

# TMF FORUM & ONAP COLLABORATION ON standardizing vnf test api specification



23rd Sep 2019 @ ONS EU

[edward.pershwitz@iteatechnologies.com](mailto:edward.pershwitz@iteatechnologies.com)

[frank.massoudian@iteatechnologies.com](mailto:frank.massoudian@iteatechnologies.com)

[michel.valette@orange.com](mailto:michel.valette@orange.com)

[Kanagaraj.Manickam@Huawei.com](mailto:Kanagaraj.Manickam@Huawei.com)

[pgauthier@tmforum.org](mailto:pgauthier@tmforum.org)



**tmforum**

# TMF model vs VTP model

## TMF

<https://app.swaggerhub.com/search?owner=fmassoudian>

### Test-Result-Management

This API provides a mechanism to manage test results and r

API

### Test-Execution

This API provides a contract for test case execution. Please s

API

OAS3

### Test-Case-Management

This API provides a mechanism to manage test case models

API

### Test-Environment-Management

This API provides a mechanism to manage test environment

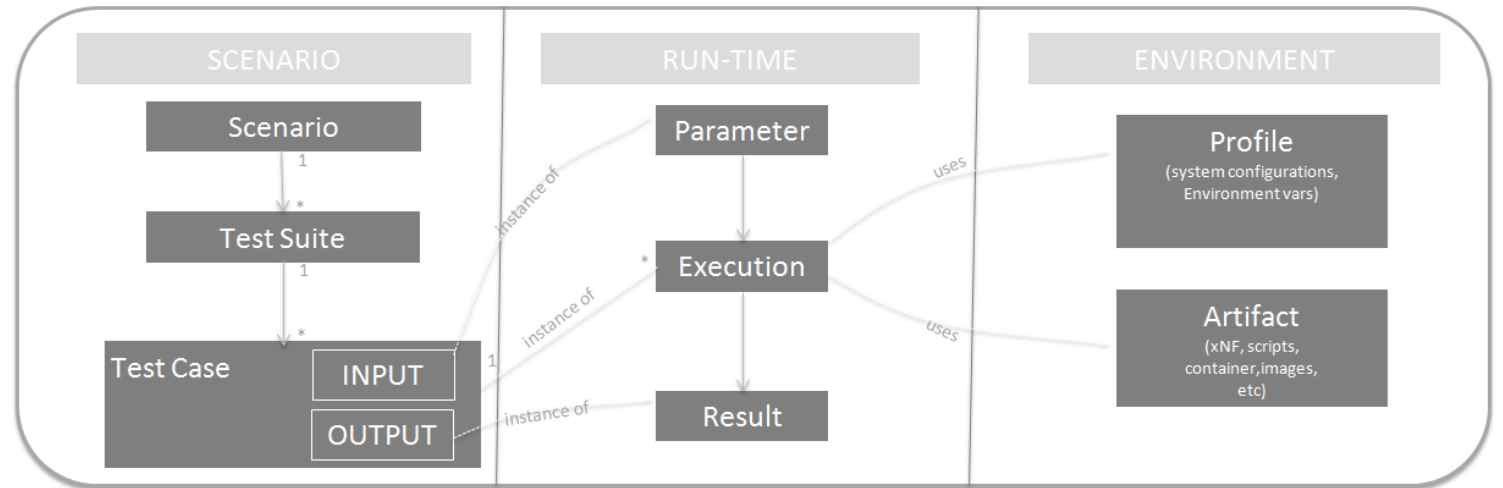
API

### Test-Data-Management

This API provides a mechanism to manage test data models

API

## VTP



## API Feature Mapping

TMF	VTP
Test case management	Scenario Management
Test data management Test Environment management	Environment management
Test execution Test Result management	Execution management

# TMF vs VTP Model mapping

TMF Feature	TMF Model	VTP Model	Notes
Test case management	Test Case	Test Case	1-1 mapping
	Test Suite	Test Suite	1-1 mapping
	High Level Function	Test Case	1-1 mapping VTP can model the high level function as another test case model itself Not exactly, used across multiple test caese
	Test Case Segment	Test Case	1-1 mapping VTP can model the test case segment as another test case model itself
	Test Flow	Test Suite	1-1 mapping VTP can model the test flow as another test case model itself (Test scripts with flows)
Test data management	Test suite	Scenario	It could be mapped to Test suite in TMF
	Test data instance	Profile	1-1 mapping Profile in VTP models the specific test case parameter with value {key: value}
	Test data schema	Parameter	1-1 mapping Test Parameter in VTP models the test case specific data.
	Test resources	Artifact	1-1 mapping
Test environment management	Environment meta-model	NIL	Domain knowledge of concrete resources,
	Environment instance	profile	1-1 mapping
	Resource api	NIL	Look into how to fit for ONAP
Test Execution management	Execution	Execution	1-1 mapping
	Run-time environment	Profile	1-1 mapping
Test result management	Test result	Test result	1-1 mapping

# TMF API vs VTP API mapping

TMF API	VTP API	Notes
<b>Test scenario management</b>	Scenario management	0-1
Add scenario management inline with VTP	<b>GET/vtp/scenarios</b> <b>POST/vtp/scenarios</b> <b>GET/vtp/scenarios/{scenario}</b> <b>PUT/vtp/scenarios/{scenario}</b> <b>DELETE/vtp/scenarios/{scenario}</b>	<b>Scenario brought into TMF from VTP</b>
Test suite management	Test Suite management	1-1
<b>GET/test-suite-models</b> <b>POST/test-suite-models</b> <b>GET/test-suite-models/{modelId}</b> <b>PUT/test-suite-models/{modelId}</b> <b>DELETE/test-suite-models/{modelId}</b>	<b>GET/vtp/scenarios/{scenario}/testsuites</b> <b>POST/vtp/scenarios/{scenario}/testsuites</b> <b>GET/vtp/scenarios/{scenario}/testsuites/{testsuite}</b> <b>PUT/vtp/scenarios/{scenario}/testsuites/{testsuite}</b> <b>DELETE/vtp/scenarios/{scenario}/testsuites/{testsuite}</b>	
Test case management	Test case management	1-1
<b>GET/test-case-models</b> <b>POST/test-case-models</b> <b>GET/test-case-models/{modelId}</b> <b>PUT/test-case-models/{modelId}</b> <b>DELETE/test-case-models/{modelId}</b>	<b>PUT/vtp/testcases</b> <b>GET/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases</b> <b>GET/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases/{testcase}</b> <b>DELETE/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases/{testcase}</b> <b>GET/vtp/scenarios/{scenario}/testcases</b>	
Test segment management	Test case management	1-1'
<b>GET/test-segment-models</b> <b>POST/test-segment-models</b> <b>GET/test-segment-models/{modelId}</b> <b>PUT/test-segment-models/{modelId}</b> <b>DELETE/test-segment-models/{modelId}</b>	<b>PUT/vtp/testcases</b> <b>GET/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases</b> <b>GET/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases/{testcase}</b> <b>DELETE/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases/{testcase}</b> <b>GET/vtp/scenarios/{scenario}/testcases</b>	Test segment could be modeled using test case model in VTP

# TMF API vs VTP API mapping

TMF API	VTP API	Notes
High level functions management	Test case management	1-1'
GET/high-level-functions POST/high-level-functions GET/high-level-functions/{modelId} PUT/high-level-functions/{modelId} DELETE/high-level-functions/{modelId}	PUT/vtp/testcases GET/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases GET/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases/{testcase} DELETE/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases/{testcase} GET/vtp/scenarios/{scenario}/testcases	high level functional test model could be modeled as Test case in VTP
Non-functional management	Test case management	1-1'
GET/non-functional-test-models POST/non-functional-test-models GET/non-functional-test-models/{modelId} PUT/non-functional-test-models/{modelId} DELETE/non-functional-test-models/{modelId}	PUT/vtp/testcases GET/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases GET/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases/{testcase} DELETE/vtp/scenarios/{scenario}/testsuites/{testsuite}/testcases/{testcase} GET/vtp/scenarios/{scenario}/testcases	Non-functional test model could be modeled as Test case in VTP
Test data instance management	Profile management	1-1'
GET/test-data-instances POST/test-data-instances GET/test-data-instances/{instanceId} PUT/test-data-instances/{instanceId} DELETE/test-data-instances/{instanceId}	GET/vtp/profiles POST/vtp/profiles GET/vtp/profiles/{profileName} PUT/vtp/profiles/{profileName} DELETE/vtp/profiles/{profileName}	
Test data schema management	<b>Test data schema management</b>	1-0
GET/test-data-schemas POST/test-data-schemas GET/test-data-schemas/{schemaId} PUT/test-data-schemas/{schemaId} DELETE/test-data-schemas/{schemaId}	Add Test data schema management inline with TMF	<b>Test data schema brought into VTP from TMF</b>

# TMF API vs VTP API mapping

TMF API	VTP API	Notes
Environment meta-model management	<a href="#">Environment meta-model management</a>	1-0
GET/environment-meta-model POST/environment-meta-model GET/environment-meta-model/{modelId} PUT/environment-meta-model/{modelId} DELETE/environment-meta-model/{modelId}	Add environment meta-model inline with TMF	<b>Environment brought into VTP from TMF</b>
Environment instance management	Profile management	1-1'
GET/environment-instances POST/environment-instances GET/environment-instances/{modelId} PUT/environment-instances/{modelId} DELETE/environment-instances/{modelId}	GET/vtp/profiles POST/vtp/profiles GET/vtp/profiles/{profileName} PUT/vtp/profiles/{profileName} DELETE/vtp/profiles/{profileName}	Environment details are modeled as profile in VTP.
Resource api		1-0
GET/resource-apis POST/resource-apis GET/resource-apis/{apild} PUT/resource-apis/{apild} DELETE/resource-apis/{apild}		Investigating on how to leverage it VTP
<a href="#">Provisioning artifact management</a>	Artifact management	0-1
Add artifact management inline with TMF	GET/vtp/artifacts POST/vtp/artifacts GET/vtp/artifacts/{artifactCategory}/{artifactName} PUT/vtp/artifacts/{artifactCategory}/{artifactName} DELETE/vtp/artifacts/{artifactCategory}/{artifactName}	<b>Artifact brought into TMF from VTP</b>

# TMF API vs VTP API mapping

TMF API	VTP API	Notes
Execution management	Execution management	1-1
<b>POST/test-execution</b> <b>GET/test-execution/{executionId}</b> <b>DELETE/test-execution/{executionId}</b>	<b>GET/vtp/executions/{executionId}</b> <b>GET/vtp/executions</b> <b>POST/vtp/executions</b>	
Runtime test environment	Profile management	1-1'
<b>POST/runtime-test-environment</b> <b>GET/runtime-test-environment/{environmentId}</b> <b>DELETE/runtime-test-environment/{environmentId}</b>	<b>GET/vtp/profiles</b> <b>POST/vtp/profiles</b> <b>GET/vtp/profiles/{profileName}</b> <b>PUT/vtp/profiles/{profileName}</b> <b>DELETE/vtp/profiles/{profileName}</b>	VTP provides profile which could be used to model runtime test environment as well
Result management	Execution management	1-1'
<b>GET/test-results</b> <b>POST/test-results</b> <b>GET/test-results/{id}</b> <b>PUT/test-results/{id}</b> <b>DELETE/test-results/{id}</b>	<b>GET/vtp/executions/{executionId}</b> <b>GET/vtp/executions</b>	VTP models the execution along with results. It would be add values to VTP if VTP could add additional capability to provide TMF result management by enabling searching ability

TOTAL resources in TMF : 13

TOTAL resources in VTP : 6

- 0-1 (TMF could leverage from VTP) : 2 no.s – **Scenario, Artifact**
- 1-0 (VTP could leverage from TMF) : 2 no.s – **Environment metadata, test data schema**
- 1-1 (directly mapped between VTP and TMF) : 3 no.s – Test suite, Test case, executions are
- 1-1' (Equivalent available in VTP for TMF) : 6 no.s

- Test Segment, High level functional, non-functional from TMF are mapped with Test case from VTP
- Test data, environment and run-time environment from TMF are mapped to profile from VTP.

## VTP and TMF meeting Outcome: Summary

ONAP VTP and TMF forum teams had discussed on synergizing the API between TMF JAD test REST API spec and ONAP VNF test platform (VTP) REST API spec. Based on the discussion, following conclusions are made:

- TMF leverage following features from VTP:  
**Scenario, Artifact**
- VTP leverage following features from TMF:  
**Environment metadata, test data schema**

Following features are directly mapped between TMF and VTP as it is:  
Test suite, Test case, executions

There are Equivalent feature available between VTP for TMF as given below:

TMF features (Test Segment, High level functional, non-functional) are mapped with Test case in VTP

TMF features ( Test data, environment and run-time environment ) are mapped to profile in VTP.



# TMF enhancements with Scenario and Artifact from VTP

The screenshot shows the SwaggerHub interface with a search bar containing 'owner:fmassoudian'. The search results are sorted by 'Recently updated'. Five API entries are visible, each with a title, description, and tags. The first four entries are: 'Test-Result-Management', 'Test-Execution', 'Test-Case-Management', and 'Test-Environment-Management'. The fifth entry, 'Test-Data-Management', is partially obscured by a red-bordered box. This box contains two entries: 'Test Scenario managment' and 'Test Artifact managment', both with descriptive text. The bottom of the search results shows 'SHOWING 1-5 OF 5'.

Owner	API Name	Description	Tags
fmassoudian	Test-Result-Management	This API provides a mechanism to manage test results and related artifacts. Please see description for individual tags for more detail on the ma...	API
fmassoudian	Test-Execution	This API provides a contract for test case execution. Please see description for individual tags for more detail on the managed artifacts.	API, OAS3
fmassoudian	Test-Case-Management	This API provides a mechanism to manage test case models and related artifacts. Please see description for individual tags for more detail on t...	API
fmassoudian	Test-Environment-Management	This API provides a mechanism to manage test environment models and related artifacts. Please see description for individual tags for more d...	API
fmassoudian	Test-Data-Management	This API provides a mechanism to manage test data models and related artifacts. Please see description for individual tags for more detail on t...	API
		<b>Test Scenario managment</b> <i>This API provides the mechanism to manage the different testing scenario like OVP, 5G, etc</i>	
		<b>Test Artifact managment</b> <i>This API provides the mechanims to manage the different artifact such as VNF image, scripts, etc</i>	



# Final: TMF API SPEC merged with VTP API SPEC

<https://app.swaggerhub.com/search?owner=fmassoudian>

fmassoudian

PUBLIC | UNPUBLISHED

Test-Result-Management

This API provides a mechanism to manage test results and related artifacts. Please see description for in...

API

fmassoudian

PUBLIC | UNPUBLISHED

Test-Execution

This API provides a contract for test case execution. Please see description for individual tags for more d...

API

OAS3

fmassoudian

PUBLIC | UNPUBLISHED

Test-Case-Management

This API provides a mechanism to manage test case models and related artifacts. Please see description ...

API

fmassoudian

PUBLIC | UNPUBLISHED

Test-Environment-Management

This API provides a mechanism to manage test environment models and related artifacts. Please see des...

API

fmassoudian

PUBLIC | UNPUBLISHED

Test-Data-Management

This API provides a mechanism to manage test data models and related artifacts. Please see description ...

API

mrkanag

PUBLIC | PUBLISHED

VNF-Test-Platform-Artifact

VNF-Test-Platform-Artifact - [1.1.0]

API

<https://app.swaggerhub.com/search?owner=mrkanag>

mrkanag

PUBLIC | PUBLISHED

VNF-Test-Platform-Scenario

VNF-Test-Platform-Scenario - [1.1.0]

API



Thank you

# Artifact analysis

Need of Artifact at different place

**Test case management**

```
GET/test-case-models
POST/test-case-models
GET/test-case-models/{modelId}
PUT/test-case-models/{modelId}
DELETE/test-case-models/{modelId}
```

**Environment instance management**

```
GET/environment-instances
POST/environment-instances
GET/environment-instances/{modelId}
PUT/environment-instances/{modelId}
DELETE/environment-instances/{modelId}
```

**Run-time artifact**

**USER NEEDS TO PROVIDE THE BINARIES LIKE VNF, IMAGES, ETC WHILE RUNNING TEST CASE**

POST /test-case-models  
While user deploys the testcases, he/she would post the test case  
Scripts, jars as part of POST request as below  
{test-case-model-json + **testcase binaries**}

POST /environment-instances  
While user setup environment such as ONAP, he/she would post the environment  
With required installer binaries as part of POST request as below  
{environment-instance-json + **environment binaries**}

When user executes the test cases, he/she would want to pass **the run-time binaries** such as VNF, App, etc. For this one, there could be different category of artifacts depends on the test case needs.

The need for binaries management is there across test-case mgmt, environment mgmt and run-time execution mgmt. To handle all these needs, VTP is providing Artifact management with **Categorized** which is required for organizing the binaries, something similar to file system **directory**, while **binary** could be considered as **file**. So its better to keep the artifact mgmt as separate group, and use it across test-case mgmt, environment mgmt and run-time mgmt.

Solution

In POST /test-case-models  
Instead of {test-case-model-json + **testcase binaries**}

↓

Use {test-case-model-json + **testcase binary URI**  
(/artifact/**testcase**/tc-jar1)}

Pre-defined Category: **testcase**

In POST /environment-instances  
Instead of {environment-instance-json + **environment binaries**}

↓

Use {environment-instance-json + **environment binary URI**  
(/artifact/**environment**/env-onap-1)}

Pre-defined Category: **environment**

**Artifact management**

```
GET/vtp/artifacts
POST/vtp/artifacts
GET/vtp/artifacts/{artifactCategory}/{artifactName}
PUT/vtp/artifacts/{artifactCategory}/{artifactName}
DELETE/vtp/artifacts/{artifactCategory}/{artifactName}
```