

MEF Legato Interface Profile SpecificationFebruary 2019MEF 73 draft 0.1

Working Draft

MEF 73 draft 0.1

MEF Legato Interface Profile Specification

April 2019

This draft represents MEF work in progress and is subject to change.

Disclaimer

© MEF Forum 2019. 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:

1. 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
2. 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
3. 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

[1 List of Contributing Members 4](#_Toc13580295)

[2 Abstract 4](#_Toc13580296)

[3 Terminology and Abbreviations 5](#_Toc13580297)

[4 Compliance Levels 6](#_Toc13580298)

[5 Introduction 7](#_Toc13580299)

[6 Use Cases and Flow 7](#_Toc13580300)

[7 Interface Requirements 7](#_Toc13580301)

[8 Interface Information Model 8](#_Toc13580302)

[8.1 UML Diagrams 8](#_Toc13580303)

[8.1.1 [d.name/] 8](#_Toc13580304)

[8.2 Class Descriptions 9](#_Toc13580305)

[8.2.1 [cl.name/] 9](#_Toc13580306)

[8.3 Data Types and Enumeration 11](#_Toc13580307)

[8.3.1 Data Type: [dt.name/] 11](#_Toc13580308)

[8.3.2 Enumeration: [dt.name/] 12](#_Toc13580309)

[8.3.3 Primitive: [dt.name/] 12](#_Toc13580310)

[8.4 Interface Operations 13](#_Toc13580311)

[8.4.1 [it.name/] 13](#_Toc13580312)

[9 MEF Service Common Model (MSCM) Mapping 14](#_Toc13580313)

[10 References 14](#_Toc13580314)

[Appendix A Appendix Title (Informative) 15](#_Toc13580315)

[A.1 MEF Access E-Line Service Specification 15](#_Toc13580316)

[A.1.1 Appendix second-level subsection 15](#_Toc13580317)

# List of Figures

**No table of figures entries found.**

# List of Tables

[Table 1 – Terminology and Abbreviations 5](#_Toc2090094)

# 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.

1. This list will be finalized before Letter Ballot. Any member that comments in at least one CfC is eligible to be included by opting in before the Letter Ballot is initiated. Note it is the MEF member that is listed here (typically a company or organization), not their individual representatives.

* ABC Networks
* XYZ Communications

# Abstract

The MEF Legato Interface Profile Specification (IPS) describes use cases, requirements and the information model (including Class diagrams and descriptions, Interface Operations, Sequence diagrams, State diagrams, etc) for the MEF Legato Service API (MLSAPI). The MLSAPI is expected to be applied at the MEF LSO Legato Interface Reference Point (IRP).

# 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 X [1] are included in this document by reference, and are not repeated in the table below.

| Term | Definition | Reference |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

Table 1 – Terminology and Abbreviations

# 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**.**

1. The following paragraph will be deleted if no conditional requirements are used in the document.

A paragraph preceded by **[CRa]<** specifies a conditional mandatory requirement that **MUST** be followed if the condition(s) following the “<” have been met. For example, “**[CR1]<**[D38]” indicates that Conditional Mandatory Requirement 1 must be followed if Desirable Requirement 38 has been met. A paragraph preceded by **[CDb]<** specifies a Conditional Desirable Requirement that **SHOULD** be followed if the condition(s) following the “<” have been met. A paragraph preceded by **[COc]<** specifies a Conditional Optional Requirement that **MAY** be followed if the condition(s) following the “<” have been met.

# Introduction

# Use Cases and Flow

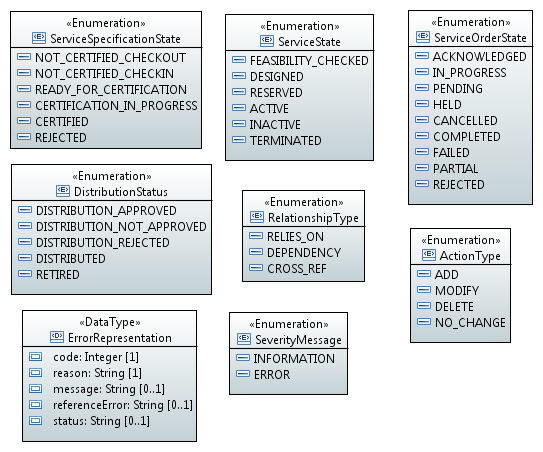
# Interface Requirements

# Interface Information Model

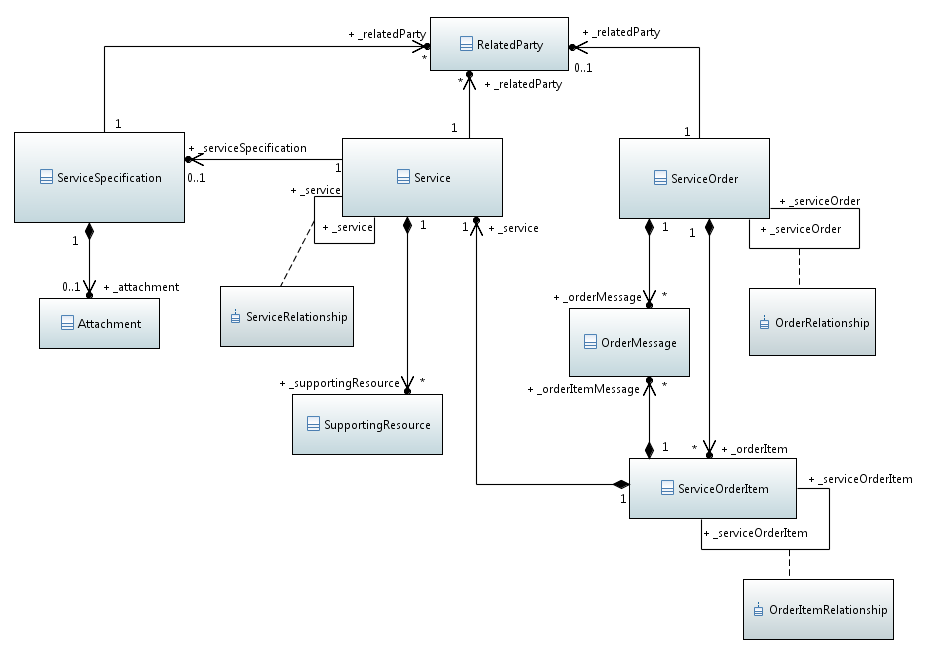
The following information has been extracted from the MLSAPI *Eclipse Papyrus* UML information model using the *Eclipse Gendoc* tool. The machine-readable information model files are maintained in the MEF-GIT repository.

## UML Diagrams

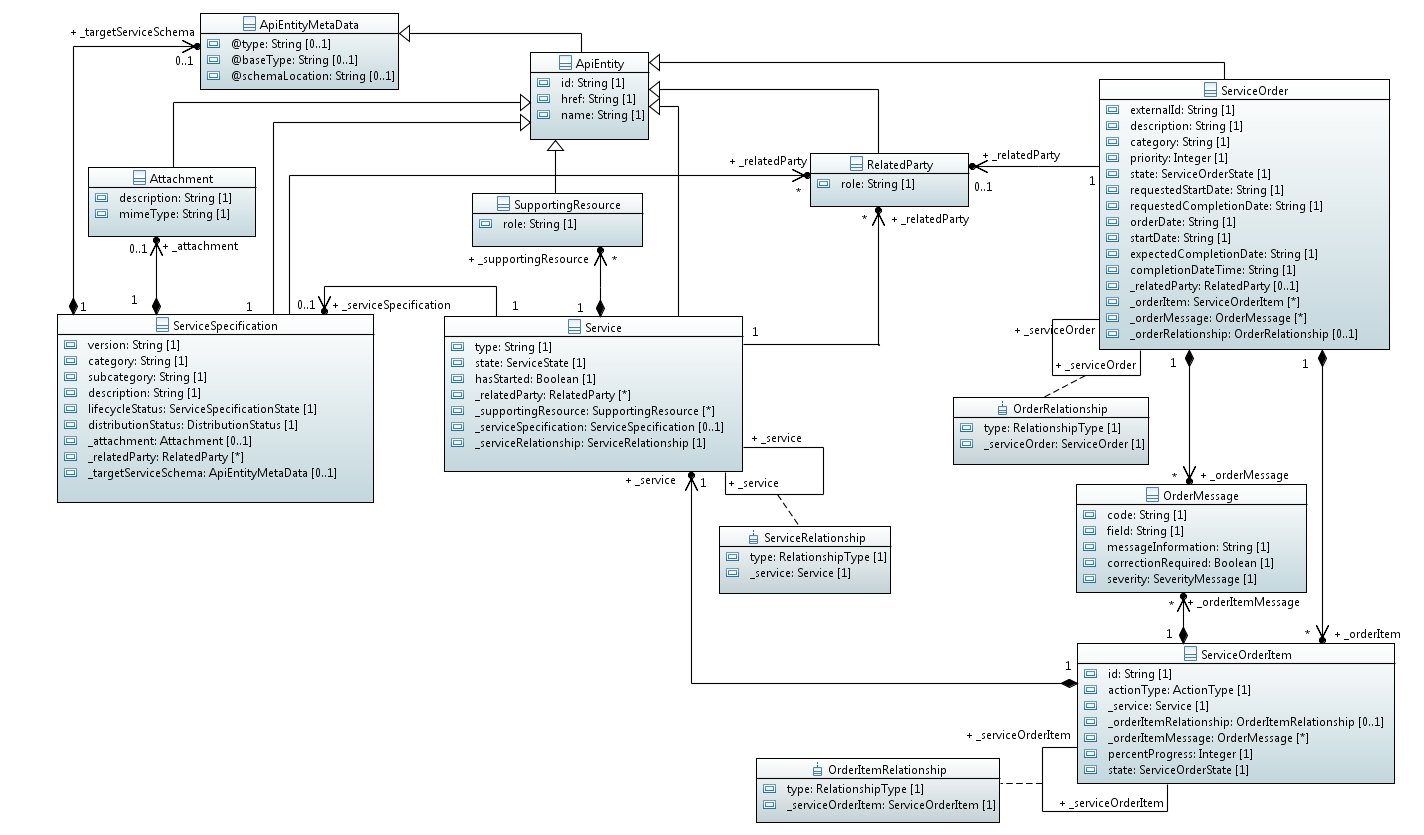
### Data Types



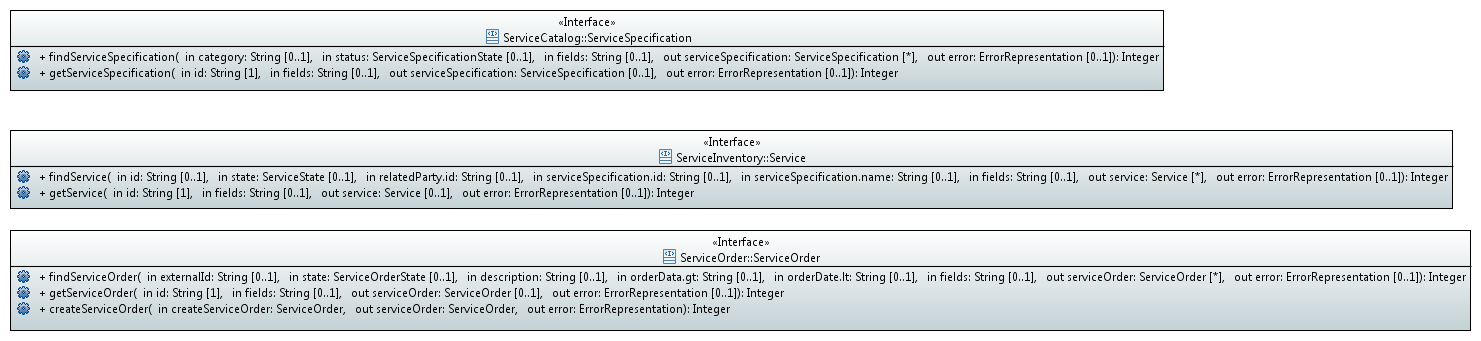
### Object Classes



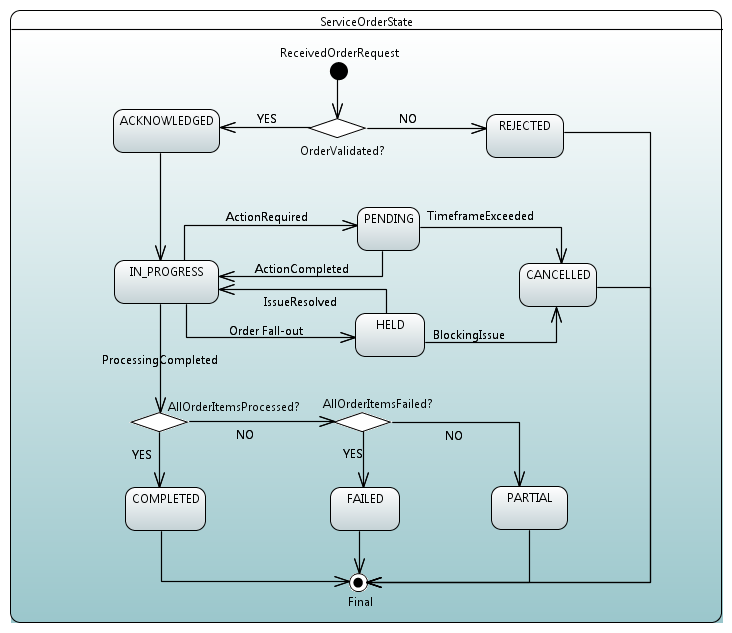
### Object Classes Details



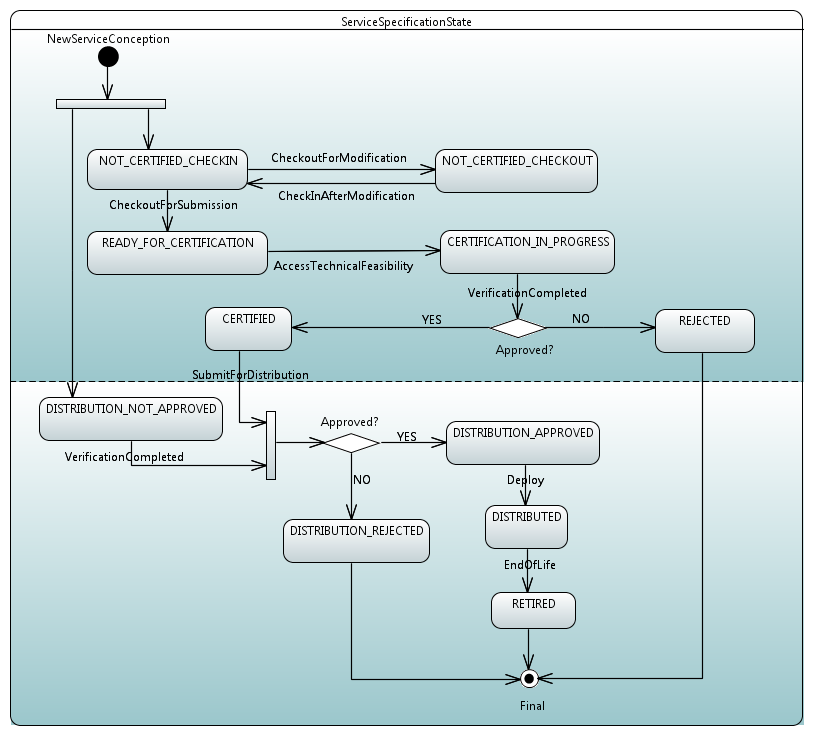
### Operations



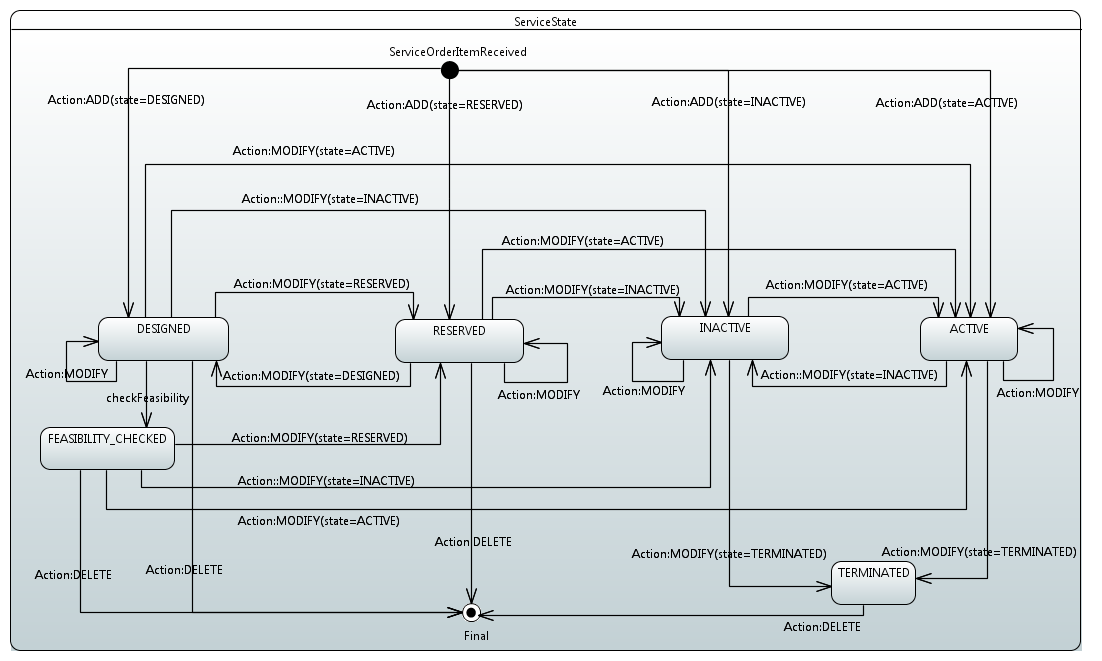
### Service Order State



### Service Specification State



### Service State



## Class Descriptions

### ApiEntity

The abstract super-class of all Legato Service API resource (object) classes for purposes to encapsulating some common attributes that will be inherited by all concrete descendents.

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| id | String | 1 | R | Unique identifier of an object instance of this entity class. |
| href | String | 1 | R | URI reference of an object instance of this entity class. |
| name | String | 1 | RW | Name of an object instance of this entity class. |
| @type | String | 0..1 | RW | The class type of a REST resource |
| @baseType | String | 0..1 | RW |  |
| @schemaLocation | String | 0..1 | RW | Provides a link to the schema describing the REST resource in URI format. |

### ApiEntityMetaData

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| @type | String | 0..1 | RW | The class type of a REST resource |
| @baseType | String | 0..1 | RW |  |
| @schemaLocation | String | 0..1 | RW | Provides a link to the schema describing the REST resource in URI format. |

### Attachment

An attchment represents a file that is used to complement the description of an entity through video, pictures, etc...

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| description | String | 1 | RW | A narrative text describing the content of the attachment |
| mimeType | String | 1 | RW | Attachment type such as video, picture, etc... |
| id | String | 1 | R | Unique identifier of an object instance of this entity class. |
| href | String | 1 | R | URI reference of an object instance of this entity class. |
| name | String | 1 | RW | Name of an object instance of this entity class. |
| @type | String | 0..1 | RW | The class type of a REST resource |
| @baseType | String | 0..1 | RW |  |
| @schemaLocation | String | 0..1 | RW | Provides a link to the schema describing the REST resource in URI format. |

### OrderItemRelationship

Reference to an related Service Order Item and the type of association.
The type of related Service Order Item, can be : "dependency" if this Order Item needs to be "not started" until another order item is complete, "relies-on" if this Order Item relies on the related order item or simply "cross-ref" to keep track of the source Order Item

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| type | RelationshipType | 1 | RW |  |

### OrderMessage

This class offers the capability to provide additional information about the Order.

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| code | String | 1 | RW | A code associated to this message |
| field | String | 1 | RW | Service Order attribute related to this error message |
| messageInformation | String | 1 | RW | Additional description text related to this message |
| correctionRequired | Boolean | 1 | RW | Indicator that an action is required to allow service order fullfilment to follow up |
| severity | SeverityMessage | 1 | RW | Severity of the message (information, error) |

### OrderRelationship

Reference to an related Service Order and the type of association.
The type of related Service Order, can be : "dependency" if the order needs to be "not started" until another order is complete or "cross-ref" to keep track of the source order

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| type | RelationshipType | 1 | RW |  |

### RelatedParty

A related party defines a party and its role, that is linked to a specific entity (such as ServiceSpecification, ServiceOrder, Service, etc)

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| role | String | 1 | RW | Role of the related party |
| id | String | 1 | R | Unique identifier of an object instance of this entity class. |
| href | String | 1 | R | URI reference of an object instance of this entity class. |
| name | String | 1 | RW | Name of an object instance of this entity class. |
| @type | String | 0..1 | RW | The class type of a REST resource |
| @baseType | String | 0..1 | RW |  |
| @schemaLocation | String | 0..1 | RW | Provides a link to the schema describing the REST resource in URI format. |

### Service

Service class offers the ability to describe the characteristics of an instance of specific type of service.
The Business Application (BA) triggers the Service Orchestration Function (SOF) to create/update/delete a Service via the Service Order/Service Order Item operations.
The SOF is responsible for orchestrating the Service fulfillment and maintaining Service instance state/attributes and notifying BA of any changes.
Functionally, a Service is instantiated using a related Service Specification instance as its template.
Thus the Services that share the same Service Specification instance would therefore share the same set of characteristics.
The MEF Service classes that describe a specific type of MEF Service instance will be associated with the Service Specification class,
as well as the Service instances will contain a reference to the MEF Service schema.
The assumption is that the Service Specifications describing the Service instance in the Service Order Item are already available (to both BA & SOF).

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| type | String | 1 | RW | Type of the service - TBD – e.g. OrderedService or PartnerService, or EVC, OVC |
| state | ServiceState | 1 | RW | The lifecycle state of the Service instance being operated upon. The "intended" state of the Service instance may be specified by the BA/requestor in the Service Order and then further maintained by the SOF. This state attribute value-transition follows the "Service Lifecycle State Machine" diagram described elsewhere in this specification. |
| hasStarted | Boolean | 1 | R | This attribute if TRUE, indicates that this Service has already been started. If the value of this attribute is FALSE, then this signifies that this Service has NOT been Started. |
| \_relatedParty | RelatedParty | 0..\* | RW |  |
| \_supportingResource | SupportingResource | 0..\* | R | A collection of references to Resource(s) supporting the Service instance. |
| \_serviceSpecification | ServiceSpecification | 0..1 | RW | The reference to the Service Specification (in the Service Catalog) used as template to instantiate the Service. |
| \_serviceRelationship | ServiceRelationship | 1 | RW | Reference to an related Service and the type of association. The type of related Service can be : "dependency" if this Service needs to be "not started" until another Service instance is complete, "relies-on" if this Order Item relies on the related Service instance or simply "cross-ref" to keep track of the source Service instance |
| id | String | 1 | R | Unique identifier of an object instance of this entity class. |
| href | String | 1 | R | URI reference of an object instance of this entity class. |
| name | String | 1 | RW | Name of an object instance of this entity class. |
| @type | String | 0..1 | RW | The class type of a REST resource |
| @baseType | String | 0..1 | RW |  |
| @schemaLocation | String | 0..1 | RW | Provides a link to the schema describing the REST resource in URI format. |

### ServiceOrder

A Service Order is used to request operations on a Service instance.
A Service Order groups one or more one Service Order Items - one per specific action on a Service instance.
The Action associated with the Service Order Item describes the operation (add, remove, update) to be applied on the specified Service instance.
The Service Order Item and its associated Action can operate on both existing (remove, update) as well as future (add) Service instance.
The Service Order is triggered from the Business Application (BA) system in charge of the Product Order management to the Service Orchestration Function (SOF) system that will orchestrate the Service fulfillment.
The assumption is that the Service Specifications describing the Service instance in the Service Order Item are already available (to both BA & SOF).

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| externalId | String | 1 | RW |  |
| description | String | 1 | RW | A free-text description of the service order. |
| category | String | 1 | RW | Used to categorize the order that can be useful for the OM system (e.g. "broadband", "TVOption", …) |
| priority | Integer | 1 | RW | A way that can be used by order requester to prioritize orders in Service Order Management system (from 0 to 4: 0 is the highest priority, and 4 the lowest). It could be for example valued by BA based on customer order requested priority |
| state | ServiceOrderState | 1 | R | Used to indicate the current status of the Service Order This read-only state attribute is maintained by the SOF. This state attribute value-transition follows the "Service Order State Machine" diagram described elsewhere in this specification. The Service Order and Order Items state transitions are tightly coupled. |
| requestedStartDate | String | 1 | RW | Order start date wished by the requestor |
| requestedCompletionDate | String | 1 | RW | Requested delivery date from the requestor perspective |
| orderDate | String | 1 | R | Date when the order was created |
| startDate | String | 1 | R | Date when the order fulfillment start – valued by server side (not requester) |
| expectedCompletionDate | String | 1 | R | Expected delivery date amended by the provider |
| completionDateTime | String | 1 | R | Date when the order was completed. |
| \_relatedParty | RelatedParty | 0..1 | RW |  |
| \_orderItem | ServiceOrderItem | 0..\* | RW |  |
| \_orderMessage | OrderMessage | 0..\* | R | An optional array of messages associated with the Order. |
| \_orderRelationship | OrderRelationship | 0..1 | RW | Reference to an related Service Order and the type of association. The type of related Service Order, can be : "dependency" if the order needs to be "not started" until another order is complete or "cross-ref" to keep track of the source order |
| id | String | 1 | R | Unique identifier of an object instance of this entity class. |
| href | String | 1 | R | URI reference of an object instance of this entity class. |
| name | String | 1 | RW | Name of an object instance of this entity class. |
| @type | String | 0..1 | RW | The class type of a REST resource |
| @baseType | String | 0..1 | RW |  |
| @schemaLocation | String | 0..1 | RW | Provides a link to the schema describing the REST resource in URI format. |

### ServiceOrderItem

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| id | String | 1 | RW |  |
| actionType | ActionType | 1 | RW | The action to be carried out on the Service instance. Can be: • add • modify • delete • noChange |
| percentProgress | Integer | 1 | R |  |
| state | ServiceOrderState | 1 | R | Used to indicate the current status of the Service Order Item. This read-only state attribute is maintained by the SOF. This state attribute value-transition follows the "Service Order Item State Machine" diagram described elsewhere in this specification. The Service Order and Order Items state transitions are tightly coupled. |
| \_service | Service | 1 | RW | The Service instance and its characteristics/attributes to be acted on by the Order (Item). |
| \_orderItemMessage | OrderMessage | 0..\* | R | An optional array of messages associated with the Order Item. |
| \_orderItemRelationship | OrderItemRelationship | 0..1 | RW | Reference to an related Service Order Item and the type of association. The type of related Service Order Item, can be : "dependency" if this Order Item needs to be "not started" until another order item is complete, "relies-on" if this Order Item relies on the related order item or simply "cross-ref" to keep track of the source Order Item |

### ServiceOrderState

### ServiceRelationship

Reference to an related Service and the type of association.
The type of related Service can be : "dependency" if this Service needs to be "not started" until another Service instance is complete, "relies-on" if this Order Item relies on the related Service instance or simply "cross-ref" to keep track of the source Service instance

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| type | RelationshipType | 1 | RW |  |

### ServiceSpecification

ServiceSpecification class offers the ability to describe the characteristics of a specific type of service.
Functionally, it acts as a template by which Services may be instantiated.
By sharing the same specification, these services would therefore share the same set of characteristics.
The MEF Service classes that describe a specific type of MEF Service will be associated with this class
via a <specify> stereotype to enable generation of specification characteristics (via tooling or otherwise).

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| version | String | 1 | R | Service specification version. |
| category | String | 1 | R | The category is used to group service specifications in logical containers. Categories can be further sub-categorized into other categories. |
| subcategory | String | 1 | R | The category is used to group service specifications in logical containers. Categories can be further sub-categorized into other categories. |
| description | String | 1 | R | A narrative that explains the details of the service specification. |
| lifecycleStatus | ServiceSpecificationState | 1 | R | Used to indicate the current lifecycle status of the Service Specification. This read-only state attribute is maintained by the SOF. This state attribute value-transition follows the "Service Specification Lifecycle State Machine" diagram described elsewhere in this specification. |
| distributionStatus | DistributionStatus | 1 | RW |  |
| \_attachment | Attachment | 0..1 | R | Complements the description of a service specification |
| \_relatedParty | RelatedParty | 0..\* | R | Party linked to the Service Specification |
| \_targetServiceSchema | ApiEntityMetaData | 0..1 | RW |  |
| id | String | 1 | R | Unique identifier of an object instance of this entity class. |
| href | String | 1 | R | URI reference of an object instance of this entity class. |
| name | String | 1 | RW | Name of an object instance of this entity class. |
| @type | String | 0..1 | RW | The class type of a REST resource |
| @baseType | String | 0..1 | RW |  |
| @schemaLocation | String | 0..1 | RW | Provides a link to the schema describing the REST resource in URI format. |

### ServiceSpecificationState

### ServiceState

### SupportingResource

This class provides reference to the Resource supporting the Service instance.

| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| --- | --- | --- | --- | --- |
| role | String | 1 | RW |  |
| id | String | 1 | R | Unique identifier of an object instance of this entity class. |
| href | String | 1 | R | URI reference of an object instance of this entity class. |
| name | String | 1 | RW | Name of an object instance of this entity class. |
| @type | String | 0..1 | RW | The class type of a REST resource |
| @baseType | String | 0..1 | RW |  |
| @schemaLocation | String | 0..1 | RW | Provides a link to the schema describing the REST resource in URI format. |

## Data Types and Enumeration

### Data Type: ErrorRepresentation

Representation of an error.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute Name** | **Type** | **Mult.** | **Access** | **Description** |
| code | Integer | 1 | RW | Application related error code (as defined in the API or from a common list). |
| reason | String | 1 | RW | Text that explains the reason for error. This can be shown to a client user. |
| message | String | 0..1 | RW | Text that provide more details and corrective actions related to the error. This can be shown to a client user |
| referenceError | String | 0..1 | RW | url pointing to documentation describing the error |
| status | String | 0..1 | RW | http error code extension like 400-2 |

### Enumeration: ActionType

Contains Enumeration Literals:

* ADD:
* MODIFY:
* DELETE:
* NO\_CHANGE:

### Enumeration: DistributionStatus

Contains Enumeration Literals:

* DISTRIBUTION\_APPROVED:
* DISTRIBUTION\_NOT\_APPROVED:
* DISTRIBUTION\_REJECTED:
* DISTRIBUTED:
* RETIRED:

### Enumeration: RelationshipType

Contains Enumeration Literals:

* RELIES\_ON:
* DEPENDENCY:
* CROSS\_REF:

### Enumeration: ServiceOrderState

Contains Enumeration Literals:

* ACKNOWLEDGED:
  + Order has been received and has passed message validations and basic business validations
* IN\_PROGRESS:
  + Service fulfillment has started
* PENDING:
  + Order is currently in a waiting stage for an explicity action/activity to be completed before the order can progress further, pending order amend or cancel assessment.
    A pending stage can lead into auto cancellation of an order, if no action is taken within the defined timeframes to be described under the Agreement.
* HELD:
  + Order cannot be progressed due to an temporary issuue.
    For example, processing is temporarily delayed to resolve an infrastructure shortfall to facilitate supply of order.
    Upon resolution of the issue, the order will continue to progress
* CANCELLED:
  + In-Flight Order has been successfully cancelled
* COMPLETED:
  + Provisioning completed and the Service is now active
* FAILED:
  + All of the Order items processing have failed which results in the entire Order to be Failed
* PARTIAL:
  + Some of the Order items processing have failed and some have succeeded so the entire Order is in a Partial state.
    This provides support for partial Failure of an Order
* REJECTED:
  + An order failed the Order Feasibility checks.
    Service technical eligibility is not done though service order API but with dedicated serviceQualification API (from preOrdering domain)
    Invalid information is provided through the order request
    The order request fails to meet business rules for ordering.

### Enumeration: ServiceSpecificationState

Contains Enumeration Literals:

* NOT\_CERTIFIED\_CHECKOUT:
* NOT\_CERTIFIED\_CHECKIN:
* READY\_FOR\_CERTIFICATION:
* CERTIFICATION\_IN\_PROGRESS:
* CERTIFIED:
* REJECTED:

### Enumeration: ServiceState

Contains Enumeration Literals:

* FEASIBILITY\_CHECKED:
* DESIGNED:
* RESERVED:
* ACTIVE:
* INACTIVE:
* TERMINATED:

### Enumeration: SeverityMessage

Contains Enumeration Literals:

* INFORMATION:
* ERROR:

## Interface Operations

### ServiceCatalog::ServiceSpecification

This Interface class is intended to encapsulate details of the API operations for Service Catalog management.
The intent of this API is to provide a consistent/standardized mechanism to query the Service Catalog.
The Service Catalog API can be invoked by the Business Applications (BA) systems to update/sync its internal cache/repository with Service Catalog maintained by the Service Orchestration Function (SOF)

#### Operation: findServiceSpecification()

This operation returns a list of Service Specification instances matching the query parameters from the Service Catalog maintained by the SOF.
Attribute selection is possible using the "fields" parameter to filter retrieved attribute(s) for each Service Specification instance.
If an exception is encountered, then appropriate returnCode and error information is populated and returned as specified.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter Name** | **Type** | **Dir.** | **Multi.** | **Description** |
| category | String | in | 0..1 | Filter by Category |
| status | ServiceSpecificationState | in | 0..1 | Filter by Service Specification status |
| fields | String | in | 0..1 | fields attribute may be used to filter retrieved attribute(s) for returned Service Specification instances. Specified as string value containing a list of attribute names separated by space |
| serviceSpecification | ServiceSpecification | out | 0..\* | Return Code 200: Success. The list of Service Specification instances matching the input query parameters. If the "fields" input parameter is specified, the returned Service Specification instance will only contain the request attributes. |
| error | ErrorRepresentation | out | 0..1 | 400: Bad Request List of supported error codes: - 20: Invalid URL parameter value - 21: Missing body - 22: Invalid body - 23: Missing body field - 24: Invalid body field - 25: Missing header - 26: Invalid header value - 27: Missing query-string parameter - 28: Invalid query-string parameter value 401: Unauthorized List of supported error codes: - 40: Missing credentials - 41: Invalid credentials - 42: Expired credentials 403: Forbidden List of supported error codes: - 50: Access denied - 51: Forbidden requester - 52: Forbidden user - 53: Too many requests 404: Not Found List of supported error codes: - 60: Resource not found 422: Unprocessable entity Functional error - Specific business errors for current operation will be encapsulated in this 500: Internal Server Error List of supported error codes: - 1: Internal error 503: Service Unavailable List of supported error codes: - 5: The service is temporarily unavailable - 6: API is over capacity, retry later ! |
| returnCode | Integer | return | 1 | 200: Success 400: Bad Request 401: Unauthorized 403: Forbidden 404: Not Found 422: Unprocessable entity 500: Internal Server Error 503: Service Unavailable |

#### Operation: getServiceSpecification()

This operation retrieves a Service Specification instance identified by the Id parameter from the Service Catalog maintained by the SOF.
Attribute selection is possible using the "fields" parameter to filter retrieved attribute(s) for the returned Service Specification instance.
If an exception is encountered, then appropriate returnCode and error information is populated and returned as specified.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter Name** | **Type** | **Dir.** | **Multi.** | **Description** |
| id | String | in | 1 | Id of the specific Service Specification instance to be retrieved |
| fields | String | in | 0..1 | fields attribute may be used to filter retrieved attribute(s) for returned Service Specification instance. Specified as string value containing a list of attribute names separated by space |
| serviceSpecification | ServiceSpecification | out | 0..1 | Return Code 200: Success The Service Specification instance matching the input Id parameters. If the "fields" input parameter is specified, the returned Service instance will only contain the request attributes. |
| error | ErrorRepresentation | out | 0..1 | 400: Bad Request List of supported error codes: - 20: Invalid URL parameter value - 21: Missing body - 22: Invalid body - 23: Missing body field - 24: Invalid body field - 25: Missing header - 26: Invalid header value - 27: Missing query-string parameter - 28: Invalid query-string parameter value 401: Unauthorized List of supported error codes: - 40: Missing credentials - 41: Invalid credentials - 42: Expired credentials 403: Forbidden List of supported error codes: - 50: Access denied - 51: Forbidden requester - 52: Forbidden user - 53: Too many requests 404: Not Found List of supported error codes: - 60: Resource not found 422: Unprocessable entity Functional error - Specific business errors for current operation will be encapsulated in this 500: Internal Server Error List of supported error codes: - 1: Internal error 503: Service Unavailable List of supported error codes: - 5: The service is temporarily unavailable - 6: API is over capacity, retry later ! |
| returnCode | Integer | return | 1 | 200: Success 400: Bad Request 401: Unauthorized 403: Forbidden 404: Not Found 422: Unprocessable entity 500: Internal Server Error 503: Service Unavailable |

### ServiceInventory::Service

This Interface class is intended to encapsulate details of the API operations for Service Inventory management.
The intent of this API is to provide a consistent/standardized mechanism to query the Service inventory.
The Service Inventory API can be invoked by the Business Applications (BA) systems to update/sync its internal cache/repository with Service Inventory maintained by the Service Orchestration Function (SOF)

#### Operation: findService()

This operation returns a list of Service instances matching the query parameters from the Service Inventory maintained by the SOF.
Attribute selection is possible using the "fields" parameter to filter retrieved attribute(s) for each Service instance.
If an exception is encountered, then appropriate returnCode and error information is populated and returned as specified.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter Name** | **Type** | **Dir.** | **Multi.** | **Description** |
| id | String | in | 0..1 | Id of the specific Service instance to be retrieved. This is an optional parameter for this operation. |
| state | ServiceState | in | 0..1 | Filter by Service instance status |
| relatedParty.id | String | in | 0..1 | Id of the relatedParty associated with the Service instance. |
| serviceSpecification.id | String | in | 0..1 | Id of the associated Service Specification used to instatiate the Service. It is possible for multiple Services to be associated with a single Service Specification. |
| serviceSpecification.name | String | in | 0..1 | Name of the associated Service Specification used to instatiate the Service. It is possible for multiple Services to be associated with a single Service Specification. |
| fields | String | in | 0..1 | fields attribute may be used to filter retrieved attribute(s) for returned Service instances. Specified as string value containing a list of attribute names separated by space |
| service | Service | out | 0..\* | Return Code 200: Success. The list of Service instances matching the input query parameters. If the "fields" input parameter is specified, the returned Service instance will only contain the request attributes. |
| error | ErrorRepresentation | out | 0..1 | 400: Bad Request List of supported error codes: - 20: Invalid URL parameter value - 21: Missing body - 22: Invalid body - 23: Missing body field - 24: Invalid body field - 25: Missing header - 26: Invalid header value - 27: Missing query-string parameter - 28: Invalid query-string parameter value 401: Unauthorized List of supported error codes: - 40: Missing credentials - 41: Invalid credentials - 42: Expired credentials 403: Forbidden List of supported error codes: - 50: Access denied - 51: Forbidden requester - 52: Forbidden user - 53: Too many requests 404: Not Found List of supported error codes: - 60: Resource not found 422: Unprocessable entity Functional error - Specific business errors for current operation will be encapsulated in this 500: Internal Server Error List of supported error codes: - 1: Internal error 503: Service Unavailable List of supported error codes: - 5: The service is temporarily unavailable - 6: API is over capacity, retry later ! |
| returnCode | Integer | return | 1 | 200: Success 400: Bad Request 401: Unauthorized 403: Forbidden 404: Not Found 422: Unprocessable entity 500: Internal Server Error 503: Service Unavailable |

#### Operation: getService()

This operation retrieves a Service instance identified by the Id parameter from the Service Inventory manintained by the SOF.
Attribute selection is possible using the "fields" parameter to filter retrieved attribute(s) for the returned Service instance.
If an exception is encountered, then appropriate returnCode and error information is populated and returned as specified.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter Name** | **Type** | **Dir.** | **Multi.** | **Description** |
| id | String | in | 1 | Id of the specific Service instance to be retrieved |
| fields | String | in | 0..1 | fields attribute may be used to filter retrieved attribute(s) for returned Service instance. Specified as string value containing a list of attribute names separated by space |
| service | Service | out | 0..1 | Return Code 200: Success The Service instance matching the input Id parameters. If the "fields" input parameter is specified, the returned Service instance will only contain the request attributes. |
| error | ErrorRepresentation | out | 0..1 | 400: Bad Request List of supported error codes: - 20: Invalid URL parameter value - 21: Missing body - 22: Invalid body - 23: Missing body field - 24: Invalid body field - 25: Missing header - 26: Invalid header value - 27: Missing query-string parameter - 28: Invalid query-string parameter value 401: Unauthorized List of supported error codes: - 40: Missing credentials - 41: Invalid credentials - 42: Expired credentials 403: Forbidden List of supported error codes: - 50: Access denied - 51: Forbidden requester - 52: Forbidden user - 53: Too many requests 404: Not Found List of supported error codes: - 60: Resource not found 422: Unprocessable entity Functional error - Specific business errors for current operation will be encapsulated in this 500: Internal Server Error List of supported error codes: - 1: Internal error 503: Service Unavailable List of supported error codes: - 5: The service is temporarily unavailable - 6: API is over capacity, retry later ! |
| returnCode | Integer | return | 1 | 200: Success 400: Bad Request 401: Unauthorized 403: Forbidden 404: Not Found 422: Unprocessable entity 500: Internal Server Error 503: Service Unavailable |

### ServiceOrder::ServiceOrder

This Interface class is intended to encapsulate details of the API operations for Service Order management.
The intent of this API is to provide a consistent/standardized mechanism to manipulate the Service instances (and in turn the Service fulfillment) through Service Orders.
The Service Order API can be invoked by the Business Applications (BA) systems to initiate Service Orders that manipulate the Service fulfillment process by the Service Orchestration Function (SOF)

#### Operation: createServiceOrder()

This operation creates a Service Order instance in the Service Order management system maintained by the SOF.
A Service Order is used to request operations on a Service instance.
A Service Order groups one or more one Service Order Items - one per specific action on a Service instance.
The Action associated with the Service Order Item describes the operation (add, remove, update) to be applied on the specified Service instance.
The Service Order Item and its associated Action can operate on both existing (remove, update) as well as future (add) Service instance.
The Service Order is triggered from the Business Application (BA) system in charge of the Product Order management to the Service Orchestration Function (SOF) system that will orchestrate the Service fulfillment.
The assumption is that the Service Specifications describing the Service instance in the Service Order Item are already available (to both BA & SOF).
If an exception is encountered, then appropriate returnCode and error information is populated and returned as specified.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter Name** | **Type** | **Dir.** | **Multi.** | **Description** |
| createServiceOrder | ServiceOrder | in | 1 | The Service Order to be instatiated. The mandatory and optional attributes are specified in the Service Order / Service Order Item classes/schema. The readOnly attributes cannot be specified as will be assigned by SOF and returned in the output Service Order instance. The Service Instance attributes are to be specified and constrained as per the associated Service Specification template. |
| serviceOrder | ServiceOrder | out | 0..1 | Return Code 201: Success The created Service Order instance with Id and Href fields assigned by SOF. |
| error | ErrorRepresentation | out | 0..1 | 400: Bad Request List of supported error codes: - 20: Invalid URL parameter value - 21: Missing body - 22: Invalid body - 23: Missing body field - 24: Invalid body field - 25: Missing header - 26: Invalid header value - 27: Missing query-string parameter - 28: Invalid query-string parameter value 401: Unauthorized List of supported error codes: - 40: Missing credentials - 41: Invalid credentials - 42: Expired credentials 403: Forbidden List of supported error codes: - 50: Access denied - 51: Forbidden requester - 52: Forbidden user - 53: Too many requests 404: Not Found List of supported error codes: - 60: Resource not found 422: Unprocessable entity Functional error - Specific business errors for current operation will be encapsulated in this as follows: \*\* 100: OrderItem with 'add' action but serviceSpecification id missing \*\* 101: OrderItem with 'change'/'noChange'/'remove' but service id missing \*\* 102: OrderItem with 'add' action - serviceSpecification id provided but not existing \*\* 103: OrderItem with 'add' action but service id already existing in the inventory \*\* 104: A customer for existing service(s) is provided but he did not exist \*\* 105: OrderItem with 'change'/'noChange'/'remove' - Service id provided but it is not existing in the inventory 500: Internal Server Error List of supported error codes: - 1: Internal error 503: Service Unavailable List of supported error codes: - 5: The service is temporarily unavailable - 6: API is over capacity, retry later ! |
| returnCode | Integer | return | 1 | 200: Success 400: Bad Request 401: Unauthorized 403: Forbidden 404: Not Found 422: Unprocessable entity 500: Internal Server Error 503: Service Unavailable |

#### Operation: findServiceOrder()

This operation returns a list of Service Order instances matching the query parameters from the Service Order management system maintained by the SOF.
Attribute selection is possible using the "fields" parameter to filter retrieved attribute(s) for each Service Order instance.
If an exception is encountered, then appropriate returnCode and error information is populated and returned as specified.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter Name** | **Type** | **Dir.** | **Multi.** | **Description** |
| externalId | String | in | 0..1 | Filter by externalId |
| state | ServiceOrderState | in | 0..1 | Filter by Service Order state |
| description | String | in | 0..1 | Filter by Service Order description |
| orderData.gt | String | in | 0..1 | Filter Service Orders with orderDate greater than or equal to the specified date |
| orderDate.lt | String | in | 0..1 | Filter Service Orders with orderDate less than equal to the specified date |
| fields | String | in | 0..1 | fields attribute may be used to filter retrieved attribute(s) for returned Service Order instances. Specified as string value containing a list of attribute names separated by space |
| serviceOrder | ServiceOrder | out | 0..\* | Return Code 200: Success. The list of Service Order instances matching the input query parameters. If the "fields" input parameter is specified, the returned Service Order instance will only contain the request attributes. |
| error | ErrorRepresentation | out | 0..1 | 400: Bad Request List of supported error codes: - 20: Invalid URL parameter value - 21: Missing body - 22: Invalid body - 23: Missing body field - 24: Invalid body field - 25: Missing header - 26: Invalid header value - 27: Missing query-string parameter - 28: Invalid query-string parameter value 401: Unauthorized List of supported error codes: - 40: Missing credentials - 41: Invalid credentials - 42: Expired credentials 403: Forbidden List of supported error codes: - 50: Access denied - 51: Forbidden requester - 52: Forbidden user - 53: Too many requests 404: Not Found List of supported error codes: - 60: Resource not found 422: Unprocessable entity Functional error - Specific business errors for current operation will be encapsulated in this 500: Internal Server Error List of supported error codes: - 1: Internal error 503: Service Unavailable List of supported error codes: - 5: The service is temporarily unavailable - 6: API is over capacity, retry later ! |
| returnCode | Integer | return | 1 | 200: Success 400: Bad Request 401: Unauthorized 403: Forbidden 404: Not Found 422: Unprocessable entity 500: Internal Server Error 503: Service Unavailable |

#### Operation: getServiceOrder()

This operation retrieves a Service Order instance identified by the Id parameter from the Service Order managment system maintained by the SOF.
Attribute selection is possible using the "fields" parameter to filter retrieved attribute(s) for the returned Service Order instance.
If an exception is encountered, then appropriate returnCode and error information is populated and returned as specified.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Parameter Name** | **Type** | **Dir.** | **Multi.** | **Description** |
| id | String | in | 1 | Id of the specific Service Order instance to be retrieved |
| fields | String | in | 0..1 | fields attribute may be used to filter retrieved attribute(s) for returned Service Order instance. Specified as string value containing a list of attribute names separated by space |
| serviceOrder | ServiceOrder | out | 0..1 | Return Code 200: Success The Service Order instance matching the input Id parameters. If the "fields" input parameter is specified, the returned Service instance will only contain the request attributes. |
| error | ErrorRepresentation | out | 0..1 | 400: Bad Request List of supported error codes: - 20: Invalid URL parameter value - 21: Missing body - 22: Invalid body - 23: Missing body field - 24: Invalid body field - 25: Missing header - 26: Invalid header value - 27: Missing query-string parameter - 28: Invalid query-string parameter value 401: Unauthorized List of supported error codes: - 40: Missing credentials - 41: Invalid credentials - 42: Expired credentials 403: Forbidden List of supported error codes: - 50: Access denied - 51: Forbidden requester - 52: Forbidden user - 53: Too many requests 404: Not Found List of supported error codes: - 60: Resource not found 422: Unprocessable entity Functional error - Specific business errors for current operation will be encapsulated in this 500: Internal Server Error List of supported error codes: - 1: Internal error 503: Service Unavailable List of supported error codes: - 5: The service is temporarily unavailable - 6: API is over capacity, retry later ! |
| returnCode | Integer | return | 1 | 200: Success 400: Bad Request 401: Unauthorized 403: Forbidden 404: Not Found 422: Unprocessable entity 500: Internal Server Error 503: Service Unavailable |

# MEF Service Common Model (MSCM) Mapping

# References

1. IETF RFC 2119, *Key words for use in RFCs to Indicate Requirement Levels*, March 1997
2. IETF RFC 8174, *Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words*, May 2017

1. Appendix Title (Informative)

Appendix content.

* 1. MEF Access E-Line Service Specification

More content

* + 1. Appendix second-level subsection

Yet more content.