	7502528	Failed	
8281989	7502530	Failed	
8281989	7502532	Failed	
8281989	7502534	Failed	
8281989	7502536	Failed	

Created with the Personal Edition of HelpNDoc: [Create cross-platform Qt Help files](#)

24.8 Test Objectives

To fetch Test Objectives from PTC, User should connect with PTC from MITE. To connect with PTC we need to login with PTC credentials through MITE.

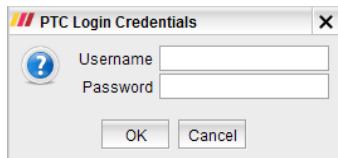


Figure 137: PTC LOGIN CREDENTIALS

If user logged in successfully, then user can fetch test objectives from PTC after internal process as like below

The screenshot shows the MITE interface with a "PROJECT" tab selected. The "PROJECT" tab displays a log of events:

- Code : ...
- Name : ...
- Project : ...
- Created by : e38903 on Mar 8, 2017 12:29:09 AM
- Modified by : ganeshm on Aug 19, 2019 14:40:40 PM

The log entries include:

- 27/08/2019 11:41:48] Int. Starting for Reload Project
- 27/08/2019 11:42:01] Host is reachable and the connection is fair.
- 27/08/2019 11:42:01] Host is reachable and the connection is fair.
- 27/08/2019 11:42:01] Int. Starting for Reload Project
- 27/08/2019 11:42:04] sending ping request to PTC host to check the connection health
- 27/08/2019 11:42:05] Host is reachable and the connection is fair.
- 27/08/2019 11:42:05] Int. Starting for Reload Project
- 27/08/2019 11:42:05] Int. PTC Authentication is Started
- 27/08/2019 11:42:05] Int. PTC Authentication is Started
- 27/08/2019 11:42:05] Process: Related Project Information -
- 27/08/2019 11:42:05] Process: Related Project Information -
- 27/08/2019 11:42:05] Process: Project Name -null
- 27/08/2019 11:42:05] Process: Fetching the Test Session Data in the Project

Figure 138: PROCESS OF FETCHING TEST OBJECTIVES FROM PTC

Once test objectives are fetched from PTC, Then process will be terminated as shown in below.

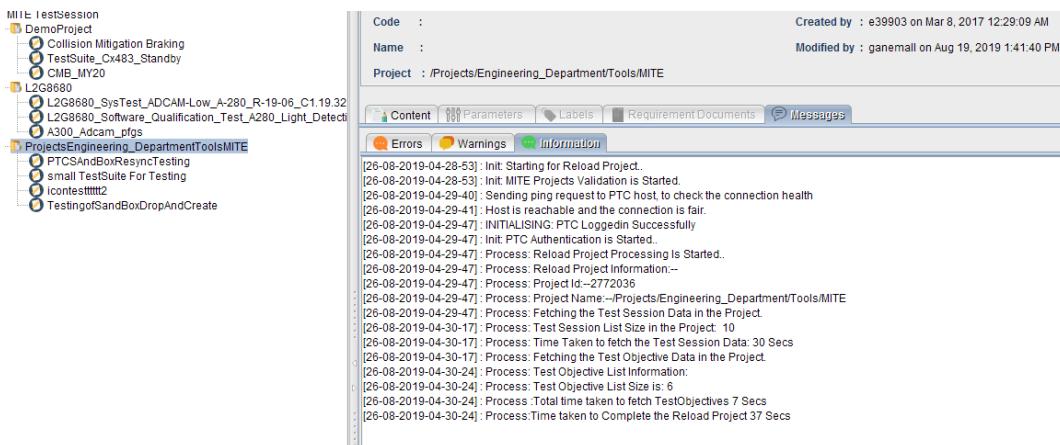


Figure 139: COMPLETION OF FETCHING TEST OBJECTIVES FROM PTC

All the test objectives are added in to directly in the add test session dialog box, as shown in below

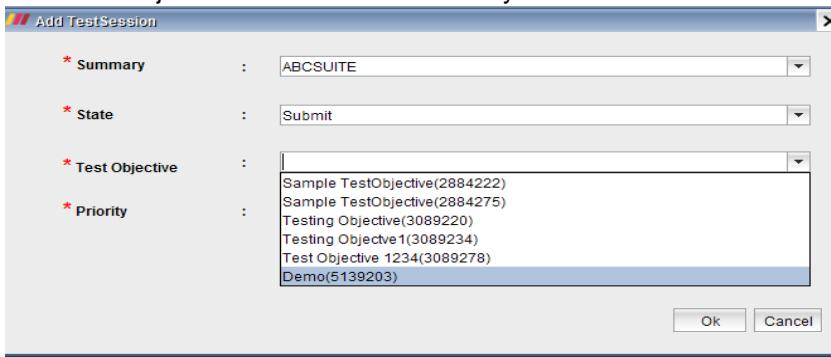


Figure 140: LIST OF TEST OBJECTIVES TO ADD TESTSESSION IN PROJECT

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

24.9 Delete Test Session

When User want to delete the Test Session, it is added in the Test Session right click option. Now user can click on that for performing delete test session.

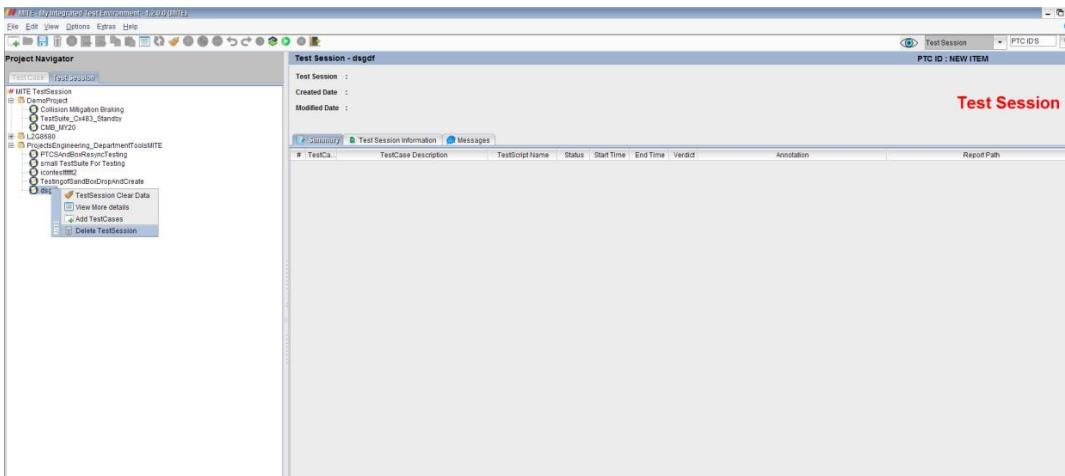


Figure 146: DELETE TEST SESSION OPTION FROM TREE RIGHT CLICK POP-UP

It asks the user for confirmation to delete, click on yes- it should delete otherwise click on No-it is under the project

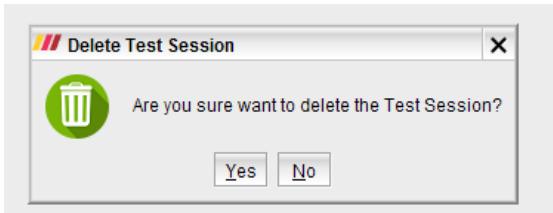


Figure 147: CONDITIONAL DIALOGUE BOX FOR DELETE TEST SESSION.

Created with the Personal Edition of HelpNDoc: [Create help files for the Qt Help Framework](#)

24.10 Consolidated HTML report

With the current implementation of MITE-v2.1.8.1 ,User will be able to get consolidated Test Run Report in one HTML report.

Step 1 :

When User will right click on Test Session level,s/he will be able to view Generate HTML report as shown below:-

#	TestCase ID	Test
1	8430370	StandardPreCon
2	8430372	StandardPostCo
3	8430376	To verify ECU Re
4	8430378	To verify Hard Re
5	8430380	To verify SID11 I
6	8430382	To verify After a r
7	8430386	To verify NRC 12
8	8430388	To verify NRC 13
9	8430390	To verify NRC 7E
10	8430392	To check defined
11	8430396	To verify TAKE C
12	8430400	NRC 13 Invalid I
13	8430402	NRC 12 Subfun
14	8430404	NRC 35 Invalid I
15	8430406	NRC 24 Request
16	8430408	NRC 37 Require

Step 2:-User will click on Generate HTML Report then it will redirect the HTML report in browser level as shown below:-

The screenshot shows a web-based test report interface. At the top, there's a header with the MITE logo and the title "MITE TEST RUN REPORT". Below the header, a sub-header says "Report Generated by MITE". The main content area is divided into sections: "TEST-SESSION DETAILS" and "TEST RUN DETAILS". The "TEST-SESSION DETAILS" section contains tables with session metadata like name, ID, state, and project. The "TEST RUN DETAILS" section contains a large table with columns for Test-Suite ID, Test-Case ID, Execution Duration, Verdict, and Report-Path. Each row in the table corresponds to a test case with its specific details.

Test-Suite ID	Test-Case ID	Execution Duration	Verdict	Report-Path
8430369	8430414	0.7	Passed	TC_Report_8430414
8430369	8430404	20.5	Skipped	TC_Report_8430404
8430369	8430402		Failed	TC_Report_8430402
8430369	8430424		Failed	TC_Report_8430424
8430369	8430388		Passed	TC_Report_8430388
8430369	8430378		Passed	TC_Report_8430378
8430369	8430400		Passed	TC_Report_8430400
8430369	8430422	0.9	Passed	TC_Report_8430422
8430369	8430386		Passed	TC_Report_8430386
8580093	85800312		Passed	TC_Report_85800312
8430369	8430376		Failed	TC_Report_8430376
8430369	8430396		Error	TC_Report_8430396
8430369	8430382		Failed	TC_Report_8430382
TestSuite-1580186926341	8430380		Skipped	TC_Report_8430380

**NOTE : If user will click on Report Path it will redirect to Automation Desk report.
: User will get report only for test case which have verdict and Report Path .**

	Verdict	Report-Path
	Passed	TC_Report_8430414
	Skipped	TC_Report_8430404
	Failed	TC_Report_8430402
	Failed	TC_Report_8430424
	Passed	TC_Report_8430388
	Passed	TC_Report_8430378
	Passed	TC_Report_8430400
	Passed	TC_Report_8430422
	Passed	TC_Report_8430386
	Passed	TC_Report_85800312
	Failed	TC_Report_8430376
	Error	TC_Report_8430396
	Failed	TC_Report_8430382
	Skipped	TC_Report_8430380

Created with the Personal Edition of HelpNDoc: [Easily create EBooks](#)

24.11 Global Header Test Run details

Description:- In Customized Tree table user can perform different action like add rows, tables, delete rows ,delete tables save the data and reset the table.

- In Test Session level left side of the window when we will select the project, inside the project test session is there.

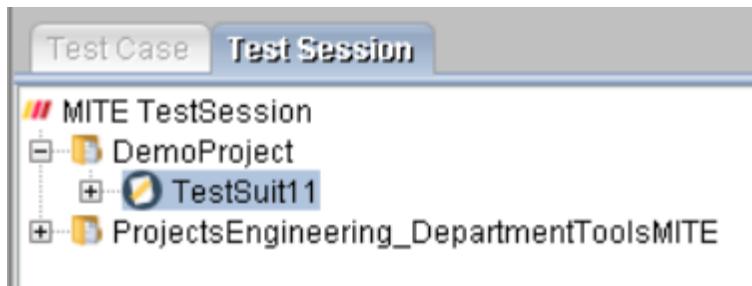


Fig.1

- In test session user perform right click.



Fig.2

After right click on Tree Table Information user will get Test Run Details Table.

Tree Name	Enable	Sno	ParameterName	ParameterValue
Test Run Details	<input checked="" type="checkbox"/>			
▶ Global Information	<input checked="" type="checkbox"/>			
▶ system Information	<input checked="" type="checkbox"/>			
▶ project	<input checked="" type="checkbox"/>			
▶ general info	<input checked="" type="checkbox"/>			
▶ check	<input checked="" type="checkbox"/>			
▶ Mite	<input checked="" type="checkbox"/>			

Fig. 3

In this table user can generate new table or update the previous table.

- In global information user can get general information about test session like test suite names and ids.

Tree Name	Enable	Sno	ParameterName	ParameterValue
Test Run Details	<input type="checkbox"/>			
▶ Global Information	<input checked="" type="checkbox"/>		SerialNum...	ParameterName
		1	TestSuit11	ParameterValue
		2	TestSuit11	11
▶ system Information	<input checked="" type="checkbox"/>			
▶ project	<input type="checkbox"/>			

Fig.4

ADD:-

If user wants to add another table he can add, for adding the table he has to select header (Test Run Details) and then has to press add button. After pressing the Add button user will get the popup

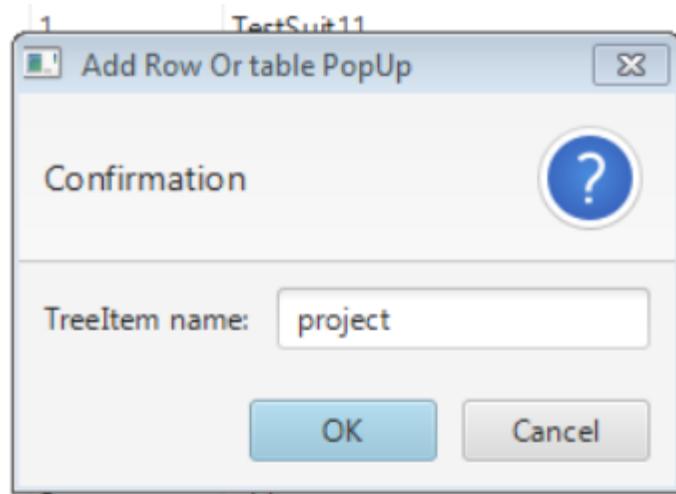


Fig.5

User has to enter the name then it will be show as a table's name
After adding the table user will get by Default three rows with table header.

Tree Name	Enable	Sno	ParameterName	ParameterValue
Test Run Details	<input type="checkbox"/>		SerialNum...	ParameterName
Global Information	<input checked="" type="checkbox"/>	1	TestSuit11	11
		2	TestSuit11	
system Information	<input checked="" type="checkbox"/>			
project	<input type="checkbox"/>		SerialNum...	ParameterName
		1		ParameterValue
		2		
		3		

Fig.6

Note :-If user enter any data in the cell he has to press enter key for storing the data or edit the data inside the cell.

If user wants to add more rows he can add by selecting the table header or any row.

Test Run Details				
Tree Name	Enable	Sno	ParameterName	ParameterValue
▼ Test Run Details	<input type="checkbox"/>			
▼ Global Information	<input checked="" type="checkbox"/>			
			SerialNum...	ParameterName
			1	TestSuit11
			2	TestSuit11
► system Information	<input checked="" type="checkbox"/>			
▼ project	<input type="checkbox"/>			
			SerialNum...	ParameterName
			1	
			2	
			3	
			4	
			5	

Fig .7

Delete:-

Delete table:-

For delete the table user has to select the tree name.

Delete row:-

For delete the row user has to select the table header or row.

Reset:-

If user did some changes or delete something, if he didn't save the action then he can get the previous data by Reset Button.

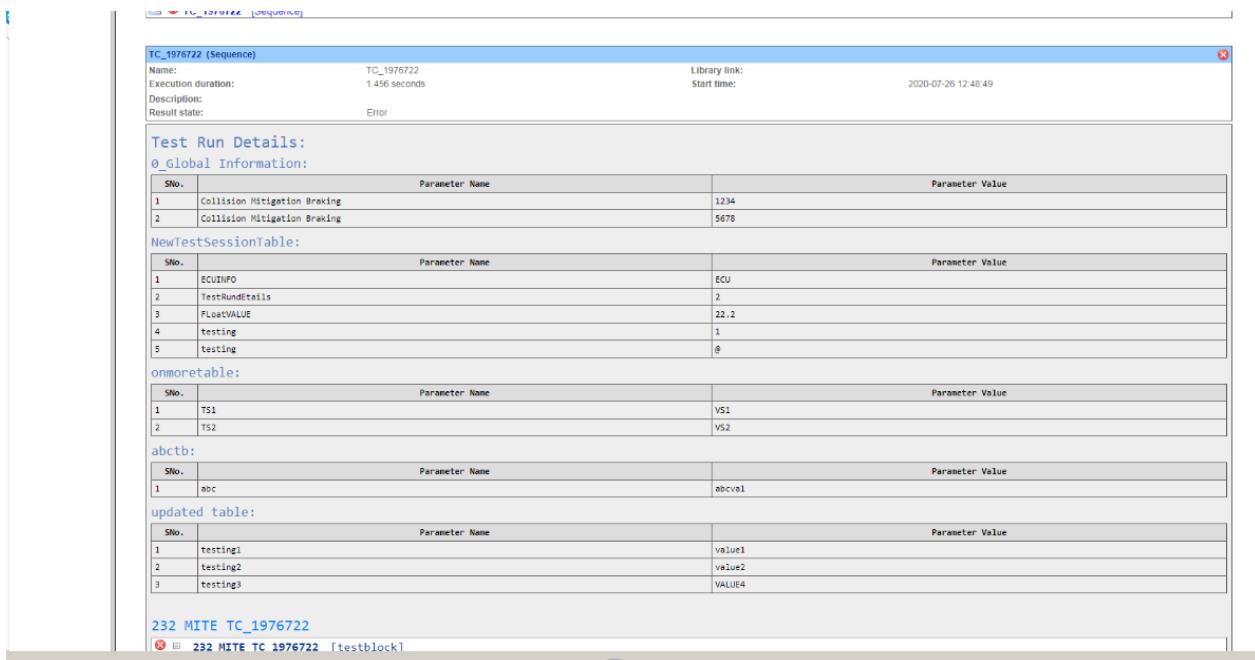
Save:-

Using save button user can save their data in backend.

After adding table check the table which needs to be updated in report

#	Tree Name	Enable	Sno	ParameterName	ParameterValue
197	Test Run Details				
197	Global Information	<input checked="" type="checkbox"/>		SerialNum...	ParameterName
197			1	Collision Mitigation Braking	1234
197			2	Collision Mitigation Braking	5678
197	NewTestSessionTable	<input checked="" type="checkbox"/>		SerialNum...	ParameterName
197			1	ECUINFO	ECU
197			2	TestRundetails	2
197			3	FLoatVALUE	22.2
197			4	testing	1
197			5	testing	@
197	updated table	<input checked="" type="checkbox"/>		SerialNum...	ParameterName
197			1	testing1	value1
197			2	testing2	value2
197			3	testing3	VALUE4
197	onmoretable	<input checked="" type="checkbox"/>		SerialNum...	ParameterName
197			1	TS1	VS1
197			2	TS2	VS2

Generate Project for test-Suite from Test-Session and execute .
Output :



Created with the Personal Edition of HelpNDoc: [Full-featured multi-format Help generator](#)

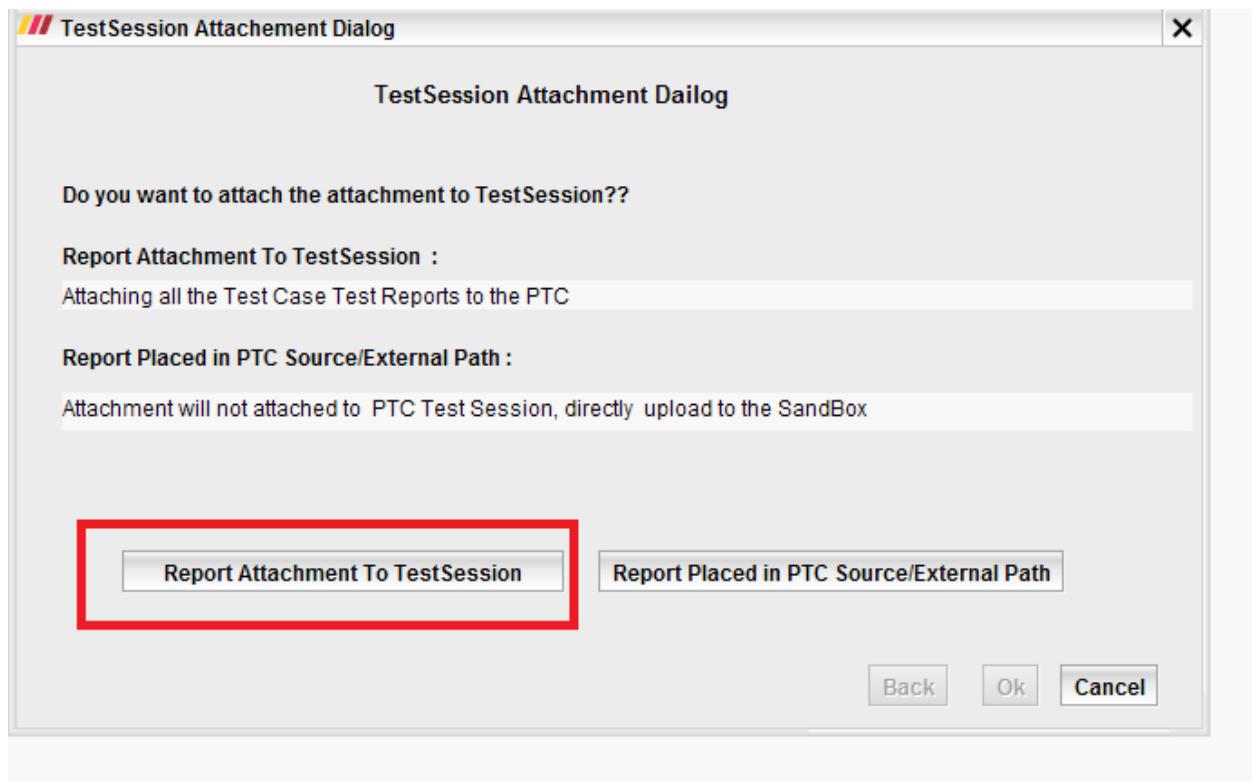
24.12 Test Session Flushing into PTC Report Path

Description:-

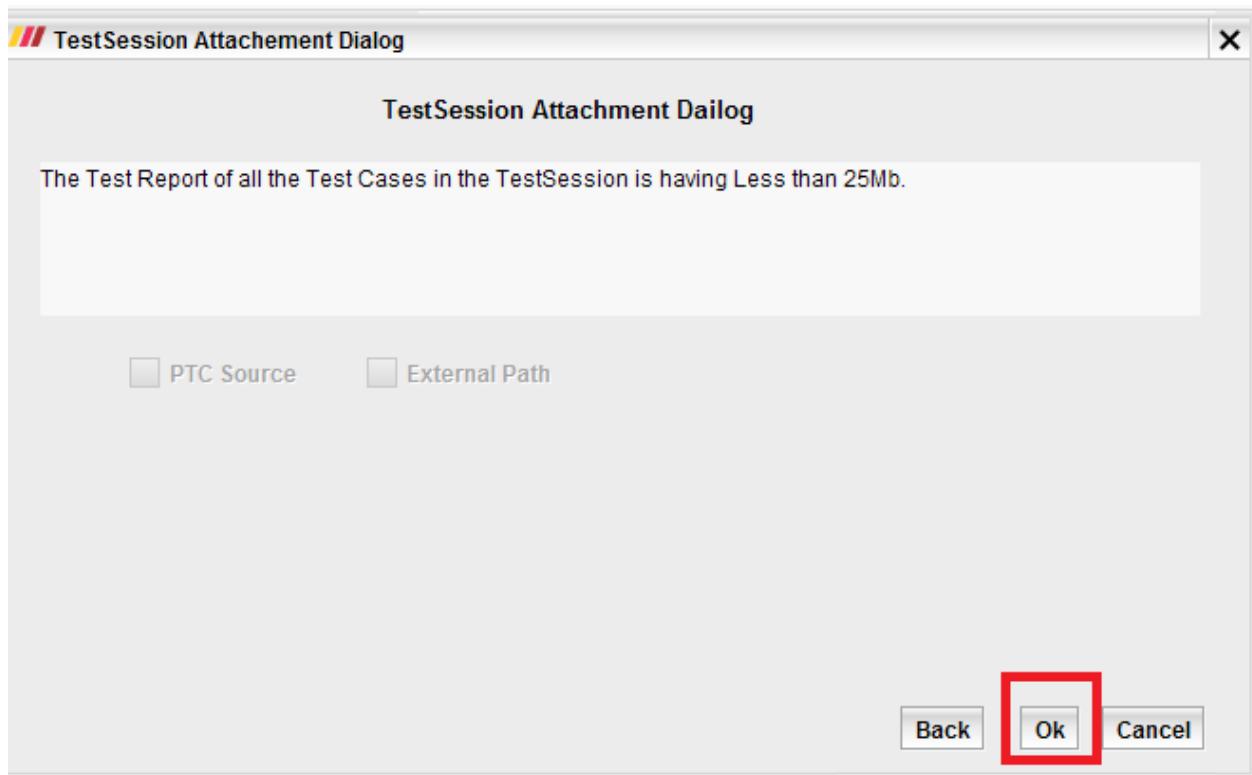
When User will flush the test cases in Test session into the PTC if the test case report is less than 25MB the flushing is done successfully. But when the test case report size is more than 25MB then flushing is not done from our end as because Test Session results file which is greater than 25 MB is not uploaded to PTC due to the restriction in the PTC.

With Version 2.2.0.0 User will be able to get the report path on clicking the Flush Icon which will shown as below:-

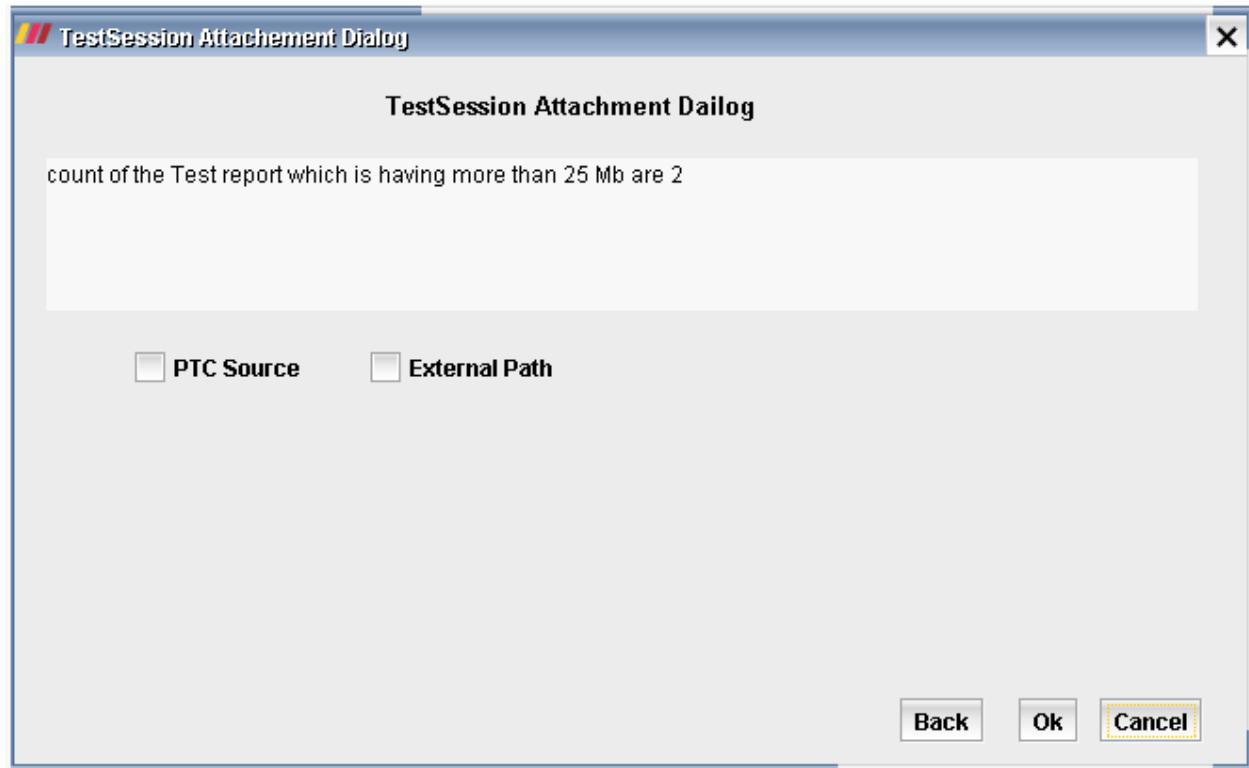
Step 1:-After clicking on Flush icon User can click on any option as suitable shown below-



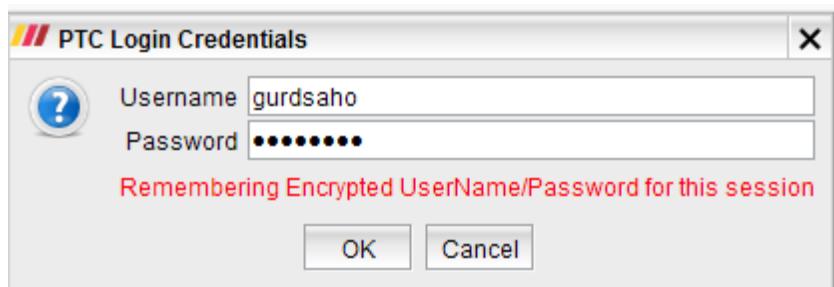
Step 2:-User will be able to view the screen as displayed below and choose the appropriate path for the reports:-



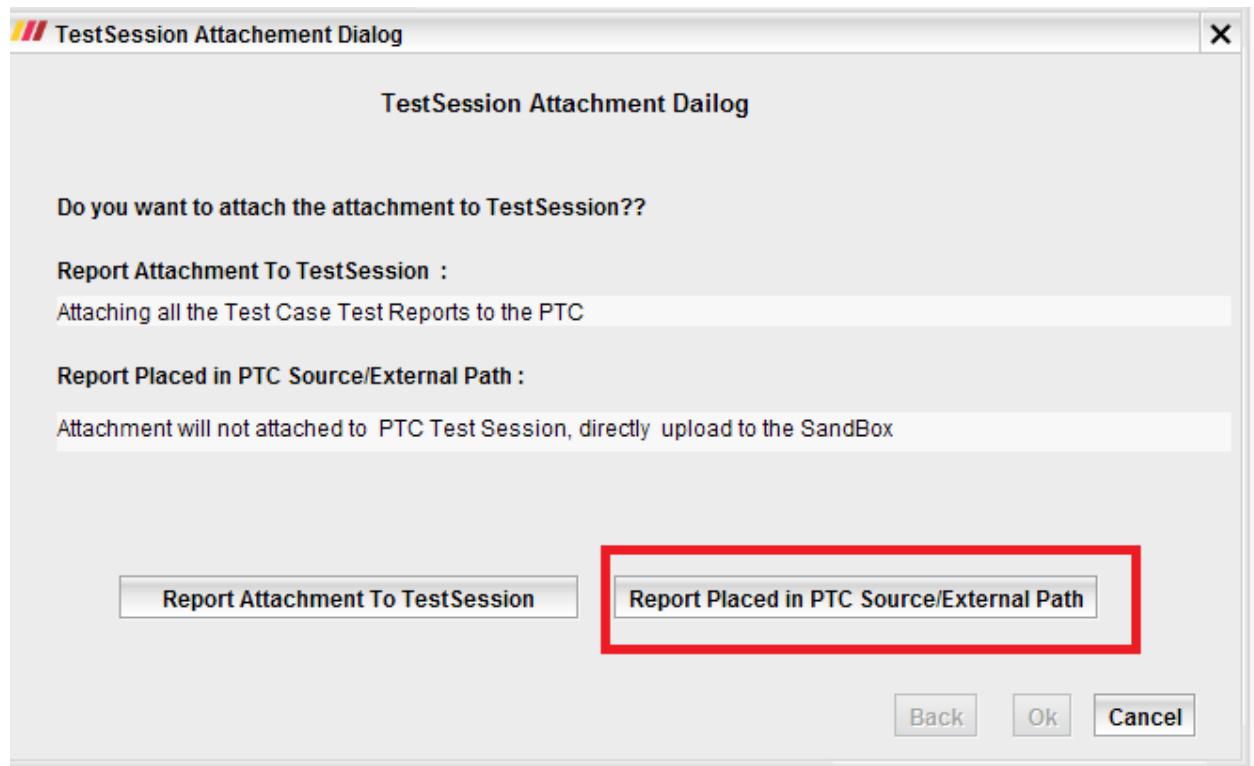
If it exceed more than 25 mb it will be shown in following path as shown below



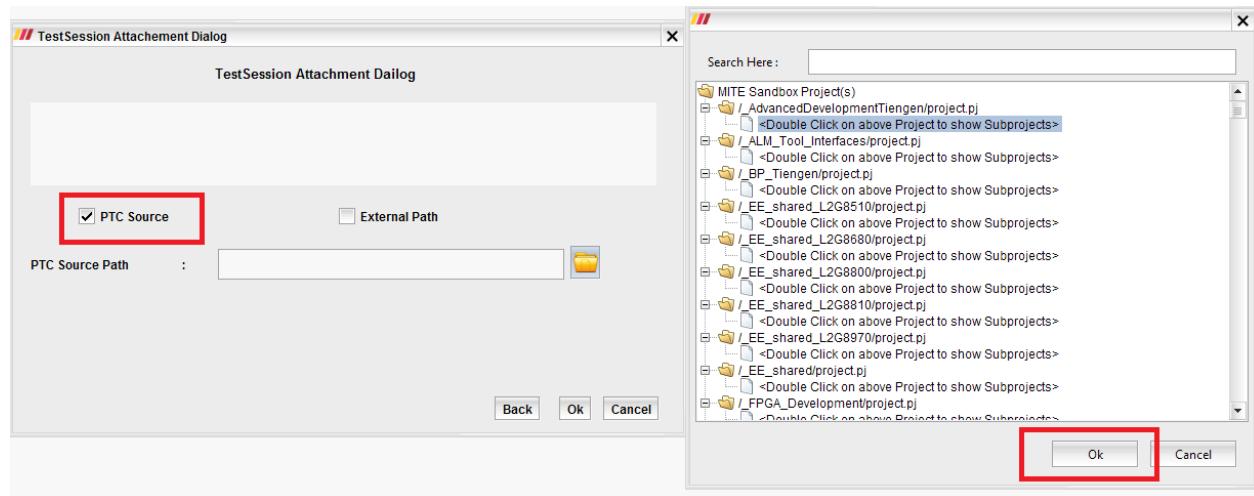
Step 3:-User need to click the PTC credentials and click on OK option as shown below:-



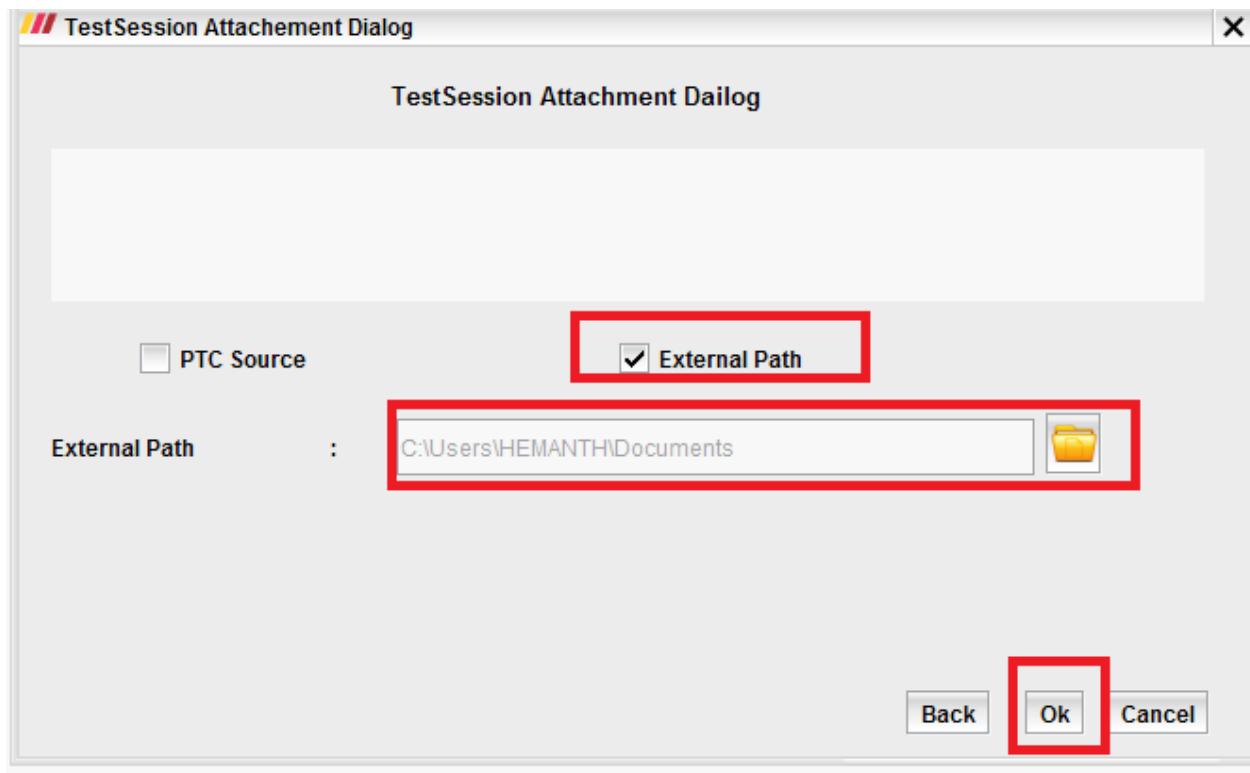
Step 4:-User need to choose other option to redirect to sandbox path or external path as displayed below:-



Step 5:-On clicking the PTC source check box and providing PTC credentials it will ask for sandbox projects as shown below and click on OK:-



Step 6:-User need to provide External path to get the report as shown below:-



Created with the Personal Edition of HelpNDoc: [Produce online help for Qt applications](#)

24.13 Multiple Time Execution without user intervention

Description : User can execute a script multiple times by using number of iteration column .

Step 1 :

Enter how many time user want to execute the script in Number of Iteration

Step 2

Generate project for that Test - Suite

Step 3

Execute test-Session

Ouptput : Script will be executed 2 times if you have given 2 in number of iteration

#	TestCase ID	TestCase Description	TestScript Name	Status	StartTime	End Time	Verdict	Annotation	Report Path	Number of Iterations
1	1976724	To check transition from ENABLED State to...	TC_1976724	Success	2023-09-11 10:30:00	2023-09-11 10:30:00	Pass			2
2	1976722	To check the variables in ENABLED State	TC_1976722	Success	2023-09-11 10:30:00	2023-09-11 10:30:00	Pass			3
3	1976723	To check signals CPS_Collision_Preparat...	TC_1976723	Success	2023-09-11 10:30:00	2023-09-11 10:30:00	Pass			4

Created with the Personal Edition of HelpNDoc: [News and information about help authoring tools and software](#)

25 Test Case Flush

Created with the Personal Edition of HelpNDoc: [What is a Help Authoring tool?](#)

25.1 Description

Previously if User wants to Flush the changes to PTC then he/she had to flush the Test Suite to PTC so that all the updated data will be present in PTC integrity.

As a result of which if User make changes in few of the test cases then he/she had to submit Whole TestSuite to PTC integrity which will consume time to flush in PTC integrity.

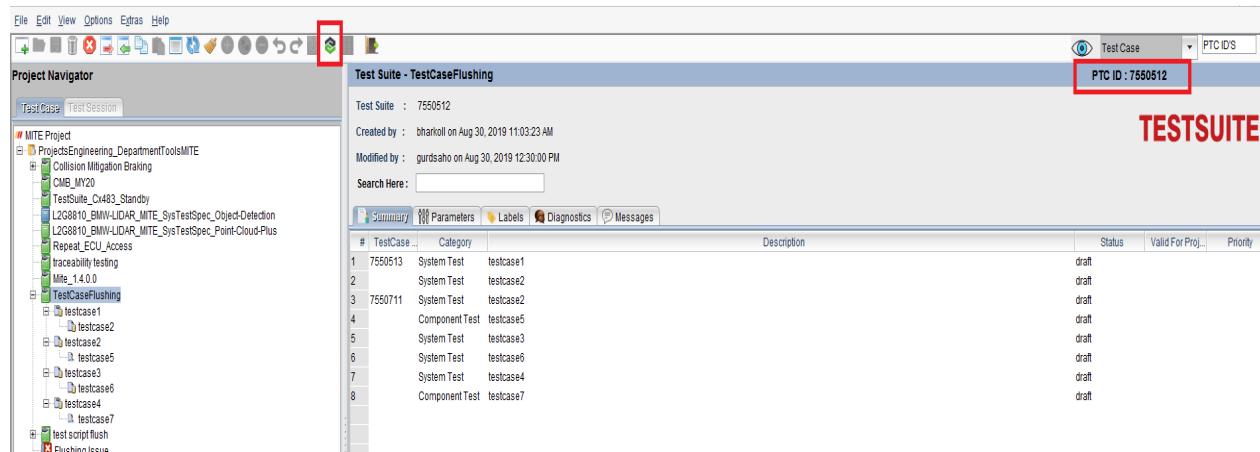
Thereafter Test Case Flush came into Picture for easy and fastest method to Submit the necessary Modified/Changed Test Cases into the PTC integrity

Created with the Personal Edition of HelpNDoc: [Easily create Help documents](#)

25.2 Implementation

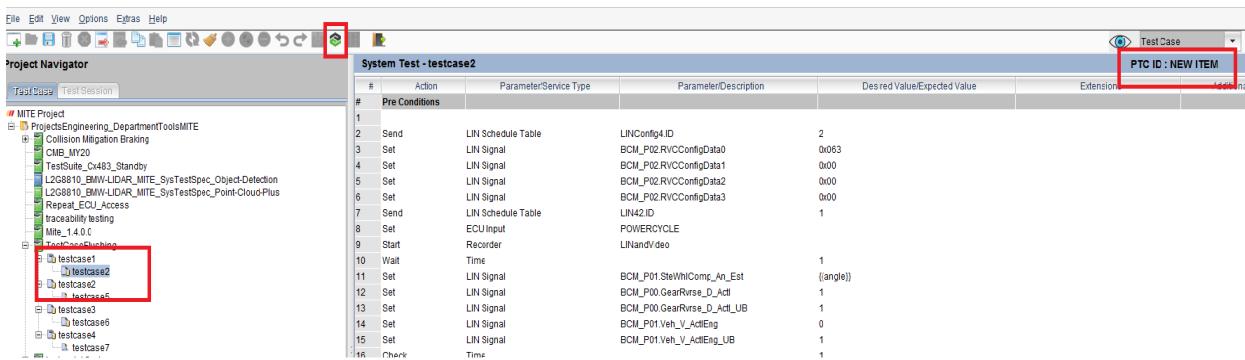
Implementation:-

- User can Submit/Flush the Test Case only if the Test Suite has the corresponding PTC id as shown below:-

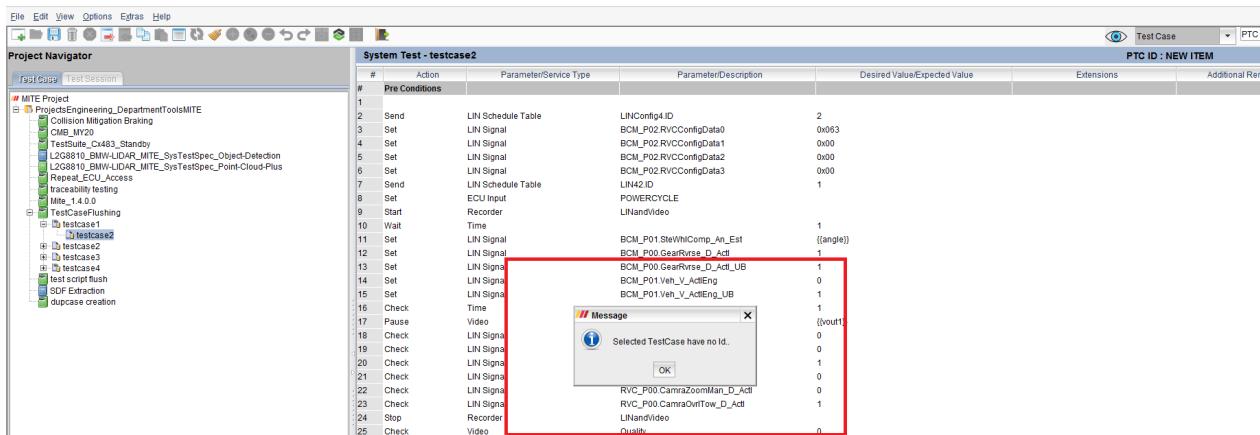


Now User wants to Modify only Individual Test Case and after Flushing User can see below image:-

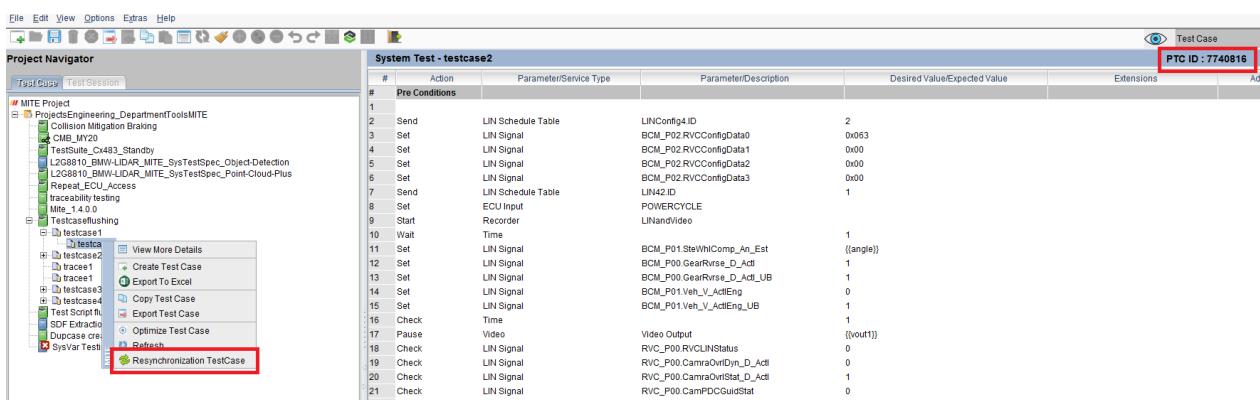
User cannot Flush the individual Test Case if it does not have PTC id as shown below:-



If User will submit the Test Case then he/she will get the following pop up:-



3)User Can update the data from PTC by clicking on Resync operation as shown below:-



Created with the Personal Edition of HelpNDoc: Benefits of a Help Authoring Tool

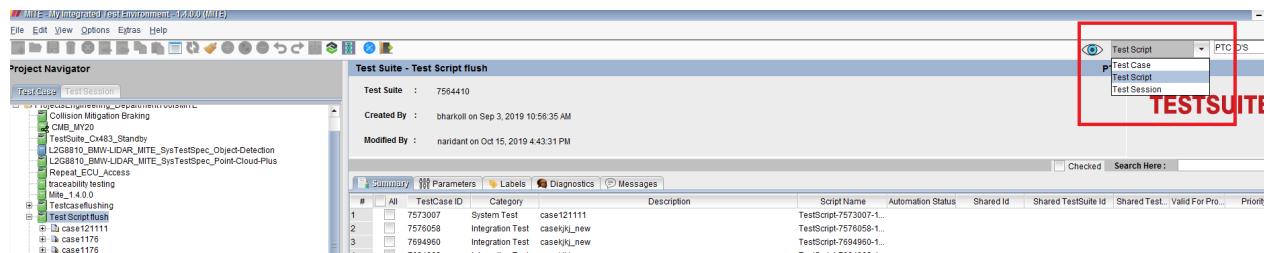
26 TEST Script Flush

Description:-

We have implemented the following process for the Test Script Flush:-

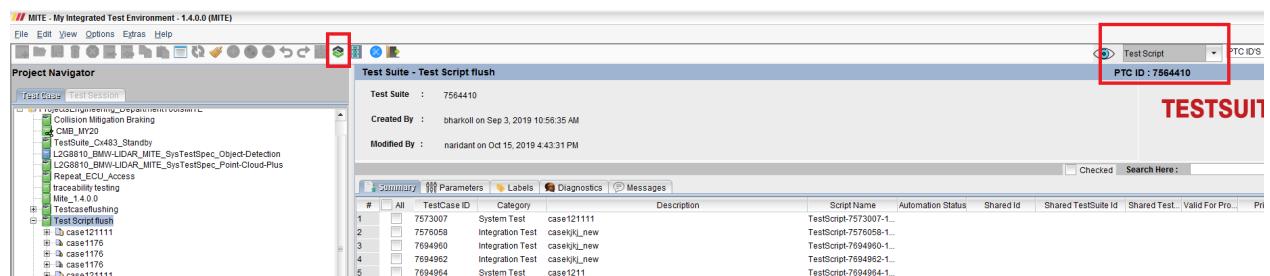
Step 1:-

User should go to right side of the screen and drop down the cursor as shown below to click on Test Script level:



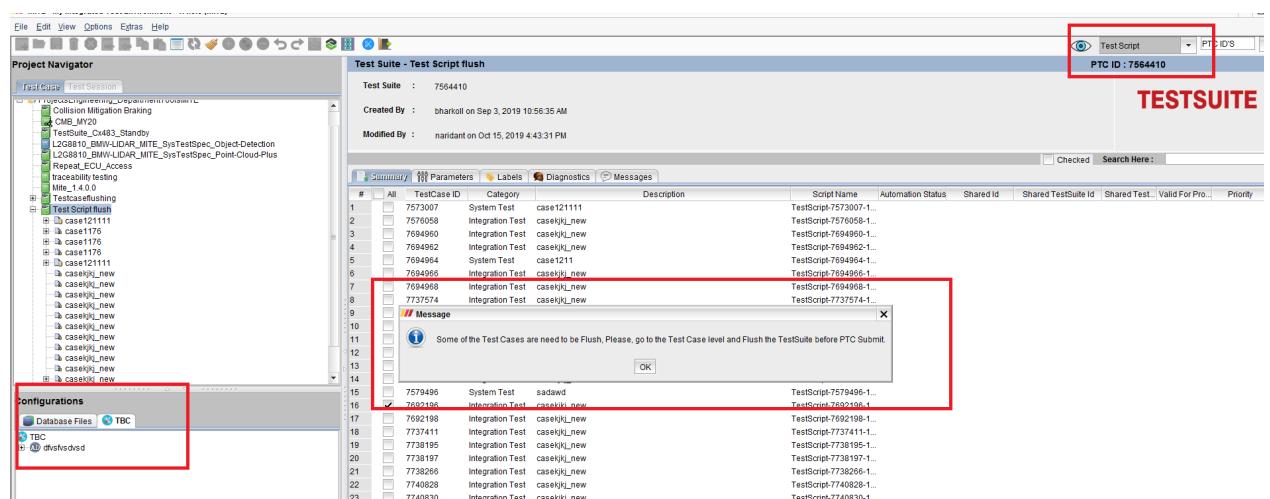
Step 2:-

User will get the flush icon button for Test Script level to submit it to PTC Integrity:



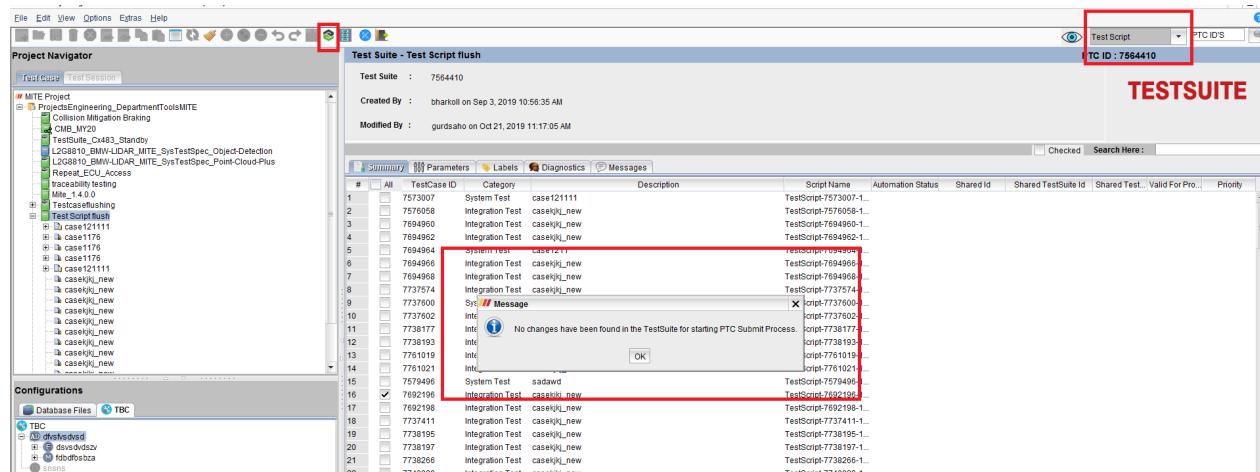
Step 3:-

When User try to flush the Test Suite in Script level, if some of the changes are there in Test case level then User will get the following pop up as shown below:



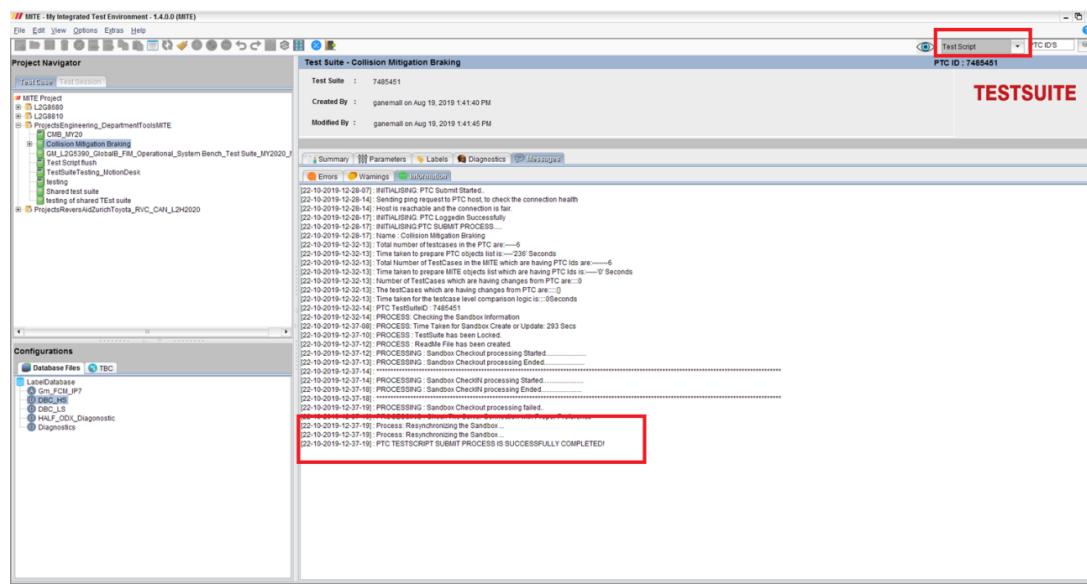
Step 4:-

If there are no changes to flush into PTC then following pop up will be shown:



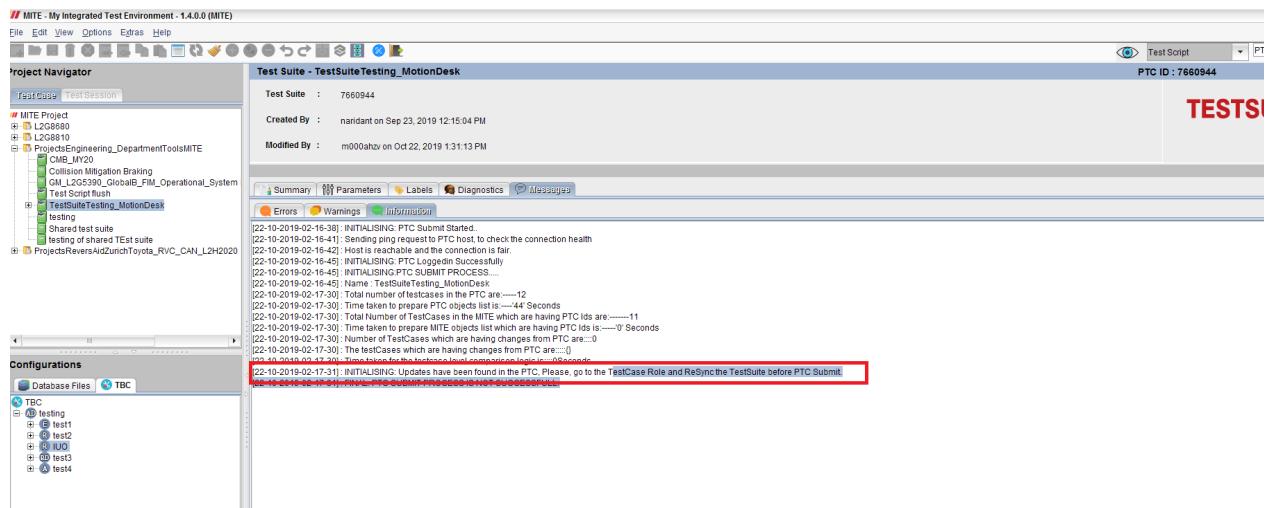
Step 5:-

If there are any changes then it will be flushed into PTC as shown below:



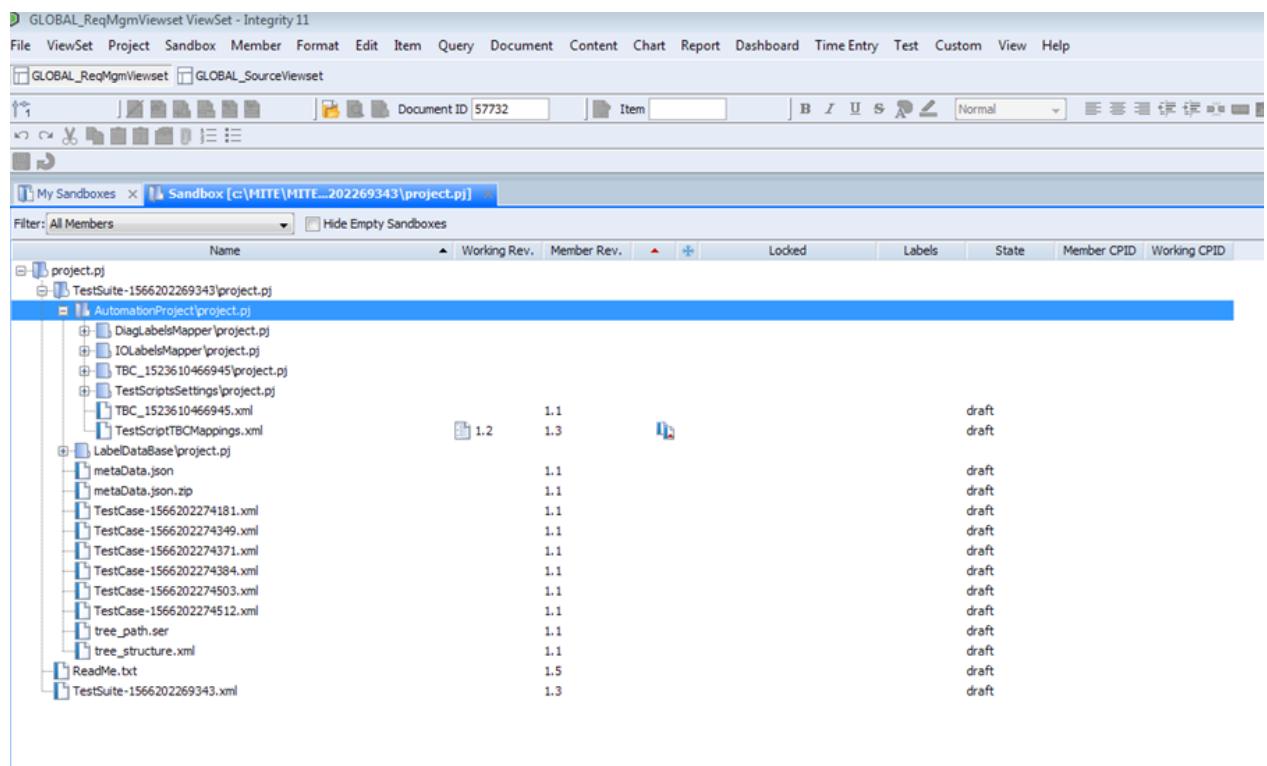
Step 6:-

If there is any updates in the Test Suite it will ask for Resync as shown below:



Step 7:-

After flushing user can check the sandbox as shown below:



Note:

To make the flush happen fastly and smoothly, try flushing the Test Suite from time to time.

Because,

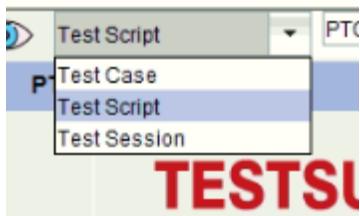
having less updates in the Test Suite will reduce the load on the Test Suite.

The decent number of Test Cases that can be flushed in one go are 800 - 1000 Test Cases.

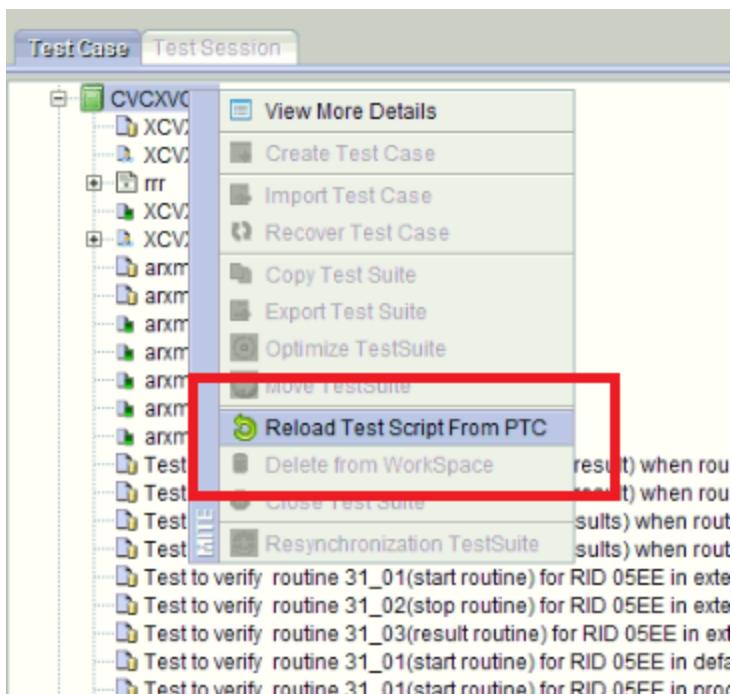
Created with the Personal Edition of HelpNDoc: [Free help authoring environment](#)

26.1 Reload test Script From PTC

When User will click on Test Script role as shown below:-



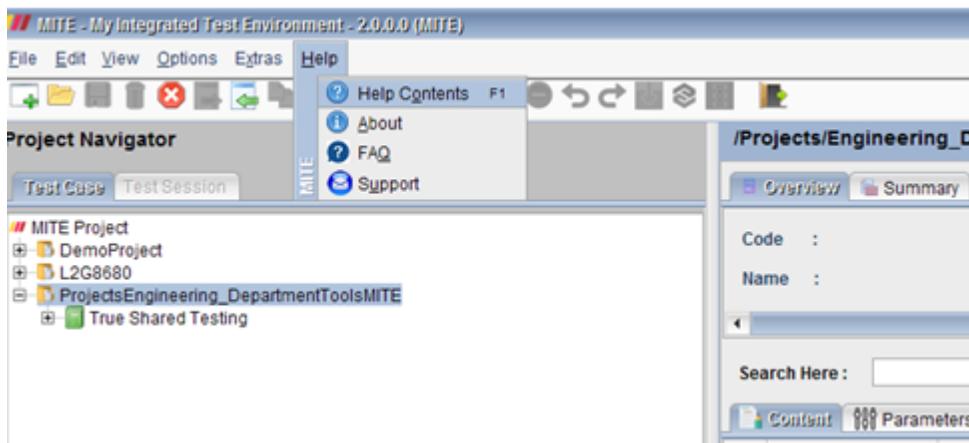
On clicking the right click on any test suite, User can get reload test script from PTC which will update the test suite automation project and PTC reload sandbox process will be done:-



Created with the Personal Edition of HelpNDoc: [Create HTML Help, DOC, PDF and print manuals from 1 single source](#)

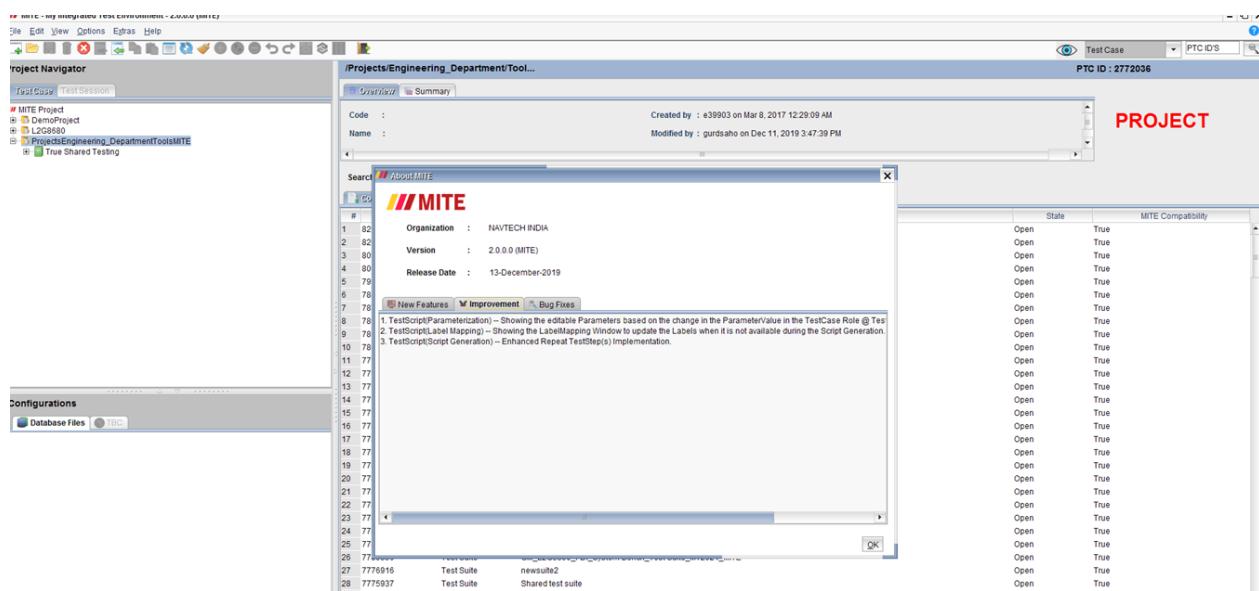
27 MITE SUPPORT

1. For MITE user manual go to HELP in Menu bar and select Help contents OR press F1



2. Go to Menu bar--> Help-->About.

You can get information of New features,Improvement,Bug fixes.



3. Go to Menu bar--> Help-->FAQ

You can find FAQ questions.

4. Go to Menu bar--> Help-->Support

You can get MITE support mail id **MITESUPPORT@NAVTECHCONSULTING.COM**

Created with the Personal Edition of HelpNDoc: [Qt Help documentation made easy](#)