



**MEF Standard**

**MEF 136**

**Allegro, Interlude and Legato Service Function Testing  
Business Requirements & Use Cases**

**April 2023**

## Disclaimer

© MEF Forum 2023. All Rights Reserved.

The information in this publication is freely available for reproduction and use by any recipient and is believed to be accurate as of its publication date. Such information is subject to change without notice and MEF Forum (MEF) is not responsible for any errors. MEF does not assume responsibility to update or correct any information in this publication. No representation or warranty, expressed or implied, is made by MEF concerning the completeness, accuracy, or applicability of any information contained herein and no liability of any kind shall be assumed by MEF as a result of reliance upon such information.

The information contained herein is intended to be used without modification by the recipient or user of this document. MEF is not responsible or liable for any modifications to this document made by any other party.

The receipt or any use of this document or its contents does not in any way create, by implication or otherwise:

- a) any express or implied license or right to or under any patent, copyright, trademark or trade secret rights held or claimed by any MEF member which are or may be associated with the ideas, techniques, concepts or expressions contained herein; nor
- b) any warranty or representation that any MEF members will announce any product(s) and/or service(s) related thereto, or if such announcements are made, that such announced product(s) and/or service(s) embody any or all of the ideas, technologies, or concepts contained herein; nor
- c) any form of relationship between any MEF member and the recipient or user of this document.

Implementation or use of specific MEF standards, specifications, or recommendations will be voluntary, and no Member shall be obliged to implement them by virtue of participation in MEF Forum. MEF is a non-profit international organization to enable the development and worldwide adoption of agile, assured and orchestrated network services. MEF does not, expressly or otherwise, endorse or promote any specific products or services.

## Table of Contents

<b>1</b>	<b>List of Contributing Members</b> .....	<b>1</b>
<b>2</b>	<b>Abstract</b> .....	<b>1</b>
<b>3</b>	<b>Terminology and Abbreviations</b> .....	<b>2</b>
<b>4</b>	<b>Compliance Levels</b> .....	<b>3</b>
<b>5</b>	<b>Introduction</b> .....	<b>4</b>
<b>6</b>	<b>Terminology Cross Reference</b> .....	<b>5</b>
<b>7</b>	<b>Use Cases</b> .....	<b>6</b>
7.1	Service Test Specification .....	8
7.1.1	Create Service Test Specification .....	8
7.1.2	Modify Service Test Specification.....	11
7.1.3	Delete Service Test Specification .....	12
7.1.4	Retrieve Service Test Specification List.....	13
7.1.5	Retrieve Service Test Specification by Identifier .....	13
7.1.6	Bundled Service Test Specifications .....	14
7.2	Service Test .....	15
7.2.1	Create Service Test .....	15
7.2.2	Create Service Test w/o Service Test Specification .....	17
7.2.3	Suspend Service Test .....	19
7.2.4	Resume Service Test.....	19
7.2.5	Modify Service Test.....	20
7.2.6	Modify Service Test w/o Test Specification.....	21
7.2.7	Delete Service Test .....	22
7.2.8	Retrieve Service Test List.....	23
7.2.9	Retrieve Service Test by Identifier .....	23
7.3	Notifications .....	24
7.3.1	Register for Notifications.....	24
7.3.2	Unregister for Notifications .....	26
7.3.3	Generate Notification.....	26
7.4	Results .....	27
<b>8</b>	<b>Process Flows</b> .....	<b>28</b>
8.1	Create Service Test Specification .....	28
8.2	Delete Service Test Specification .....	29
8.3	Modify Service Test Specification .....	31
8.4	Create Service Test Process Flow .....	33
8.5	Delete Service Test Process Flow .....	36
8.6	Modify Service Test Process Flow .....	38
8.7	Suspend Service Test Process Flow .....	40
8.8	Resume Service Test Process Flow .....	42
<b>9</b>	<b>References</b> .....	<b>45</b>
<b>Appendix A</b>	<b>Acknowledgements</b> .....	<b>46</b>
<b>Appendix B</b>	<b>Test Access Provisioning</b> .....	<b>47</b>



B.1 Test Resources ..... 47  
B.2 Test Access Configuration..... 48

## List of Figures

Figure 1 – High Level Create Service Test Specification Sequence Diagram .....	11
Figure 2 – Create Service Test Specification Process Flow .....	28
Figure 3 – Delete Service Test Specification Process Flow .....	30
Figure 4 – Modify Service Test Specification Process Flow .....	32
Figure 5 – Create Service Test Process Flow .....	34
Figure 6 – Delete Service Test Process Flow .....	37
Figure 7 – Modify Service Test Process Flow .....	39
Figure 8 – Suspend Service Test Process Flow .....	41
Figure 9 – Resume Service Test Process Flow .....	43
Figure 10 – Imbedded and External SFMP, GTF, and CTF .....	47
Figure 11 – Test Access Sequence Diagram.....	49

## List of Tables

Table 1 – Terminology.....	2
Table 2 – Abbreviations.....	2
Table 3 – Terminology Cross Reference .....	5
Table 4 – Use Case Overview.....	7
Table 5 – Service Test Specification Attributes .....	9
Table 6 – Service Test Specification Relationship Attributes .....	10
Table 7 – Related Party Attributes.....	10
Table 8 – Service Test Attributes.....	17
Table 9 – Service Test Attributes.....	18
Table 10 – Register for Notification Attributes .....	25
Table 11 – Notification Attributes .....	27
Table 12 – Create Service Test Specification States .....	29
Table 13 – Delete Service Test Specification States .....	31
Table 14 – Modify Service Test Specification States .....	33
Table 15 – Create Service Test States.....	36
Table 16 – Delete Service Test States.....	38
Table 17 – Modify Service Test States .....	40
Table 18 – Suspend Service Test States .....	42
Table 19 – Resume Service Test States .....	44

## 1 List of Contributing Members

The following members of the MEF participated in the development of this document and have requested to be included in this list.

- Lumen Technologies
- Spirent

## 2 Abstract

This document defines the Business Requirements and Use Cases to support Service Function Testing (SFT) at the Allegro, Interlude, and Legato Interface Reference Points. The requirements and use cases contained in this document support Service Function Testing for Service Activation and Service Testing performed at other times during the lifecycle of the service.

### 3 Terminology and Abbreviations

This section defines the terms used in this document. In many cases, the normative definitions to terms are found in other documents. In these cases, the third column is used to provide the reference that is controlling, in other MEF or external documents.

In addition, terms defined in MEF 55.1 [3] are included in this document by reference and are not repeated in the table below.

<b>Term</b>	<b>Definition</b>	<b>Reference</b>
<b>Bundled</b>	Two or more Service Test Specifications are related together in a bundle and are given an order in which they are run.	This document
<b>Dependency</b>	The related Service Test Specification is dependent on the success or failure of another Service Test Specification.	This document
<b>Exclusivity</b>	The related Service Test Specification requires that it be paired with a second Service Test Specification.	This document
<b>Migration</b>	The related Service Test Specification is being migrated to replace the Service Test Specification.	This document
<b>Service Function Testing</b>	The verification of the operation or definition of the Service Under Test. Includes Service Activation and testing performed on in-service Services for maintenance purposes.	This document
<b>Service Identifier</b>	The unique identifier for a specific Service.	This document
<b>Service Specification</b>	The specification of a set of attributes that define a Service type.	This document
<b>Service Test</b>	A definition of SFT for a specific Service Identifier.	This document
<b>Service Test Specification</b>	Detailed specification that includes the Service Test attributes and Service Specifications that are specified to be tested by this Service Test Specification.	This document
<b>Substitution</b>	The related Service Test Specification is substituted for the Service Test Specification.	This document

**Table 1 – Terminology**

<b>Abbreviation</b>	<b>Definition</b>	<b>Reference</b>
<b>IRP</b>	Interface Reference Point	MEF 55.1 [3]
<b>SFT</b>	Service Function Testing	This document

**Table 2 – Abbreviations**

## 4 Compliance Levels

The key words "**MUST**", "**MUST NOT**", "**REQUIRED**", "**SHALL**", "**SHALL NOT**", "**SHOULD**", "**SHOULD NOT**", "**RECOMMENDED**", "**NOT RECOMMENDED**", "**MAY**", and "**OPTIONAL**" in this document are to be interpreted as described in BCP 14 (RFC 2119 [1], RFC 8174 [2]) when, and only when, they appear in all capitals, as shown here. All key words must be in bold text.

Items that are **REQUIRED** (contain the words **MUST** or **MUST NOT**) are labeled as [**Rx**] for required. Items that are **RECOMMENDED** (contain the words **SHOULD** or **SHOULD NOT**) are labeled as [**Dx**] for desirable. Items that are **OPTIONAL** (contain the words **MAY** or **OPTIONAL**) are labeled as [**Ox**] for optional.

## 5 Introduction

This document defines the Business Requirements and Use Cases for Service Function Testing (SFT). Service Function Testing is defined as testing that is performed either as a part of service verification, service activation, or service maintenance. The requirements and Use Cases are the same for the Allegro Interface Reference Point (IRP), the Interlude IRP, and the Legato IRP. There are no differences identified within this document between them. The requirements and Use Cases within this document will be used to develop an API specification and Developer's Guide.

The Use Cases that are addressed in this document are:

- Create, Modify, Delete Service Test Specification
- Retrieve Service Test Specification List
- Retrieve Service Test Specification by ID
- Bundle Service Test Specifications
- Create, Modify, Delete, Suspend, Resume Service Test w/Service Test Specification
- Create, Modify, Delete, Suspend, Resume Service Test w/o Service Test Specification
- Retrieve Service Test List
- Retrieve Service Test by ID
- Subscribe to Notifications
- Unsubscribe to Notifications
- Generate Notifications

## 6 Terminology Cross Reference

This document uses terminology primarily from TMF653. Previous work in MEF used different terminology. To aid readers who are familiar with the previous work, Table 3 provides a cross reference between the MEF and TMF terms for the same thing.

<b>TMF Term</b>	<b>MEF Term</b>
Service Test Specification	Test Profile
Service Test	Test Job
Test Measurement	Test Results

**Table 3 – Terminology Cross Reference**

## 7 Use Cases

The Use Cases addressed within this document are shown in Table 4. Each Use Case is further detailed along with requirements associated with this Use Case.

Use Case Name	Use Case Description	Reference Section
Create Service Test Specification	The Buyer/Client creates a Service Test Specification that contains details of the measurement(s) to be performed.	Section 7.1.1
Modify Service Test Specification	The Buyer/Client modifies a Service Test Specification that contains details of the measurement(s) to be performed.	Section 7.1.2
Delete Service Test Specification	The Buyer/Client deletes a Service Test Specification that contains details of the measurement(s) to be performed.	Section 7.1.3
Retrieve Service Test Specification List	The Buyer/Client retrieves a list of Service Test Specifications that meet specified filter criteria.	Section 7.1.4
Retrieve Service Test Specification by Identifier	The Buyer/Client retrieves a specific Service Test Specification using the Service Test Specification Identifier.	Section 7.1.5
Bundled Service Test Specification	The Buyer/Client creates a bundle of Service Test Specifications with an indication of the order which they are run.	Section 7.1.6
Create Service Test w/Service Test Specification	The Buyer/Client creates a Service Test that specifies the service to be tested and references a Service Test Specification.	Section 7.2.1

Use Case Name	Use Case Description	Reference Section
Create Service Test w/o Service Test Specification	The Buyer/Client creates a Service Test that specifies the service to be tested and does not reference a Service Test Specification. Instead, attributes contained in the Service Test Specification are included in the Service Test.	Section 7.2.2
Suspend Service Test	The Buyer/Client suspends an In-Progress Service Test with or without a Service Test Specification.	Section 7.2.3
Resume Service Test	The Buyer/Client resumes a suspended Service Test with or without a Service Test Specification.	Section 7.2.4
Modify Service Test w/Service Test Specification	The Buyer/Client modifies a Service Test that specifies the service to be tested and references a Service Test Specification.	Section 7.2.5
Modify Service Test w/o Service Test Specification	The Buyer/Client modifies a Service Test that specifies the service to be tested and does not reference a Service Test Specification. Instead, the attributes contained in the Service Test Specification are contained in the Service Test.	Section 7.2.6
Delete Service Test	The Buyer/Client deletes a Service Test.	Section 7.2.7
Retrieve Service Test List	The Buyer/Client retrieves a list of Service Test that meet specified filter criteria.	Section 7.2.8
Retrieve Service Test by Identifier	The Buyer/Client retrieves a specific Service Test using the Service Test Identifier.	Section 7.2.9
Subscribe to Notifications	The Buyer/Client subscribes to SFT notifications.	Section 7.3.1
Unsubscribe to Notifications	The Buyer/Client unsubscribes to SFT notifications.	Section 7.3.2
Generate Notification	The Seller/Server generates SFT Notifications.	Section 7.3.3

**Table 4 – Use Case Overview**

## 7.1 Service Test Specification

The Use Cases and associated requirements related to the Service Test Specification are defined in this section.

### 7.1.1 Create Service Test Specification

1. The Buyer/Client can request that the Seller/Server create a Service Test Specification.

Attribute Name	Description	Value	Comments
Service Test Specification Identifier	A unique, within the Seller/Server, identifier for the Service Test Specification.	String	Set by Seller
Description	Text description of the Service Test Specification.	String	Set by Buyer
Creation Date	Date the Service Test Specification is created.	Datetime	Set by Seller
Is Bundled	Determines whether specification represents a single specification (false), or a bundle of specifications (true).	Boolean	Set by Buyer
Lifecycle Status	Used to indicate the current Lifecycle Status of this Service Test Specification.	One of: Experimental Pending Approved Deprecated	Set by Buyer
Service Test Specification Name	The name of the Service Test Specification.	String	Set by Buyer
Valid For	The period of time which this Service Test Specification is valid for.	Time Period	Set by Buyer
Version	Version of the Service Test Specification.	String	Set by Seller
Last Update	Last time the Service Test	Datetime	Set by Seller

<b>Attribute Name</b>	<b>Description</b>	<b>Value</b>	<b>Comments</b>
	Specification was modified by the Buyer.		
Service Test Specification Relationship	Related Service Specification.	See Table 6	Set by Buyer
Service Specific Test Specification Attributes	Service specific attributes that define how a test is performed for a given Test Specification.	Under Study	Set by Buyer
Service Offering Identifier	Relationship Service Offering Identifier.	String	Set by Buyer
Related Party	Parties who manage or have an interest in this Service Test Specification.	See Table 7	Set by Buyer
Target Entity Schema	Pointer to a schema that defines the target entity.	String	Set by Buyer

**Table 5 – Service Test Specification Attributes**

<b>Attribute Name</b>	<b>Description</b>	<b>Value</b>	<b>Comments</b>
Related Service Specification Identifier	The Identifier of the related Service Specification that this Service Test Specification can be used to test.	String	Set by Buyer
Service Test Specification Relationship Identifier	The unique identifier for a related Service Test Specification.	String	Set by Buyer
Service Test Specification Relationship Name	The unique name for a related Service Test Specification.	String	Set by Buyer
Service Test Specification Relationship Type	Type of relationship.	One of: Migration Substitution Dependency Exclusivity	Set by Buyer

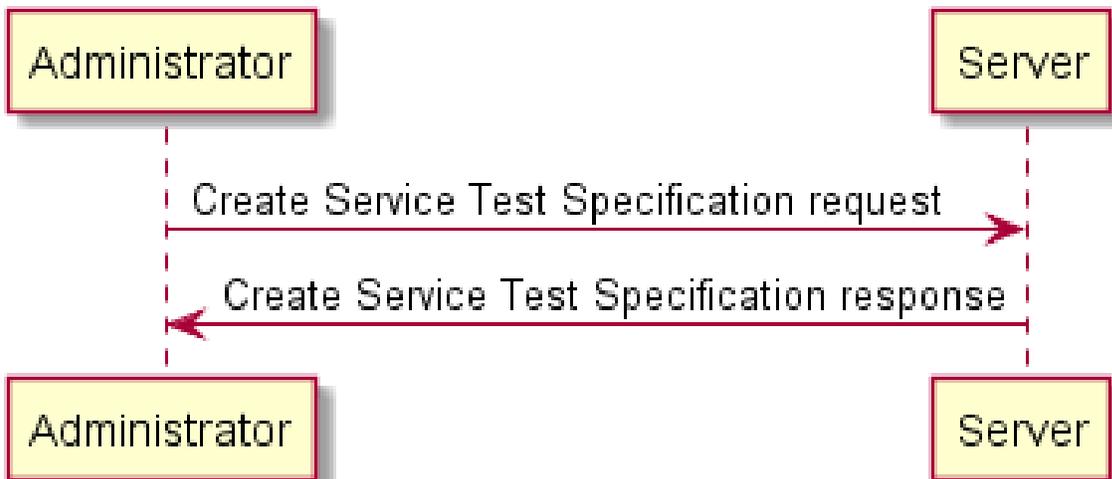
Attribute Name	Description	Value	Comments
		Bundled	
Service Test Specification Relationship Role	The role for this relationship.	One of: Primary Secondary	Set by Buyer
Service Test Specification Relationship Order	The order which the related Service Tests are run during a Service Test.	Numeric	Set by Buyer
Service Test Specification Relationship Valid For	The period for which this is valid.	Time Period	Set by Buyer

**Table 6 – Service Test Specification Relationship Attributes**

Attribute Name	Description	Value	Comments
Related Party Identifier	Unique to the Seller/Server identifier for the Related Party.	String	Set by Buyer
Related Party Name	Name of the Related Party.	String	Set by Buyer
Related Party Role	Role played by the Related Party.	String	Set by Buyer

**Table 7 – Related Party Attributes**

## Create Service Test Specification High Level



### Figure 1 – High Level Create Service Test Specification Sequence Diagram

[R1] The Buyer/Administrator's Create Service Test Specification request **MUST** include the following attributes:

- Service Test Specification Name
- Valid For
- Related Service Test Specification
- Target Entity Schema

[O1] The Buyer/Administrator's Create Service Test Specification request **MAY** include any other attributes shown as Set by Buyer/Administrator in Table 5.

2. The Seller/Server responds to the Buyer/Administrator's request.

[R2] If the request is successful, the Seller's response to a Create Service Test Specification request **MUST** echo back all Buyer/Administrator provided attributes.

[R3] If the request is successful, the Seller's response to a Create Service Test Specification request **MUST** include the following attributes:

- Service Test Specification Identifier
- Creation Date
- Last Modify Date

[R4] The Last Modify Date **MUST** be the same as the Creation Date.

[R5] If the request is not successful, the Seller/Server **MUST NOT** echo back all Buyer/Administrator provided attributes.

[R6] If the request is not successful, the Seller/Server **MUST NOT** return a Service Test Specification Identifier, Creation Date, or Last Modify Date.

[R7] If the Seller/Server encounters errors, they **MUST** return an error with explanation to the Buyer/Administrator.

#### 7.1.2 Modify Service Test Specification

1. The Buyer/Administrator can request modifications to an existing Service Test Specification.

[R8] The Buyer's Modify Service Test Specification request **MUST** include the following attributes:

- Service Test Specification Identifier
- [O2] The Buyer's Modify Service Test Specification request **MAY** include any other attributes shown as Set by Buyer/Administrator in Table 5.
2. The Seller/Server responds to the Buyer's request.
- [R9] The Seller's response to a Modify Service Test Specification request **MUST** echo back Buyer/Administrator provided attributes.
- [R10] The Seller's response to a Modify Service Test Specification request **MUST** include the following attributes:
- Service Test Specification Identifier
  - Creation Date
  - Last Modify Date
- [R11] The Last Modify Date **MUST** be set to the current date and time by the Seller.
- [R12] If the request is not successful, the Seller/Server **MUST NOT** echo back all Buyer/Administrator provided attributes.
- [R13] If the Seller/Server encounters errors, they **MUST** return an error with explanation to the Buyer/Administrator.

### 7.1.3 Delete Service Test Specification

1. The Buyer/Administrator can delete a Service Test Specification.
- [R14] The Buyer's Delete Service Test Specification request **MUST** contain the Service Test Specification Identifier.
- [R15] The Buyer's Delete Service Test Specification request **MUST NOT** contain any other attributes.
2. The Seller/Server responds to the Buyer's request.
- [R16] The Seller's response to a Delete Service Test Specification request **MUST** echo back the Buyer/Administrator provided attributes.
- [R17] If the request is not successful, the Seller/Server **MUST NOT** echo back all Buyer/Administrator provided attributes.
- [R18] If the Seller/Server encounters errors, they **MUST** return an error with explanation to the Buyer/Administrator.

#### 7.1.4 Retrieve Service Test Specification List

1. The Buyer/Administrator can retrieve a list of Service Test Specifications.

[O3] The Buyer's Retrieve Service Test Specification List request **MAY** contain none or more of the following attributes:

- Description
- Creation Date
- Last Update
- Related Party
- Related Service Specification Identifier
- Service Test Specification Relationship

2. The Seller/Server responds to the Buyer's request.

[R19] If the request is successful, the Seller's response to a Retrieve Service Test Specification List request **MUST** include the following attributes:

- Service Test Specification Identifier
- Description
- Creation Date
- Last Update

[R20] If the request is successful, the Seller/Server **MUST** echo back the criteria provided by the Buyer/Client.

[R21] If the request is successful but the Seller/Server finds no entries that match the filter criteria, they **MUST** return an empty list.

[R22] If the request is unsuccessful, the Seller/Server **MUST NOT** echo back the criteria provided by the Buyer/Client.

[R23] If the request is unsuccessful, the Seller/Server **MUST** return an error with explanation to the Buyer/Administrator.

#### 7.1.5 Retrieve Service Test Specification by Identifier

1. The Buyer/Client can retrieve a Service Test Specification by Service Test Specification Identifier.

[R24] The Buyer's Retrieve Service Test Specification by Service Test Specification Identifier **MUST** include the Service Test Specification Identifier.

[R25] The Buyer's Retrieve Service Test Specification by Service Test Specification identifier request **MUST NOT** include other attributes.

2. The Seller/Server responds to the Buyer's request.

[R26] If the request is successful, the Seller's response to a Retrieve Service Test Specification by Service Test Specification Identifier request **MUST** include the following attributes:

- Service Test Specification Identifier
- Description
- Creation Date
- Last Update
- Service Test Specification Name
- Valid For
- Version
- Related Service Specification Identifier
- Service Test Specification Relationship
- Entity Specification Relationship
- Related Party
- Target Entity Schema

[R27] If the request is unsuccessful, the Seller/Server **MUST** return an error with explanation to the Buyer/Client.

#### 7.1.6 Bundled Service Test Specifications

It is possible to show relationships between Service Test Specifications or to bundle them together to create a Bundled Service Test Specification. When bundling Service Test Specifications together, the Is Bundled attribute is set to True and the Service Test Relationship Type is set to Bundled. One Service Test Specification is identified as the primary Service Test specification and one or more Service Test Specifications are identified as secondary. The Service Test Specification Relationship Role of the primary Service Test Specification is set to Primary, and all

related Service Test Specifications are set to Secondary. By retrieving the Primary Service Test Specification by Identifier, the related Service Test Specifications can be identified.

- [R28] The Buyer/Client's Create Service Test Specification request **MUST** have the Is Bundled attribute set to True.
- [R29] The Buyer/Client's Create Service Test Specification request for Service Test Specifications that are bundled together **MUST** set the Service Test Relationship Type to Bundled.
- [R30] The Buyer/Client's Create Service Test Specification request for Service Test Specifications that are bundled together **MUST** set the Service Test Relationship Type of the primary Service Test specification to Primary.
- [R31] The Buyer/Client's Create Service Test Specification request for Service Test Specifications that are bundled together **MUST** set the Service Test Relationship Type of the related Service Test Specifications to Secondary.
- [R32] The Buyer/Client's Create Service Test Specification request for Service Test Specifications that are bundled together **MUST** identify the Service Test Specification Relationship Order as a numeral from 1-n with 1 being the first Service Test Specification run and n being the last Service Test Specification run. .
- [R33] If a Buyer performs a Retrieve Service Test Specification by Identifier and that Service Test Specification is bundled, the Seller **MUST** respond with all of the related Service Test Specifications and their roles.

## **7.2 Service Test**

The Use Cases and associated requirements related to the Service Test are defined in this section.

### **7.2.1 Create Service Test**

1. The Buyer/Client is able to request that the Seller/Server create a Service Test.
  - [R34] The Buyer's Create Service Test request **MUST** contain the following attributes:
    - Service Test Name
    - Start Date Time
    - End Date Time
    - Valid For
    - Service Test Specification Identifier

- Related Service Identifier
- Test Attribute Schema

Attribute Name	Description	Value	Comments
Service Test Identifier	The identifier of the Service Test.	String	Set by Seller
Service Test Name	The name of the Service Test.	String	Set by Buyer
Service Test Description	A description of the Service Test.	String	Set by Buyer
Start Date Time	The start date and time of the Service Test.	Date/Time	Set by Buyer  If the Buyer desires to schedule a test, this is a future date/time. If the Buyer desires an immediate test, this is the current date/time.
Actual Start Date Time	The actual time that a Service Test started.	Date/Time	Set by Seller/Server
End Date Time	The end date and time of the Service Test.	Date/Time	Set by Buyer
Actual End Date Time	The Actual end date and time of the Service Test.	Date/Time	Set by Seller/Server
Valid For	The period of time which the Service Test is valid for.	Time Period	Set by Buyer
Service Test Specification Identifier	The identifier of the Service Test Specification.	String	Set by Buyer
Service Test Specification Version	The version of the Service Test Specification.	String	Set by Buyer
Related Service Identifier	The Service Identifier for the service to be tested.	String	Set by Buyer
Related Service Name	The name of the service to be tested.	String	Set by Buyer
Test Attribute Schema	The schema to be used that defines	String	Set by Buyer

Attribute Name	Description	Value	Comments
	the Test Measure attributes.		

**Table 8 – Service Test Attributes**

2. The Seller responds to the Buyer’s request.

**[R35]** If the request is successful, the Seller’s response to a Create Service Test request **MUST** echo back all Buyer/Client provided attributes.

**[R36]** If the request is successful, the Seller **MUST** return the following attributes:

- Service Test Identifier
- Service Test State

**[R37]** If the request is not successful, the Seller/Server **MUST NOT** echo back all Buyer/Client provided attributes.

**[R38]** If the request is not successful, the Seller/Server **MUST NOT** return a Service Test Identifier or Service Test State.

**[R39]** If the Seller/Server encounters errors, they **MUST** return an error with explanation to the Buyer/Client

**7.2.2 Create Service Test w/o Service Test Specification**

1. The Buyer/Client is able to request that the Seller/Server create a Service Test without referencing a Service Test Specification.

**[R40]** The Buyer’s Create Service Test request **MUST** contain the following attributes:

- Service Test Name
- Start Date Time
- End Date Time
- Valid For
- Related Service Identifier
- Test Attribute Schema

Attribute Name	Description	Value	Comments
Service Test Identifier	The identifier of the Service Test.	String	Set by Seller
Service Test Name	The name of the Service Test.	String	Set by Buyer
Service Test Description	A description of the Service Test.	String	Set by Buyer
Start Date Time	The start date and time of the Service Test.	Date/Time	Set by Buyer  If the Buyer desires to schedule a test, this is a future date/time. If the Buyer desires an immediate test, this is the current date/time.
Actual Start Date Time	The actual time that a Service Test started.	Date/Time	Set by Seller/Server
End Date Time	The end date and time of the Service Test.	Date/Time	Set by Buyer
Actual End Date Time	The Actual end date and time of the Service Test.	Date/Time	Set by Seller/Server
Valid For	The period of time which the Service Test is valid for.	Time Period	Set by Buyer
Related Service Identifier	The Service Identifier for the service to be tested.	String	Set by Buyer
Related Service Name	The name of the service to be tested.	String	Set by Buyer
Test Attribute Schema	The schema to be used that defines the Test Measure attributes.	String	Set by Buyer

**Table 9 – Service Test Attributes**

2. The Seller responds to the Buyer’s request.

**[R41]** If the request is successful, the Seller’s response to a Create Service Test request **MUST** echo back all Buyer/Client provided attributes.

**[R42]** If the request is successful, the Seller **MUST** return the following attributes:

- Service Test Identifier
- Service Test State

[R43] If the request is not successful, the Seller/Server **MUST NOT** echo back all Buyer/Client provided attributes.

[R44] If the request is not successful, the Seller/Server **MUST NOT** return a Service Test Identifier or Service Test State.

[R45] If the Seller/Server encounters errors, they **MUST** return an error with explanation to the Buyer/Client

### 7.2.3 Suspend Service Test

1. The Buyer/Client submits a Suspend Service Test request to suspend an existing Service Test.

[R46] The Buyer/Client's Suspend Service Test request **MUST** include the Service Test Identifier.

[R47] The Service Test **MUST** be in the IN\_PROGRESS state.

2. The Seller/Server responds to the Buyer/Client's request by changing the state of the Service Test to SUSPENDED.

[R48] If the request is successful, the Seller/Server **MUST** suspend all testing and measurements being performed on the Service Test and place the Service Test in the SUSPENDED state when they receive a Suspend Service Test request from the Buyer/Client.

[R49] While in the SUSPENDED state, the Service Test **MUST NOT** perform any testing or measurements.

[R50] If the request is not successful, the Seller/Server **MUST NOT** suspend the Service Test.

[R51] If the Seller/Server encounters errors, they **MUST** return an error with explanation to the Buyer/Client

3. The Service Test tests and measurements are suspended.

### 7.2.4 Resume Service Test

1. The Buyer/Client submits a Resume Service Test request to resume an existing Service Test.

[R52] The Buyer/Client's Resume Service Test request **MUST** include the Service Test Identifier.

- [R53] The Service Test **MUST** be in the SUSPENDED state.
2. The Seller/Server responds to the Buyer/Client's request by changing the state of the Service Test to IN\_PROGRESS.
- [R54] If the request is successful, the Seller/Server **MUST** resume all testing and measurement being performed on the Service Test and place the Service Test in the IN\_PROGRESS state when they receive a Resume Service Test request from the Buyer/Client.
- [R55] If the request is not successful, the Seller/Server **MUST NOT** resume the Service Test.
- [R56] If the Seller/Server encounters errors, they **MUST** return an error with explanation to the Buyer/Client
3. The Service Test tests and measurements are resumed.

#### 7.2.5 Modify Service Test

1. The Buyer/Client is able to request modifications to an existing Service Test.
- [R57] The Buyer's Modify Service Test request **MUST** include the Service Test Identifier.
- [R58] The Buyer's Modify Service Test request **MUST** contain at least one of the following attributes:
- Service Test Description
  - Service Test End Date/Time
  - Mode
  - Service Test Name
  - Service Test Start Date/Time
  - Related Service
  - Test Specification Identifier
  - Test Specification Version
  - Test Attribute Schema
- [R59] The Service Test **MUST** be in the SUSPENDED or SCHEDULED state for a Buyer/Client to modify it.

2. The Seller/Server responds to the Buyer's request.
  - [R60] The Seller's response to a Modify Service Test request **MUST** echo back Buyer/Client provided attributes.
  - [R61] The Seller's response to a Modify Service Test request **MUST** include the following attributes:
    - Service Test Identifier
    - Service Test State
  - [R62] If the request is not successful, the Seller/Server **MUST NOT** echo back all Buyer/Client provided attributes.
  - [R63] If the request is not successful, the Seller/Server **MUST NOT** return a Service Test Identifier or Service Test State.
  - [R64] If the Seller/Server encounters errors, they **MUST** return an error with explanation to the Buyer/Client.
3. The Seller/Server modifies the Service Test as requested.

#### 7.2.6 Modify Service Test w/o Test Specification

1. The Buyer/Client is able to request modifications to an existing Service Test.
  - [R65] The Buyer's Modify Service Test request **MUST** include the Service Test Identifier.
  - [R66] The Buyer's Modify Service Test request **MUST** contain at least one of the following attributes:
    - Service Test Description
    - Service Test End Date/Time
    - Mode
    - Service Test Name
    - Service Test Start Date/Time
    - Related Service
    - Test Attribute Schema
  - [R67] The Service Test **MUST** be in the SUSPENDED or SCHEDULED state for a Buyer/Client to modify it.

2. The Seller/Server responds to the Buyer's request.
  - [R68] The Seller's response to a Modify Service Test request **MUST** echo back Buyer/Client provided attributes.
  - [R69] The Seller's response to a Modify Service Test request **MUST** include the following attributes:
    - Service Test Identifier
    - Service Test State
  - [R70] If the request is not successful, the Seller/Server **MUST NOT** echo back all Buyer/Client provided attributes.
  - [R71] If the request is not successful, the Seller/Server **MUST NOT** return a Service Test Identifier or Service Test State.
  - [R72] If the Seller/Server encounters errors, they **MUST** return an error with explanation to the Buyer/Client.
3. The Seller/Server modifies the Service Test as requested.

#### 7.2.7 Delete Service Test

1. The Buyer/Client requests an existing Service Test be deleted.
  - [R73] The Buyer/Client's Delete Service Test request **MUST** contain the Service Test Identifier.
  - [R74] The Buyer/Client's Delete Service Test request **MUST NOT** contain any other attributes from Table 8.
2. The Seller/Server's response to the Buyer/Client request indicating if the request is successful or unsuccessful.
  - [R75] If successful, the Seller/Server **MUST** echo back the Service Test Identifier and indicate that the request was successful.
  - [R76] If unsuccessful, the Seller/Server **MUST NOT** echo back the Service Test Identifier and indicate the request was unsuccessful.
  - [R77] If the Seller/Server encounters errors, they **MUST** return an error with explanation to the Buyer/Client.
3. The Service Test is deleted.

Note: Test results are available for some period of time agreed to by the Buyer/Client and Seller/Server after a Service Test is deleted.

### 7.2.8 Retrieve Service Test List

1. The Buyer/Client submits a Retrieve Service Test List request using none or more of the filter criteria.

**[R78]** The Buyer/Client's Retrieve Service Test List request **MUST** contain none or more of the following filter criteria:

- Related Service Identifier
- Start Date Time
- End Date Time
- Service Test Specification Identifier

2. The Seller/Server returns a response to the Buyer.

**[R79]** If successful, the Seller/Server **MUST** return a list of Service Test Identifiers that match the filter criteria.

**[R80]** If successful but no matches to the filter criteria are found, the Seller/Server **MUST** return an empty list.

**[R81]** If unsuccessful, the Seller/Server **MUST NOT** return a list of Service Test Identifiers or an empty list.

**[R82]** If errors are encountered, the Seller/Server **MUST** return an indication of the errors.

3. The Buyer has a list of the Service Test Identifiers that match the filter criteria.

### 7.2.9 Retrieve Service Test by Identifier

1. The Buyer/Client submits a Retrieve Service Test by Identifier request that includes the Service Test Identifier.

**[R83]** The Buyer/Client's Retrieve Service Test by Identifier request **MUST** include the Service Test Identifier and only the Service Test Identifier.

2. The Seller/Server responds to the Retrieve Service Test by Identifier request.

**[R84]** If successful, the Seller/Server **MUST** include the following attributes in their response:

- Service Test Identifier
- Start Date Time

- Service Test Specification Identifier
- Service Test Specification Version
- Related Service Identifier
- Service Test Results

[O4] If successful, the Seller/Server **MAY** include the following attributes in their response:

- Service Test Name
- Service Test Description
- End Date Time
- Valid For
- Related Service Name

[R85] If unsuccessful, the Seller/Server **MUST NOT** return any attributes in their response.

[R86] If errors are encountered, the Seller/Server **MUST** return an indication of the errors encountered.

3. The Buyer/Client has all details of a Service Test.

### 7.3 Notifications

Notifications may be supported by the Seller/Server. If the Seller/Server supports notifications, the Buyer/Client may subscribe to them.

#### 7.3.1 Register for Notifications

1. The Buyer/Client registers for Notifications.

Attribute Name	Description	Value	Comments
Register Notification Identifier	The Identifier of the register for notification request	String	Set by Seller/Server
Callback Address	The Callback Address being registered	String	Set by Buyer/Client

Attribute Name	Description	Value	Comments
Notification Type	The type of notifications being registered for.	One or more of: <ul style="list-style-type: none"> <li>• Service Test Create</li> <li>• Service Test Attribute Value Change</li> <li>• Service Test Delete</li> <li>• Service Test Specification Create</li> <li>• Service Test Specification Attribute Value Change</li> <li>• Service Test Specification Delete</li> </ul>	Set by the Buyer/Client
Action	Specification to Start Notifications or Stop Notifications.	One of: <ul style="list-style-type: none"> <li>• START</li> <li>• STOP</li> </ul>	

**Table 10 – Register for Notification Attributes**

**[R87]** The Buyer/Client Register for Notifications request **MUST** include the following:

- Callback Address
- One or more Notification Types
- Action = START

2. The Seller/Server responds with an indication of success and includes the Register Notification Identifier.

**[R88]** If successful, the Seller/Server response **MUST** indicate success and include the Register Notification Identifier.

**[R89]** If successful, the Seller/Server **MUST** begin sending the appropriate notifications to the Buyer/Client.

**[R90]** The Seller/Server **MUST NOT** send notifications if the Buyer/Client has not registered for them.

[R91] If unsuccessful, the Seller/Server **MUST NOT** return a Register Notification Identifier.

[R92] If the Seller/Server experiences any errors, they **MUST** return an error indication to the Buyer/Client.

### 7.3.2 Unregister for Notifications

1. The Buyer/Client unregisters for Notifications.

[R93] The Buyer/Client Register for Notifications request **MUST** include the following:

- Callback Address
- One or more Notification Types
- Action = STOP

2. The Seller/Server responds with an indication of success.

[R94] If successful, the Seller/Server response **MUST** indicate success.

[R95] If successful, the Seller/Server **MUST** stop sending the appropriate notifications to the Buyer/Client.

[R96] If unsuccessful, the Seller/Server **MUST NOT** stop sending the appropriate notifications to the Buyer/Client.

[R97] If the Seller/Server experiences any errors, they **MUST** return an error indication to the Buyer/Client.

### 7.3.3 Generate Notification

1. The Seller/Server generates a Notification to a Buyer/Client who has registered for Notifications when a Notification Event occurs.

Attribute Name	Description	Value	Comments
Event Notification Identifier	The Identifier of the notification	String	Set by Seller/Server
Service Test Identifier	The Identifier of the Service Test	String	Set by Seller/Server
Notification Type	The type of notification being generated.	One or more of: <ul style="list-style-type: none"> <li>• Service Test Create</li> <li>• Service Test Attribute Value Change</li> </ul>	Set by the Seller/Server

Attribute Name	Description	Value	Comments
		<ul style="list-style-type: none"> <li>• Service Test Delete</li> <li>• Service Test Specification Create</li> <li>• Service Test Specification Attribute Value Change</li> <li>• Service Test Specification Delete</li> </ul>	
Event Time	The time that the event in the notification occurred.	Date/time	Set by Seller

**Table 11 – Notification Attributes**

**[R98]** The Seller/Server’s Notification **MUST** include the following attributes:

- Event Identifier
- Notification Type
- Event Date Time

2. The Buyer/Client acknowledges the receipt of the notification.

**[O5]** If the Seller/Server does not receive a response from the Buyer/Client to the notification, they **MAY** cease sending notifications to the Callback Address.

#### 7.4 Results

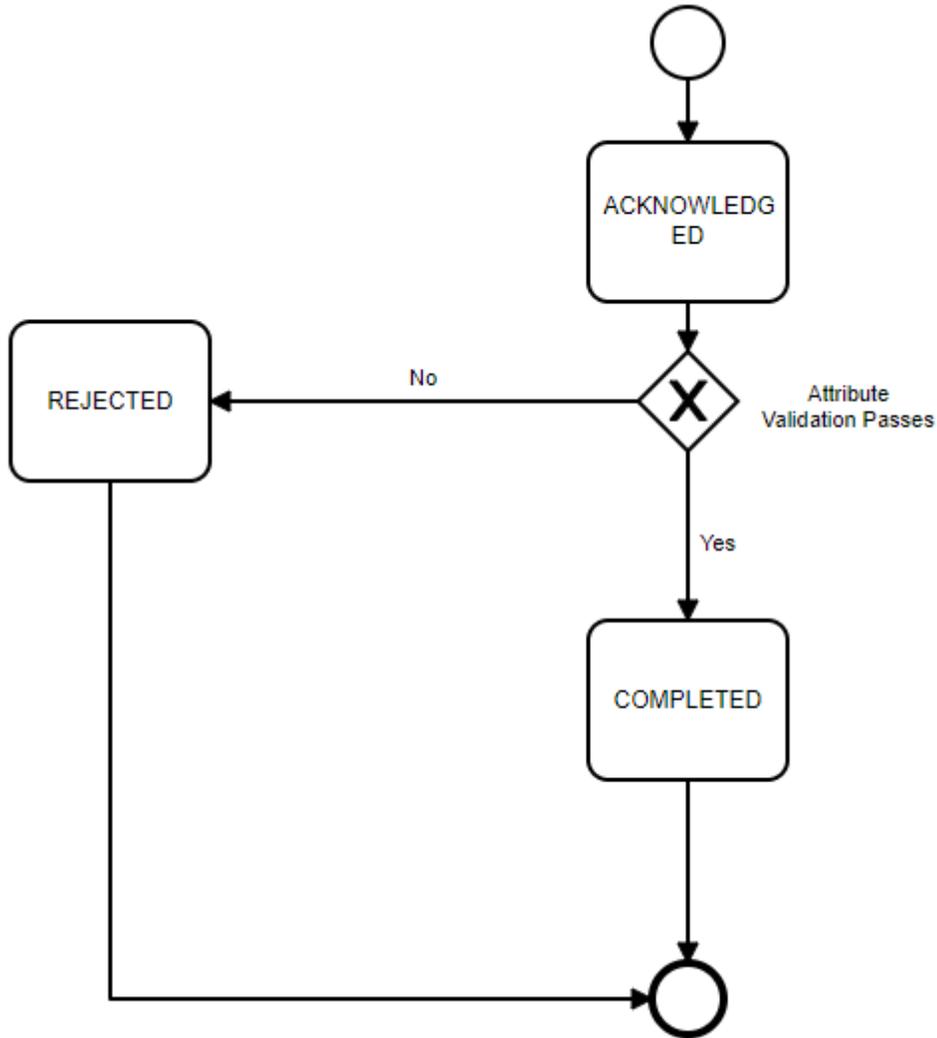
The retrieval of results from SFT is performed by using a Retrieve Service Test.

## 8 Process Flows

The Process Flows for Service Test Specification and Service Test are defined in the following sections.

### 8.1 Create Service Test Specification

The Process Flow for the Create Service Test Specification is defined in Figure 2.



**Figure 2 – Create Service Test Specification Process Flow**

The Create Service Test Specification States are detailed in Table 12.

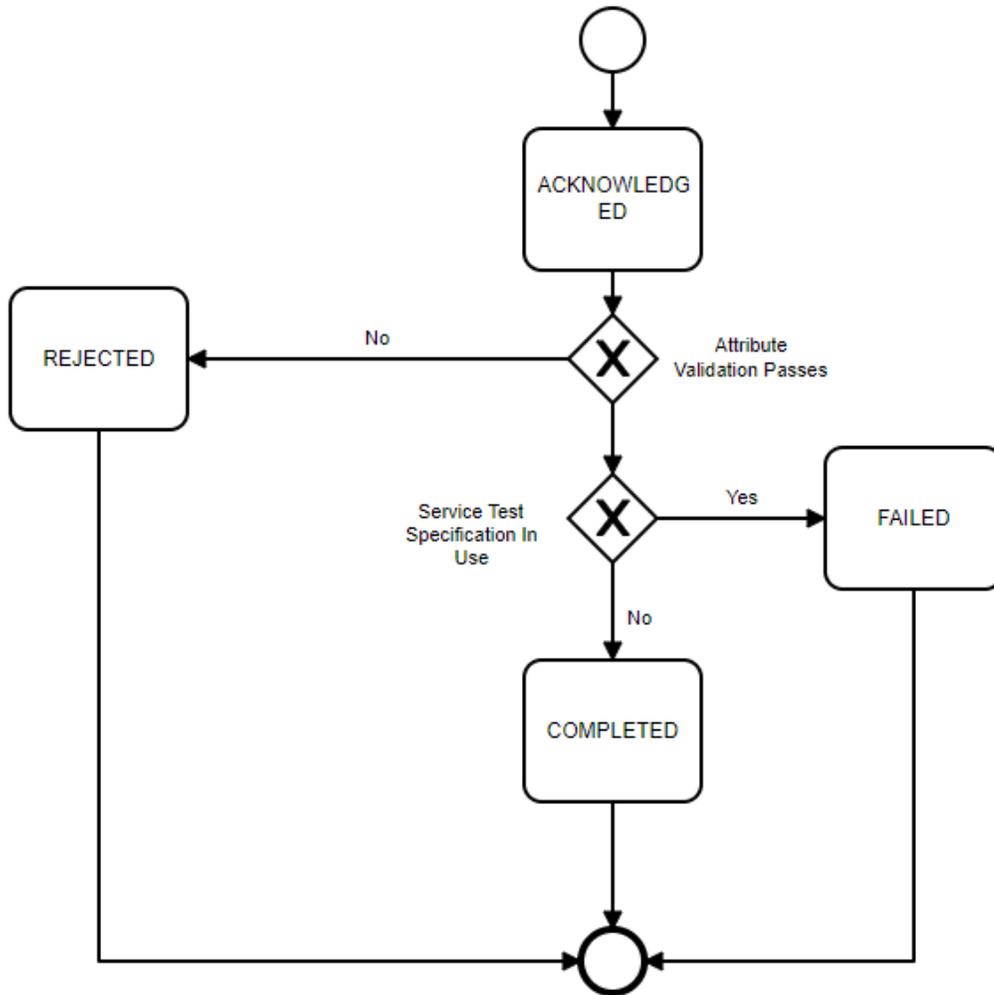
State Name	Description	Comments
ACKNOWLEDGED	The Create Service Test Specification request has	

	been received from the Buyer/Client and the Seller/Server has assigned a Service Test Specification Identifier to it. If the request attributes fail validation, the Create Service Test Specification moves to the REJECTED state. If the attributes pass validation it moves to the COMPLETED state.	
COMPLETED	The Service Test Specification has been created and is ready for use by the Buyer/Client.	
REJECTED	The Create Service Test Specification does not pass attribute validation and the Create Service Test Specification is rejected.	

**Table 12 – Create Service Test Specification States**

## 8.2 Delete Service Test Specification

The Process Flow for the Delete Service Test Specification is defined in Figure 3.



**Figure 3 – Delete Service Test Specification Process Flow**

The Delete Service Test Specification States are detailed in Table 13.

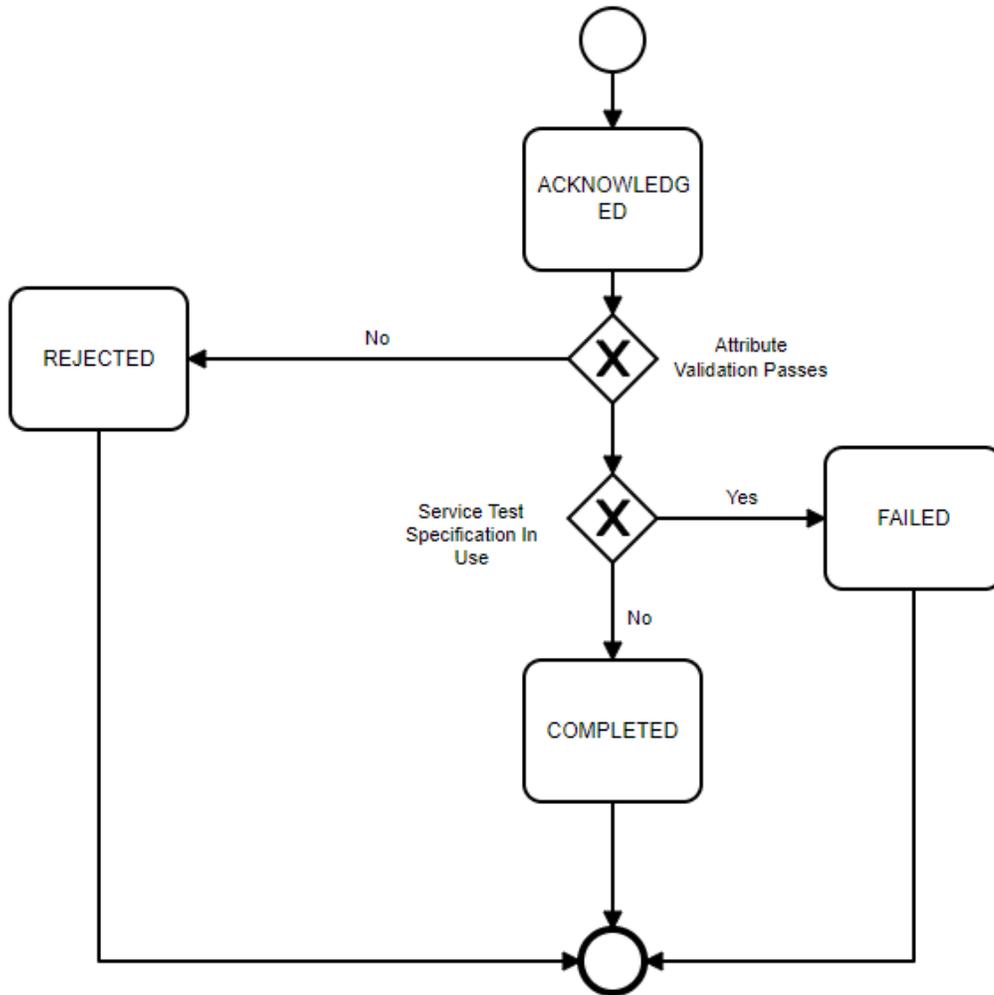
State Name	Description	Comments
ACKNOWLEDGED	The Delete Service Test Specification request has been received from the Buyer/Client and the Seller/Server has assigned a Delete Service Test Specification Identifier to it. If the request attributes fail validation, the Delete Service Test Specification moves to the REJECTED state. If the	

	Service Test Specification is in use in a Service Test, the Service Test Specification cannot be deleted and moves to the FAILED state. If the attributes pass validation and the Service Test Specification is not in use, the Service Test Specification is deleted and Delete Service Test Specification moves to the COMPLETED state.	
COMPLETED	The Service Test Specification has been deleted.	
FAILED	The Delete Service Test Specification fails because the Service Test Specification is in use in one or more Service Tests.	
REJECTED	The Delete Service Test Specification does not pass attribute validation and the Delete Service Test Specification is rejected.	

**Table 13 – Delete Service Test Specification States**

### 8.3 Modify Service Test Specification

The Process Flow for the Modify Service Test Specification is defined in Figure 4.



**Figure 4 – Modify Service Test Specification Process Flow**

The Modify Service Test Specification States are detailed in Table 13.

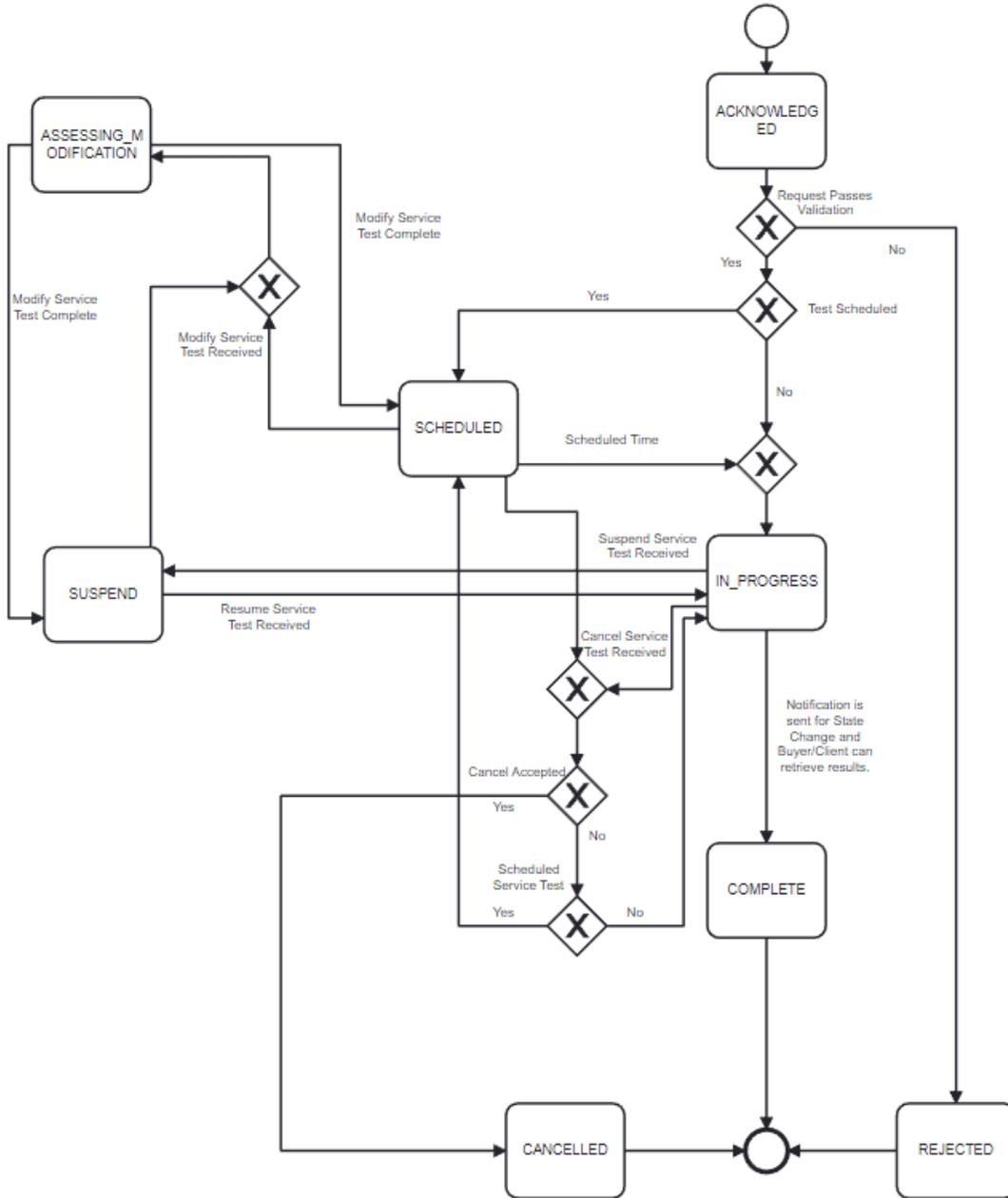
State Name	Description	Comments
ACKNOWLEDGED	The Modify Service Test Specification request has been received from the Buyer/Client and the Seller/Server has assigned a Modify Service Test Specification Identifier to it. If the request attributes fail validation, the Modify Service Test Specification moves to the REJECTED	

	state. If the Service Test Specification is in use in a Service Test, the Service Test Specification cannot be modified and moves to the FAILED state. If the attributes pass validation and the Service Test Specification is not in use, the Service Test Specification is modified and Modify Service Test Specification moves to the COMPLETED state.	
COMPLETED	The Service Test Specification has been Modified.	
FAILED	The Modify Service Test Specification fails because the Service Test Specification is in use in one or more Service Tests.	
REJECTED	The Modify Service Test Specification does not pass attribute validation and the Modify Service Test Specification is rejected.	

**Table 14 – Modify Service Test Specification States**

#### 8.4 Create Service Test Process Flow

The process Flow for Create Service Test is shown in Figure 5.



**Figure 5 – Create Service Test Process Flow**

The Create Service Test States are detailed in Table 15.

State Name	Description	Comments
ACKNOWLEDGED	The Create Service Test request has been received from the Buyer/Client and the Seller/Server has	



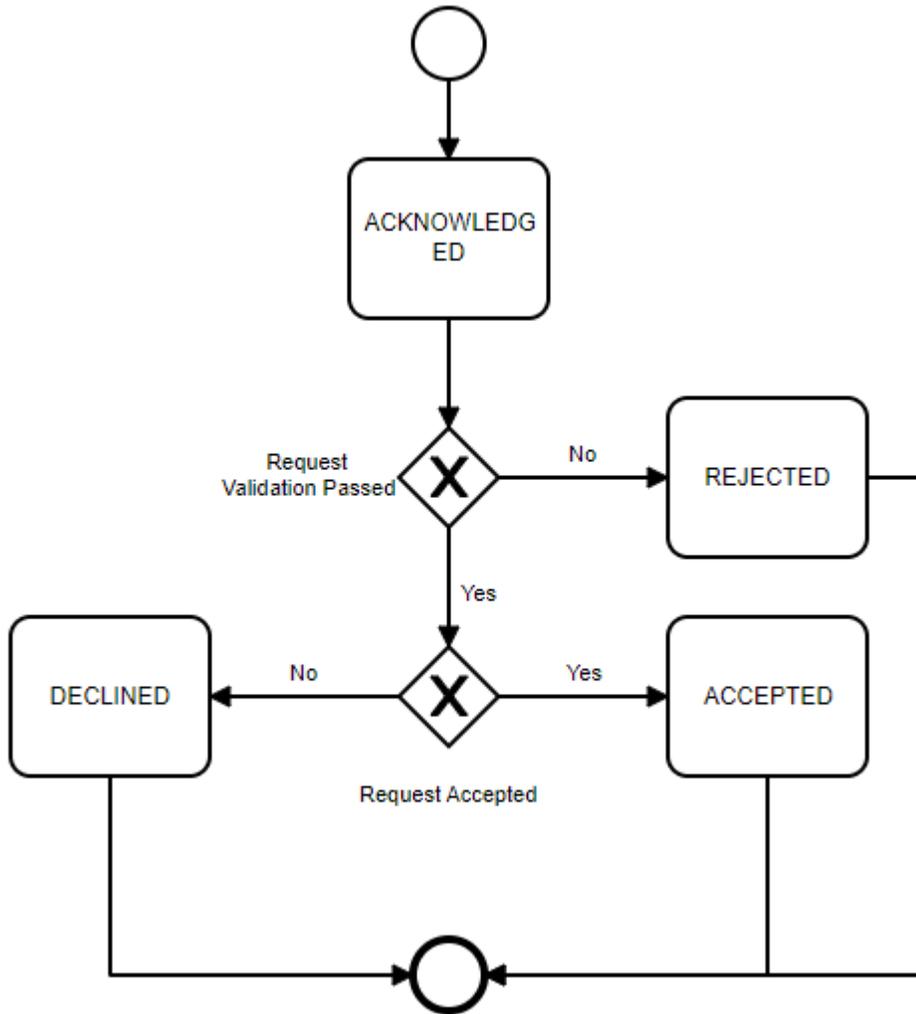
State Name	Description	Comments
	assigned a Service Test Identifier to it. If the request attributes fail validation, the Create Service Test moves to the REJECTED state. If the attributes pass validation, it is then determined if the Create Service Test Start Date Time is immediate or if the Create Service Test Start Date Time indicates that the Service Test is to be scheduled for a later Date Time. If the Service Test is to be scheduled the Service Test moves to the SCHEDULED state and awaits the scheduled date and time. If the Service Test is to be performed immediately, the Service Test moves to the IN_PROGRESS state and test measurements begin.	
ASSESSING_MODIFICATION	A Modify Service Test request while the Service Test is in the SUSPEND or SCHEDULED state. If the Modify Service Test is accepted, the Service Test is updated. If the Modify Service Test is declined, the Service Test is not updated.	
CANCELLED	A Cancel Service Test request is received from the Buyer/Client. If the request is accepted, the Service Test moves to the CANCELLED state.	
COMPLETED	The Service Test has reached the End Date Time or has completed all test measurements.	

State Name	Description	Comments
IN_PROGRESS	Whether an immediate request or a scheduled request, the Service Test moves to the IN_PROGRESS state when it begins performing test measurements. If a Cancel Service Test request is received and accepted, the Service Test moves to the CANCELLED state. If the Delete Service Test request is declined, the Service Test returns to the IN_PROGRESS state and continues test measurements until they are completed. If a Suspend Service Test request is received, the Service Test moves to the SUSPEND state.	
REJECTED	The Create Service Test request fails validation and is rejected.	
SCHEDULED	The Service Test is scheduled to start at a later time. The Service Test stays in the SCHEDULED state until the Start Date and Time is reached. The Service Test moves to IN_PROGRESS when the Start Date and Time is reached.	
SUSPEND	A Service Test in the IN_PROGRESS state receives a Suspend Service Test request. The Service Test moves to the SUSPEND state.	A Service Test cannot be modified unless it is in the SUSPEND state.

**Table 15 – Create Service Test States**

### 8.5 Delete Service Test Process Flow

The process flow for Delete Service Test is shown in Figure 6.



**Figure 6 – Delete Service Test Process Flow**

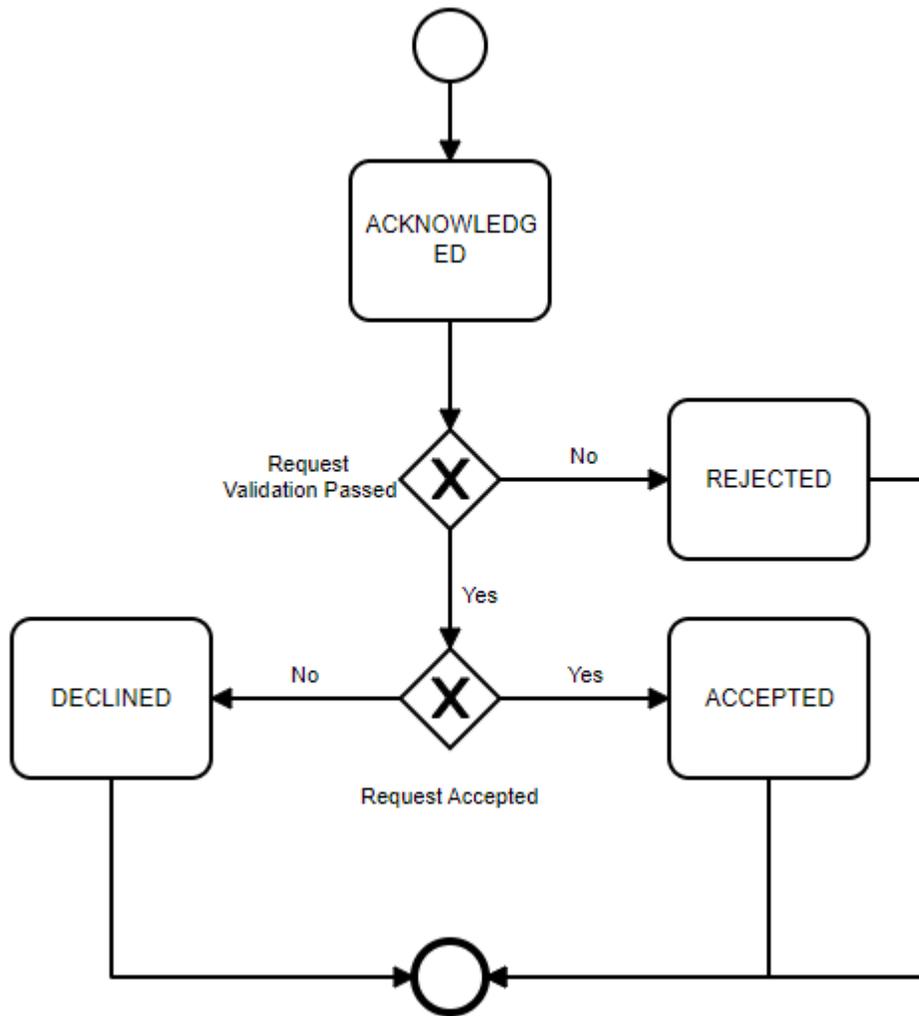
The Delete Service Test states are defined in Table 16.

State Name	Description	Comments
ACKNOWLEDGED	The Delete Service Test request has been received from the Buyer/Client and the Seller/Server has assigned a Delete Service Test Identifier to it. If the request attributes fail validation, the Delete Service Test moves to the REJECTED state. If the attributes pass validation, it is determined if the Delete Service Test is accepted or declined. If accepted, it moves to the ACCEPTED state. If declined, it moves to the DECLINED state.	
ACCEPTED	The Delete Service Test is accepted, and the Service Test is deleted.	
DECLINED	The Delete Service Test is declined, and the Service Test is not deleted.	
REJECTED	The Delete Service Test does not pass attribute validation and the Delete Service Test is rejected.	

**Table 16 – Delete Service Test States**

### 8.6 Modify Service Test Process Flow

The Modify Service Test process flow is shown in Figure 7.



**Figure 7 – Modify Service Test Process Flow**

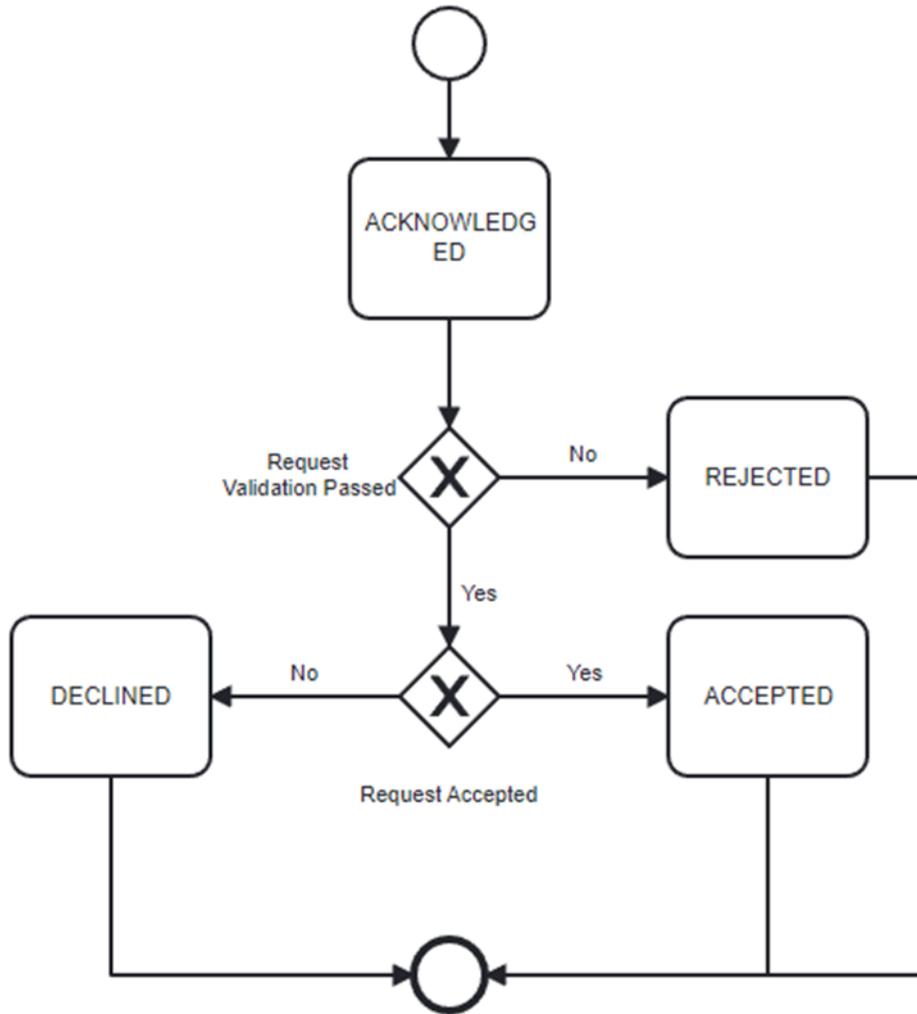
The Modify Service Test states are defined in Table 17.

State Name	Description	Comments
ACKNOWLEDGED	The Modify Service Test request has been received from the Buyer/Client and the Seller/Server has assigned a Modify Service Test Identifier to it. If the request attributes fail validation, the Modify Service Test moves to the REJECTED state. If the attributes pass validation, it is determined if the Modify Service Test is accepted or declined. If accepted, it moves to the ACCEPTED state. If declined, it moves to the DECLINED state.	
ACCEPTED	The Modify Service Test is accepted, and the Service Test is modified.	
DECLINED	The Modify Service Test is declined, and the Service Test is not modified.	
REJECTED	The Modify Service Test does not pass attribute validation and the Modify Service Test is rejected.	

**Table 17 – Modify Service Test States**

### 8.7 Suspend Service Test Process Flow

The Suspend Service Test process flow is shown in Figure 8.



**Figure 8 – Suspend Service Test Process Flow**

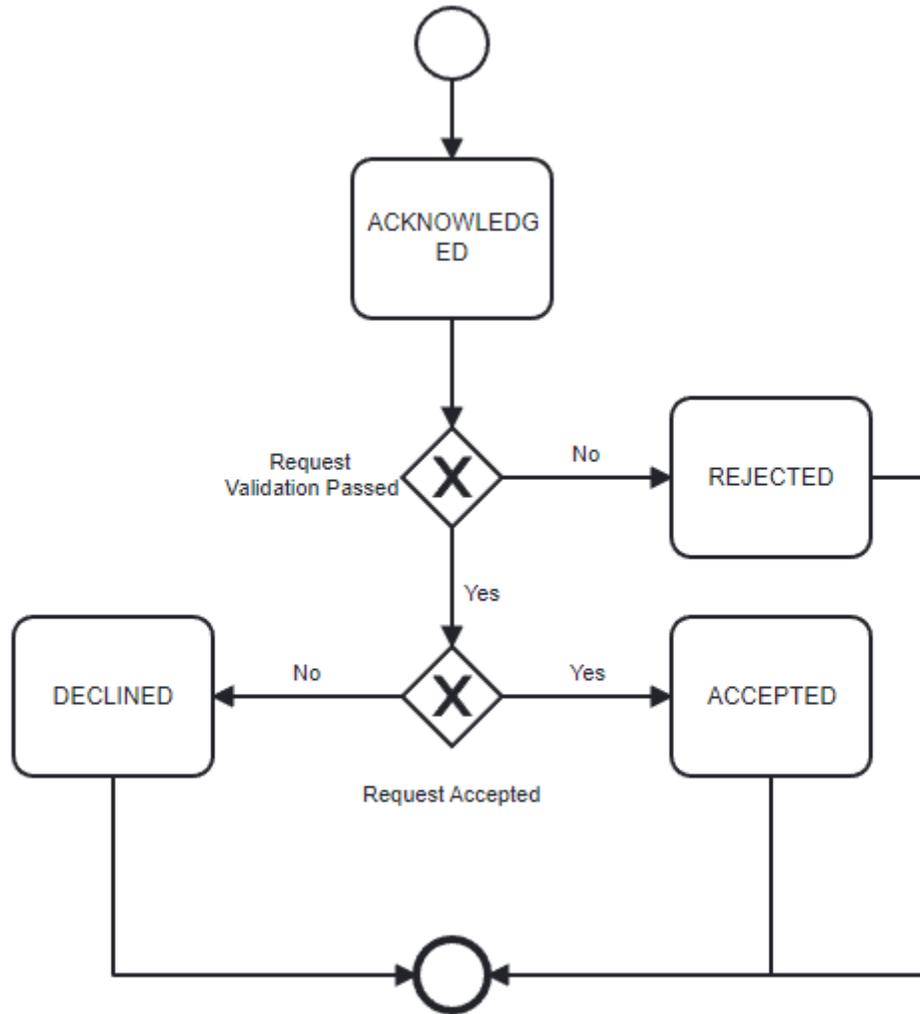
The Suspend Service Test states are defined in Table 18.

State Name	Description	Comments
ACKNOWLEDGED	The Suspend Service Test request has been received from the Buyer/Client and the Seller/Server has assigned a Suspend Service Test Identifier to it. If the request attributes fail validation, the Suspend Service Test request moves to the REJECTED state. If the attributes pass validation, it is determined if the Suspend Service Test request is accepted or declined. If accepted, it moves to the ACCEPTED state. If declined, it moves to the DECLINED state.	
ACCEPTED	The Suspend Service Test is accepted, and the Service Test is suspended.	
DECLINED	The Suspend Service Test is declined, and the Service Test is not suspended.	
REJECTED	The Suspend Service Test does not pass attribute validation and the Suspend Service Test is rejected.	

**Table 18 – Suspend Service Test States**

### 8.8 Resume Service Test Process Flow

The Resume Service Test process flow is shown in Figure 9.



**Figure 9 – Resume Service Test Process Flow**

The Resume Service Test states are defined in Table 19.

State Name	Description	Comments
ACKNOWLEDGED	The Resume Service Test request has been received from the Buyer/Client and the Seller/Server has assigned a Resume Service Test Identifier to it. If the request attributes fail validation, the Resume Service Test request moves to the REJECTED state. If the attributes pass validation, it is determined if the Resume Service Test request is accepted or declined. If accepted, it moves to the ACCEPTED state. If declined, it moves to the DECLINED state.	
ACCEPTED	The Resume Service Test is accepted, and the Service Test is resumed.	
DECLINED	The Resume Service Test is declined, and the Service Test is not resumed.	
REJECTED	The Resume Service Test does not pass attribute validation and the Resume Service Test is rejected.	

**Table 19 – Resume Service Test States**

## 9 References

- [1] IETF RFC 2119, *Key words for use in RFCs to Indicate Requirement Levels*, by S. Bradner, March 1997
- [2] IETF RFC 8174, *Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words*, by B. Leiba, May 2017, Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.
- [3] MEF 55.1, *Lifecycle Service Orchestration (LSO): Reference Architecture and Framework*, January 2021
- [4] TMF653, *Service Test Management*, July 2021

## Appendix A Acknowledgements

The following contributors participated in the development of this document and have requested to be included in this list.

- Mike **BENCHECK**
- Johanne **MAYER**
- Jack **PUGACZEWSKI**
- Karthik **SETHURAMAN**
- Mehmet **TOY**

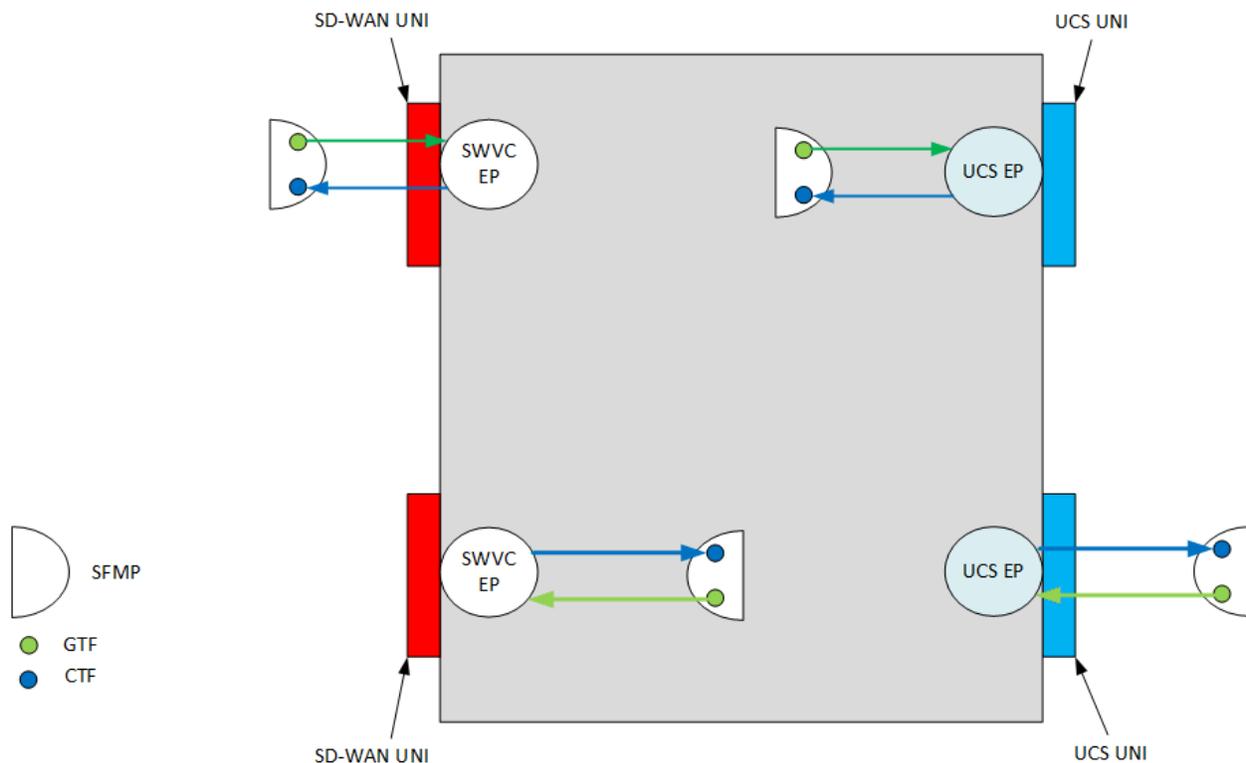
## Appendix B Test Access Provisioning

Test Access connects the Test Resources to the Service Under Test (SUT). Test Access is provisioned by the Service Provisioning system before a Service Test can be run. This appendix explains the types of Test Resources and how the Service Provisioning system configures the Test Access before a Service Test is performed.

### B.1 Test Resources

The term Test Resource is used to describe the function or device that performs the test measurements as instructed by the Service Test and Service Test Specification. There are two general types of Test Resources, Imbedded and External. An Imbedded Test Resource is included within a device or application that is within the path of a service. Examples of Imbedded Test Resources are applications that reside within a device and can be inserted into the path of a service within the device. External Test Resources are probes or other type of Test Resource that is externally connected to the device on which the SUT is provisioned.

Test Resources, as described in this document, are made up of three components, the Service Function Measurement Point (SFMP), the Generator Test Function (GTF), and the Collector Test Function (CTF). These are shown in both Imbedded and External uses in Figure 10.



**Figure 10 – Imbedded and External SFMP, GTF, and CTF**

The SFMP always contains one GTF and one CTF. In some cases, the GTF or CTF may be inactive. At least one of them is always active. As their names imply, the GTF generates test frames or packets that are formatted to be passed on the SUT. The CTF collects and counts,

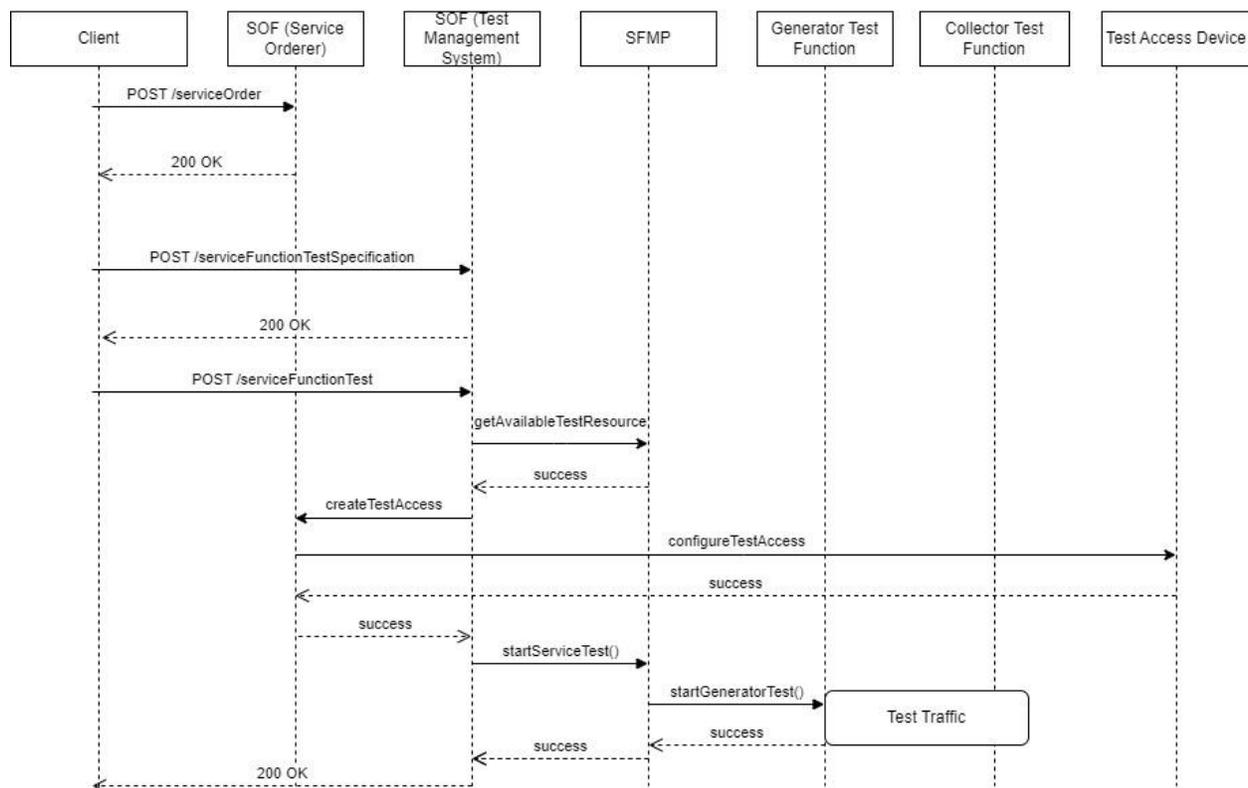
collects and processes, or collects and performs measurements on frames or packets that have been received from a GTF. Once the CTF has counted, processed, or performed measurements, they discard the frames or packets. Testing is performed using a pair of GTF/CTFs. These may be co-located as when testing to a loopback or may be at either end of a service when testing between an ordered pair of points within the service.

The Test Resources that are shown in Figure 10 are controlled via the Test Management System which provisions, configures, and retrieves test results from the Test Resources. The Test Management System is assumed to be a part of the Service Orchestration Function (SOF). The Service Test Specifications are configured in the Test Management System. Service Tests requests are sent to the Test Management System. The Test Management System then configures the SFMP to perform the test defined in the Service Test Specification and Service Test.

### B.2 Test Access Configuration

The SFMP is connected to the SUT, either as an Imbedded or an External Test Resource. The physical or logical connection between the SUT and the SFMP is made using the Service Provisioning System. The SUT is identified in the Service Test as the *Related Service Identifier*. The Test Management System uses the *Related Service Identifier* to identify the SUT to the Service Provisioning System and requests that the SUT either be connected to an Imbedded or External SFMP including what the GTF and CTF are connected to.

Once the Service Provisioning Service has connected the SUT to the SFMP, the Test Management System then configures the SFMP, and test measurements are performed as specified in the Service Test. This process is shown in the sequence diagrams shown in Figure 11.



### **Figure 11 – Test Access Sequence Diagram**

As seen in Figure 11, the request from the Test Management System verifies that an appropriate SFMP is available at the specified test point. If there is an SFMP available, the Test Management System sends a request to the Service Provisioning System to connect the SFMP to the SUT with the GTF and CTF connected appropriately to perform the Service Test.

Once the Service Provisioning System has completed the appropriate connections, it responds with a completion notice to the Test Management System. The Test Management System performs the Service Test and upon completing the Service Test, requests that the Service Provisioning System restore the SUT to its original configuration, which restores the service to its normal path.

The Test Management System then collects all measurements, performs any required calculations, and reports the results of the Service Test to the requestor.