Service Layer API for oneM2M

Draft

105 Pages

Text in Red is here to help you. Delete it when you have followed the instructions. The <RFC Title> can be set from the File>Properties:User Defined menu. To update it onscreen, press F9. To update all of the fields in the document Select All (CTRL-A), then hit F9. Set the release level by selecting one from: Draft, Final Draft, Release. The date is set automatically when the document is saved.

Abstract

10 point Arial Centered.

oneM2M is standard organization and specifies middleware for IoT, called Common Services Entities (CSE). Application can access functionality in CSE with RESTful operations, which are Create, Retrieve, Update, Delete and Notify. oneM2M allows variety of communication methods, 4 protocol bindings (HTTP, MQTT, CoAP, Websocket) and 3 serializations (XML, JSON, CBOR). This RFC describes the way to provide high level API for oneM2M RESTful operations hiding the difference of variety of communication methods.

0 Document Information

0.1 License

DISTRIBUTION AND FEEDBACK LICENSE, Version 2.0

0.2 Trademarks

OSGi™ is a trademark, registered trademark, or service mark of the OSGi Alliance in the US and other countries. Java is a trademark, registered trademark, or service mark of Oracle Corporation in the US and other countries. All other trademarks, registered trademarks, or service marks used in this document are the property of their respective owners and are hereby recognized.

0.3 Feedback

This document can be downloaded from the OSGi Alliance design repository at https://github.com/osgi/design The public can provide feedback about this document by opening a bug at https://www.osgi.org/bugzilla/.

0.4 Table of Contents

0.5 Terminology and Document Conventions

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 10.1.

```
Source code is shown in this typeface.
```

0.6 Revision History

The last named individual in this history is currently responsible for this document.

Revision	Date	Comments
Initial	*SEP 15 2017*	Initial Contribution. Hiroyuki Maeomichi, NTT, maeomichi.hiroyuki@lab.ntt.co.jp
0.0.1	*SEP 21 2017*	Updated alternatives, some figures, added description on validator. Hiroyuki Maeomichi, NTT, maeomichi.hiroyuki@lab.ntt.co.jp

Revision	Date	Comments
0.0.2	*April 17 2018*	Update based on discussion in Washington meeting. Hiroyuki Maeomichi, NTT, maeomichi.hiroyuki@lab.ntt.co.jp
0.0.3	*June 22 2018*	Add new fields and class reflecting R3 draft of oneM2M: Added fields in RequestPrimitiveDTO, ResponsePrimitiveDTO, and FilterCriteriaDTO, and ReleaseVersion enum. Organize DTOs: Added AttributeDTO, LocalIdTokenIdAssigmentDTO, and DasInfoDTO and remove DynAuthLocalIdAssignmentsDTO and DynAuthReqInfoDTO Introduce OperationIF interface as a super interface of ProtocolBinding interface and CSE interface for enabling concise application code. This replaces former simple.Client. Organize Introspection interfaces with less methods. They are moved to dedicated package.

1 Introduction

Introduce the RFC. Discuss the origins and status of the RFC and list any open items to do.

oneM2M is standard organization and specifies middleware for Internet of Things (IoT), called Common Services Entities (CSE). Applications can access CSE's functionality with RESTful operations, which are Create, Retrieve, Update, Delete and Notify. TS-0001 [2] defines more than 40 resource types to expose CSE's functionalities. oneM2M allows variety of communication methods, combination of 4 protocol bindings (HTTP, MQTT, CoAP, Websocket) and 3 serializations (XML, JSON, CBOR).

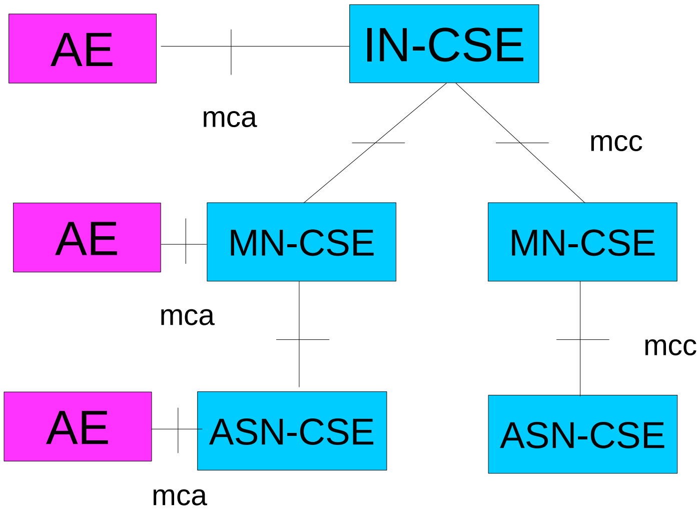
This RFP discuss the way to provide high level API (namely service layer API) for oneM2M RESTful operations hiding the difference of variety of communication methods.

2 Application Domain

This section should be copied from the appropriate RFP(s). It is repeated here so it can be extended while the RFC authors learn more subtle details.

2.1 IoT Application configuration using oneM2M

oneM2M's middleware, called CSE can be deployed in different locations and they are connected each other forming tree topology. Depending on deployed location, CSEs are categorized to 3 types, IN-CSE, MN-CSE and ASN-CSE. IN-CSE is located top of tree, ASN-CSE is located at leaf and MN-CSE is located and MN-CSE is located on middle.

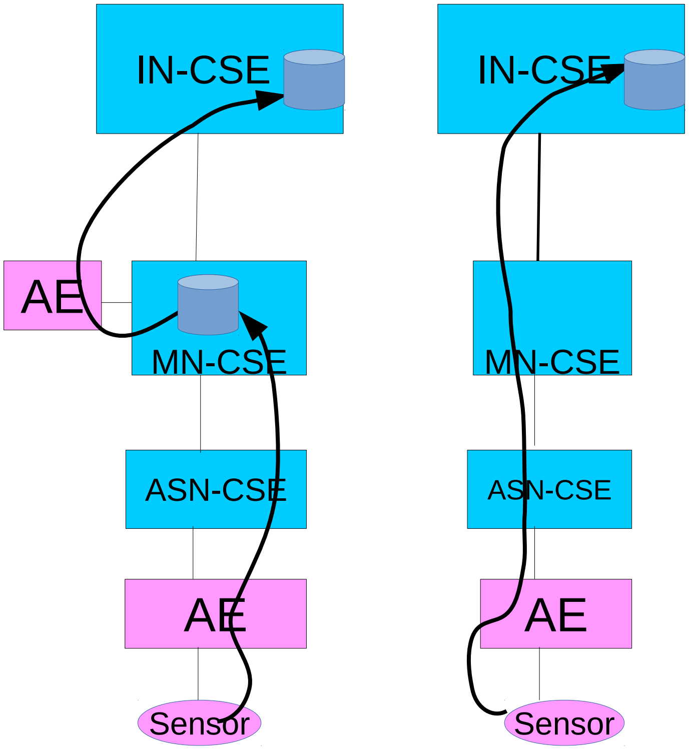


oneM2M's application, called Application Entity (AE) connects to one of CSEs. After AE connecting to the CSE, AE can access to all of CSEs, by retargeting function (similar to routing) of CSEs.

AE accesses to CSE's functionality through RESTful API, which consists of Create, Retrieve, Update, Delete and Notify in targeting more than 40 types of resources. For examples, typical resources are <*contentInstance*> that expresses IoT data and <*container*> that holds set of <*contentInstance*>s. AE can create or retrieve the <contentInstance> on any CSE by the retargeting functionality, as far as permission is allowed. Interface between CSEs is called *mcc* and interface between CSE and AE is called *mca,* both interfaces have almost same interface.

It is possible to develop variety types of distributed applications using the architecture. For example for IoT data aggregation applications, it is possible to develop gradual aggregation type or direct aggregation type. In gradual aggregation type, AE connected to ASN-CSE creates <*conentInstance*>s in ASN-CSE, and intermediate applications calculate statistics and put the result on IN-CSE as a <*contentInstance*>, while, in direct aggregation type, AE connected to ASN-CSE creates <*contentInstance*>s in IN-CSE directly.

Under CSE layer, oneM2M specifies NSE(Network Services Entity), but this RFC doesn't cover the NSE layer.



2.2 Communication methods used in oneM2M

oneM2M allows variety of communication methods, combination of 4 protocol bindings (HTTP, MQTT, CoAP, Websocket) and 3 serializations (XML, JSON, CBOR). It might be added in future. oneM2M specifies specification in different level.

Firstly TS-0001[2] specifies high level resource definitions, it defines more than 40 resource types, such as <contentInstance> for storing IoT data, <timeSeriesInstance> for periodic sensor measurement with leap detection mechanism.

Secondly TS-0004[3] specifies procedures and serializations in independent manner from protocol bindings. Resource type and protocol data unit are defined using XSD for XML serialization. Mapping between XML and other serializations are also specified.

Thirdly TS-0008, TS-0009, TS-0010, TS-0020 specify protocol specific details for CoAP, HTTP, MQTT and Web Socket respectively.

2.3 Long name and short name

oneM2M introduced two types of notation, called long name and short name for resource types, attribute and so on. Long name is human friendly string and specifications mainly use this notation, while short name is short string consist of typically 2 or 3 characters (but not limited and sometimes longer) and communication protocol use this notation. In most cases, the initial characters of long name are assigned as short name, for examples, ct for CreationTime and at for AnnounceTo.

3 Problem Description

This section should be copied from the appropriate RFP(s). It is repeated here so it can be extended while the RFC authors learn more subtle details.

oneM2M specifies protocol based interface, but doesn't specify programing level API. As previously mentioned oneM2M allows variety of communication methods which are the combinations of 4 protocol bindings (HTTP, MQTT, CoAP, Websocket) and 3 serializations (XML, JSON, CBOR).

First problem is application portability. Without standardized API, application program tends to depend on the communication method initially intend to use and it will became hard to run another environment in which uses another communication method. (For example, an application designed for XML/HTTP, tend to run on environment use JSON/Websocket)

Second problem is the latency of the communication between CSE and application. Even if CSE and application is located in the same box, current oneM2M specifications define methods through protocols which requires serialization/deserialization of data, context-switch of applications, validation of incoming data and resulted in large latency compared to the situation both CSE and Application resides in the same Java VM and communicate with Java interfaces. Large latency reduces applicable area of oneM2M based solution.

Third problem is the complexity of handling of long name and short name. Even if short name is defined by trying to use initial characters, it is not straight forward to translate them in head.

4 Requirements

This section should be copied from the appropriate RFP(s)

- R0010 – The solution MUST provide means to access outer CSE from application.
- R0011 – The solution MUST provide means to access outer CSE from client CSE.
- R0012 – The solution MUST provide means to select a communication method for application.
- R0013 – The solution MUST provide means to select a communication method for client CSE.
- R0020 – The solution MUST provide means for CSE to accept requests form outer CSE.
- R0020 – The solution MUST provide means for CSE to accept requests form outer application.

- R0030 – The solution MUST provide means to communicate through Java interface between CSE and application that are located in the same OSGi framework.
- R0040 – The solution SHOULD hide differences of communication methods, which are combinations of 4 protocol bindings and 3 serializations (XML, JSON, CBOR).
- R0050 – The solution SHOULD provide developer friendly way for handling short names.
- R0060 – The solution MUST provide asynchronous interface using ‘call by value’, such as DTO.

5 Technical Solution

First give an architectural overview of the solution so the reader is gently introduced in the solution (Javadoc is not considered gently). What are the different modules? How do the modules relate? How do they interact? Where do they come from? This section should contain a class diagram. Then describe the different modules in detail. This should contain descriptions, Java code, UML class diagrams, state diagrams and interaction diagrams. This section should be sufficient to implement the solution assuming a skilled person.

Strictly use the terminology a defined in the Problem Context.

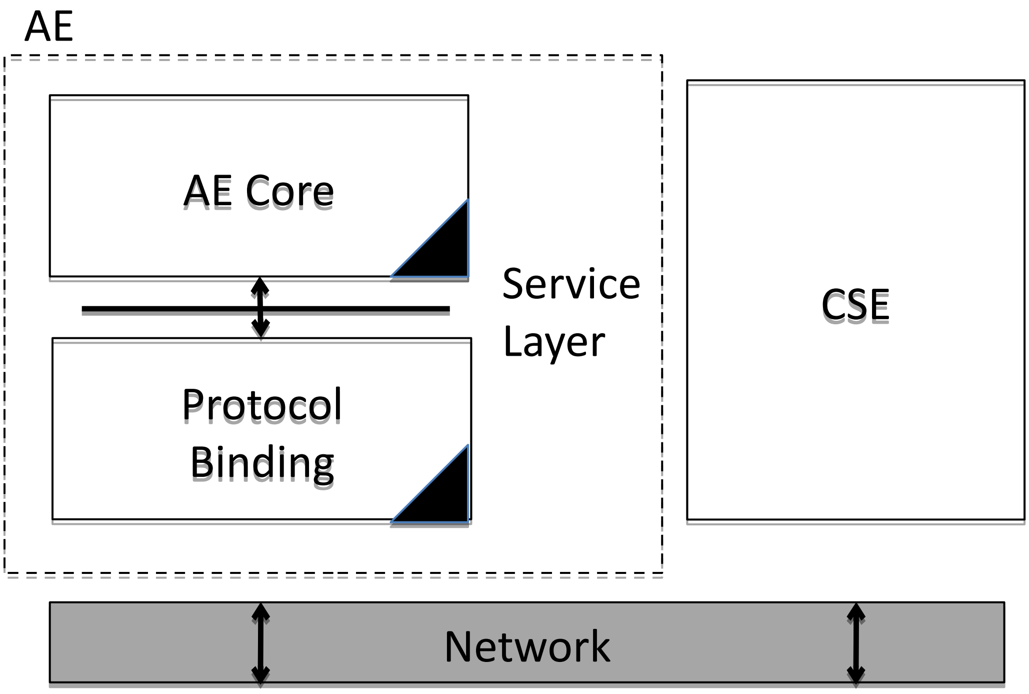
On each level, list the limitations of the solutions and any rationales for design decisions. Almost every decision is a trade off so explain what those trade offs are and why a specific trade off is made.

Address what security mechanisms are implemented and how they should be used.

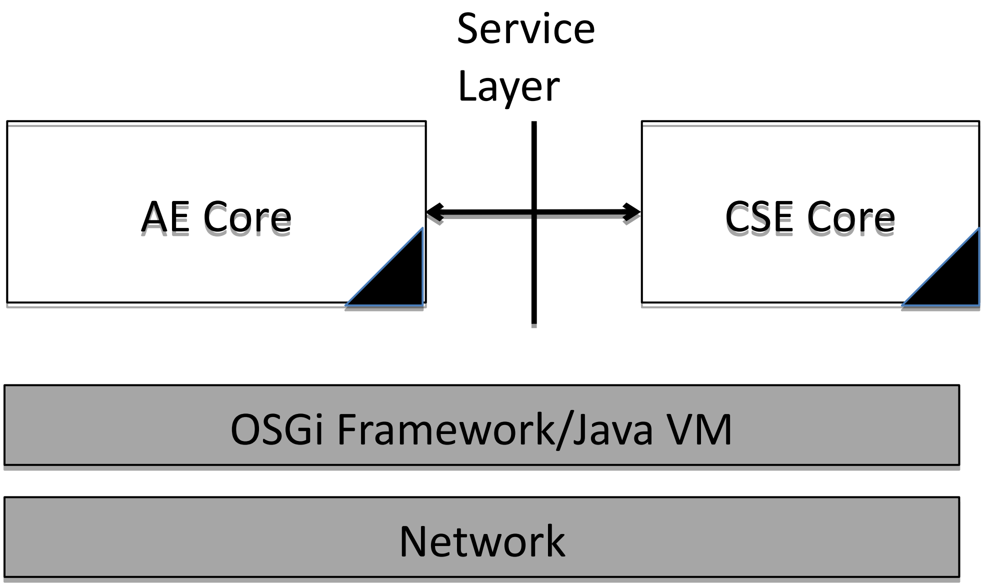
5.1 Overview for the solution

Protocol binding service is introduced to handle different protocols and serializations. oneM2M application uses the protocol binding service through Service Layer Interface to communicate CSE. The interface is protocol and serialization agnostic interface; it has no protocol and serialization specific methods, parameters, so that application can communicate to CSE without knowing which protocol is actually used.

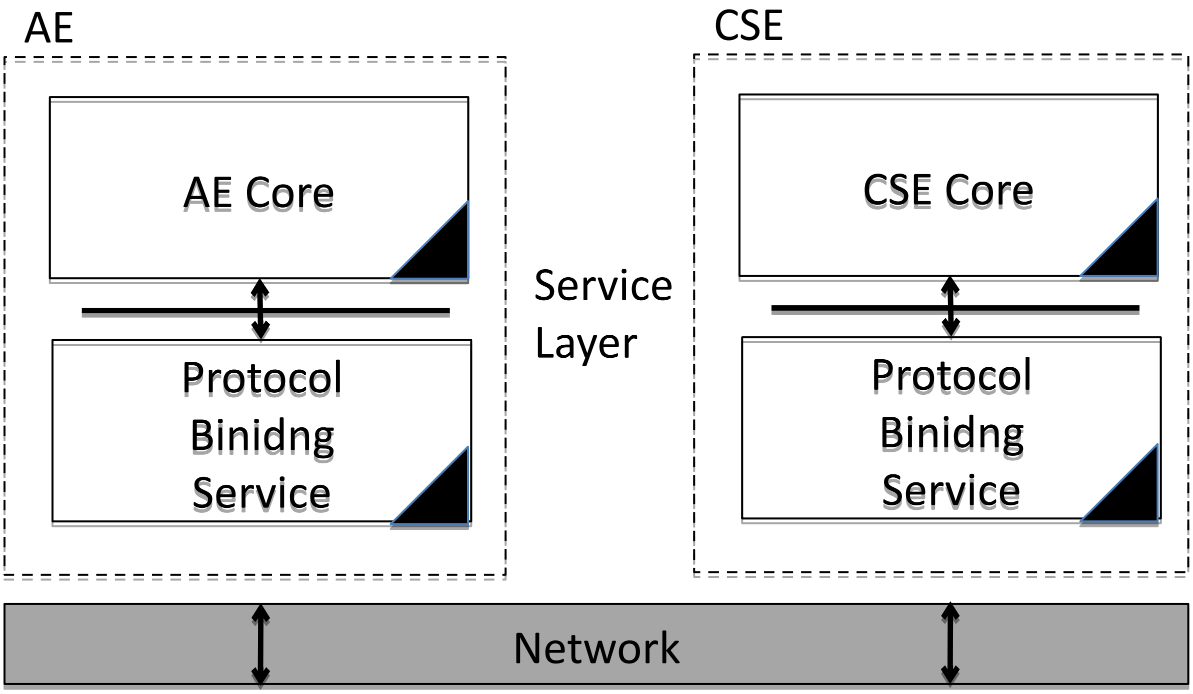
In figure, the term of Core is introduced for AE Core and CSE Core. This is for specifying parts which does not contain ProtocolBindiing of AE and CSE.



Another use case is that the AE Core and the CSE core are located on the same OSGi Framework. In this use case, the AE Core and the CSE core communicate directly with API, without inter-mediating ProtocolBinding Services. Following figure depicts overall configuration. Though this type of communication is not clearly defined in oneM2M specification, communicating directly without serializing data between AE and CSE allows shorter latency and less computational resources.



Following figure (right hand side) shows potential implementation of CSE, which are consisted fo CSE core and Protocol Binding Service, as symmetric to AE side. APIs defined in this RFC is consistent with the usage, but this RFC does not mandate that usage and it is left to implementor’s choice.



5.2 Service Layer Interfaces

Service Layer Interface is defined as follows. Only method request sends request message and return Promise for the response. Here, Promise enables asynchronous messaging.

```
package org.osgi.service.onem2m.servicelayer;

import org.osgi.onem2m.dto.RequestDTO;
import org.osgi.onem2m.dto.ResponseDTO;
import org.osgi.util.promise.Promise;

/**
 * Service Layer Interface, which locates between AE and Protocol Binding Service.
 */
public interface ServiceLayer {
	/**
	 * send a request.
```

```
         *
         * @param request request
         * @return promise for ResponseDTO.
         */
        Promise<ResponseDTO> request(RequestDTO request);
}
```

On the Service Layer interface, there are bidirectional invocations, that is, Both AE can be caller and callee of the interface. Typical and only case of AE acting as callee, is to receive notifications.

5.3 OperationIF interfaces

Though Service Layer Interface enables all possible message exchange among oneM2M entities, it can be redundant to application developers, because they are required to write composition of requestPrimitive and decomposition of responsePrimitive. This interface is provided for application developer allowing to make less application codes. It provides higher level of abstraction; operation level of resource such as create, retrieve, update, delete and so on. They don't cover all of possible message exchange but do typical ones.

Note: If this RFC doesn't provide these methods, developers likely to create similar ones in their own (various) way.

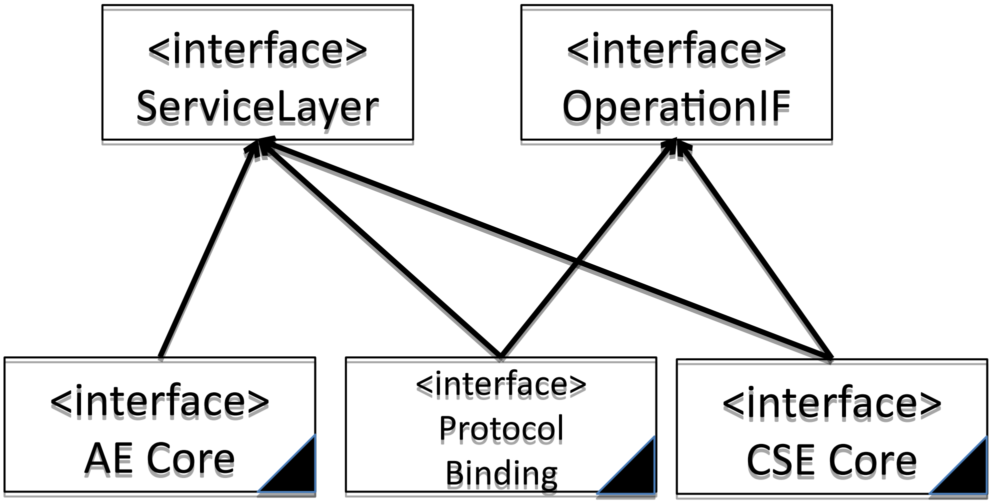
```
public interface OperationIF {
        /**
         * create resource
         *
         * @param uri URI for parent resource
         * @param resource resource data
         * @return Promise of created resource
         */
        public Promise<ResourceDTO> create(String uri, ResourceDTO resource);

        /**
         * retrieve resource
         *
         * @param uri URI for retrieving resource
         * @return retrieved resource data
         */
        public Promise<ResourceDTO> retrieve(String uri, ResourceDTO resource);

        /**
         * retrieve subset of attributes.
         *
         * @param uri URI for retrieving resource
         * @param targetNames attribute names for retrival
         * @return retrieved resource data
         */
```

```
      public Promise<ResourceDTO> retrieve(String uri, List<String> 
targetAttributes);

      /**
       * update resource
       * 
       * @param uri URI for updating resource
       * @param resource data resource
       * @return updated resource
       */
      public Promise<ResourceDTO> update(String uri, ResourceDTO resource);

      /**
       * delete resource
       * 
       * @param uri target URI for deleting resource
       */
      public Promise<Boolean> delete(String uri);

      /**
       * find resources
       * 
       * @param uri URI for top of search
       * @param fc filter criteria
       * @return list of URIs matching the condition specified in fc
       */
      public Promise<List<String>> discovery(String uri, FilterCriteriaDTO fc);

      /**
       * send notification
       * 
       * @param notification
       */
      public Promise<Boolean> notify(String uri, NotificationDTO notification );

}
```

5.4 Concrete Interfaces

The concrete servicess that implements Service Layer interfaces are AE Core, CSE Core and Protocol Bindings. For clarification purpose, These interfaces are separately defined as a sub interface of Service Layer Interface. OperationIF is also provided by CSE Core and Protocol Bindings, which are used by AE Core.

```
public interface ProtocolBinding extends ServiceLayer, OperationIF {}
public interface Cse extends ServiceLayer, OperationIF {}
public interface Ae extends ServiceLayer {}
```



5.5 Service Property for sub-interfaces of Service Layer Interface

Services implementing Service Layer Interface shall be registered with following properties.

Interface	property Name	type	explanation
ProtocolBinidng	protocol	org.osgi.service.onem2m .ProtocolBindingType	Supporting protocol.
	serialization	org.osgi.service.onem2m .SerializationType	Serialization Type
	secure	boolean	True, if secure protocol is supported, otherwise false.
	version	org.osgi.service.onem2m .dto.ReleaseVersion	Supported version
	APP-ID	String	Indicates which Application can use this service.
	POA	String[]	URIs for point of access
CSECore	CSE-ID	String	CSE-ID: ID of CSE
	SP-ID	String	ID of Service Provider
	CSE-type	org.osgi.service.onem2m .CSEType	Type of CSE. Possible values are IN, MN, or ASN
	version	org.osgi.service.onem2m .dto.ReleaseVersion	Supported version
AECore	AE-ID	String	ID of Application Entity

	APP-ID	String	Application ID
	version	org.osgi.service.onem2m .dto.ReleaseVersion	Supported version

Here, AE-ID is assigned by CSE. After receiving assigned value, AE Core should update the property.

5.6 Introspection Interface

5.6.1 General Introspection Interface

This interface provides information for data structures and checks validity of data. For oneM2M resources, there are 3 types of attributes, which are mandatory, optional and NP(Not present) and they are different between create operation and update operation. So that there are getAttributes()'s variations for getting these types as following.

```
public String[] get{OPTIONALITY}AttributesFor{OPERATION}(int resourceType);
```

here, {OPTIONALITY} is one of Mandatory, Optional or NotPresent and {OPERATION} is Create or Update.

```
package org.osgi.service.onem2m;
import org.osgi.service.onem2m.dto.*;

public interface Introspector {

      /**
       * execute Validation of Data Structure
       *
       * @param resp
       * @return array of problems
       */
      public String[] findValidationProblems(ResponsePrimitiveDTO resp);

      /**
       * execute Validation of Data Structure
       *
       * @param resp
       * @return array of problems
       */
      public String[] findValidationProblems(RequestPrimitiveDTO req);

      /**
       * execute Validation of Data Structure
       *
       * @param resp
```

```
     * @return array of problems
     */
    public String[] findValidationProblems(ResourceDTO resource);

    /**
     * get Possible Attribute Names for the given resouceType.
     *
     * @param resp
     * @return array of Possible Attributes
     */
    public String[] getAttributeNames(int resourceType);

    /**
     * get Mandatory Attributes for the given resouceType when Create.
     *
     * @param resp
     * @return array of Possible Attributes
     */
    public String[] getMandatoryAttributesForCreate(int resourceType);

    /**
     * get Optional Attributes for the given resouceType when Create.
     *
     * @param resp
     * @return array of Possible Attributes
     */
    public String[] getOptionalAttributesForCreate(int resourceType);

    /**
     * get Not Present Attributes for the given resouceType when Create.
     *
     * @param resp
     * @return array of Possible Attributes
     */
    public String[] getNotPresentAttributesForCreate(int resourceType);


    /**
     * get Mandatory Attributes for the given resouceType when Update.
     *
     * @param resp
     * @return array of Possible Attributes
     */
    public String[] getMandatoryAttributesForUpdate(int resourceType);

    /**
     * get Optional Attributes for the given resourceType when Update.
```

```
         *
         * @param resp
         * @return array of Possible Attributes
         */
        public String[] getOptionalAttributesForUpdate(int resourceType);

        /**
         * get Not Present Attributes for the given resouceType when Update.
         *
         * @param resp
         * @return array of Possible Attributes
         */
        public String[] getNotPresentAttributesForUpdate(int resourceType);

        /**
         * return Java Type for given attribute of resource type
         * @param resouceType
         * @param attribute
         * @return expected class for the specified attribute
         */
        public Class getType(int resouceType, String Attribute);

        /**
         * return Typical Data Structure for Type for given attribute of resource type
         * @param resouceType
         * @param attribute
         * @return Template Object
         */
        public Object getTemplateObject(int resouceType, String Attribute);

}
```

5.6.2 Introspection interface for FlexContainer

FlexContainer is a oneM2M Resource type that can be defined with adding custom attributes. Initially it was intended to be like JSON structure in oneM2M world. The definition of custom flex container is expressed in ‘contentDefinition’ attribute of the FlexContainer. It may be one of standardized URI or URI for location of XSD definitions; The XSD is stored as contentInstance resource in some CSE.

This interface checks validity of data structure representing <FlexContainer> resource. Following table shows service properties on the service. Being different from regular oneM2M resource, it has no

Property Name	Type	explanation
contentDefinitions	String[]	Supporting contentDefinition of <FlexContainer>. .

```
package org.osgi.service.onem2m;
```

```java
import org.osgi.service.onem2m.dto.*;

/**
 * FlexContainerInspector
 *
 *
 */
public interface FlexContainerIntrospector {

	/**
	 * Execute Validation of Data Structure
	 *
	 * @param resp
	 * @return array of problems
	 */
	public String[] findValidationProblems(ResourceDTO resource);

	/**
	 * get Possible Attributes for the given resouceType.
	 *
	 * @param resp
	 * @return array of Possible Attributes
	 */
	public String[] getCustomeAttributeNames(String containerDefinition);

	/**
	 * return Java Type for given attribute of resource type
	 *
	 * @param resourceType
	 * @param attribute
	 * @return expected class for the specified attribute
	 */
	public Class getType(String containerDefinition, String customAttributeName);

	/**
	 * return Typical Data Structure for Type for given attribute of resource type
	 *
	 * @param resourceType
	 * @param attribute
	 * @return Template Object
	 */
	public Object getTemplateObject(String containerDefinition, String 
customAttributeName);

}
```

6 Data Transfer Objects

RFC 185 defines Data Transfer Objects as a generic means for management solutions to interact with runtime entities in an OSGi Framework. DTOs provides a common, easily serializable representation of the technology.

For all new functionality added to the OSGi Framework the question should be asked: would this feature benefit from a DTO? The expectation is that in most cases it would.

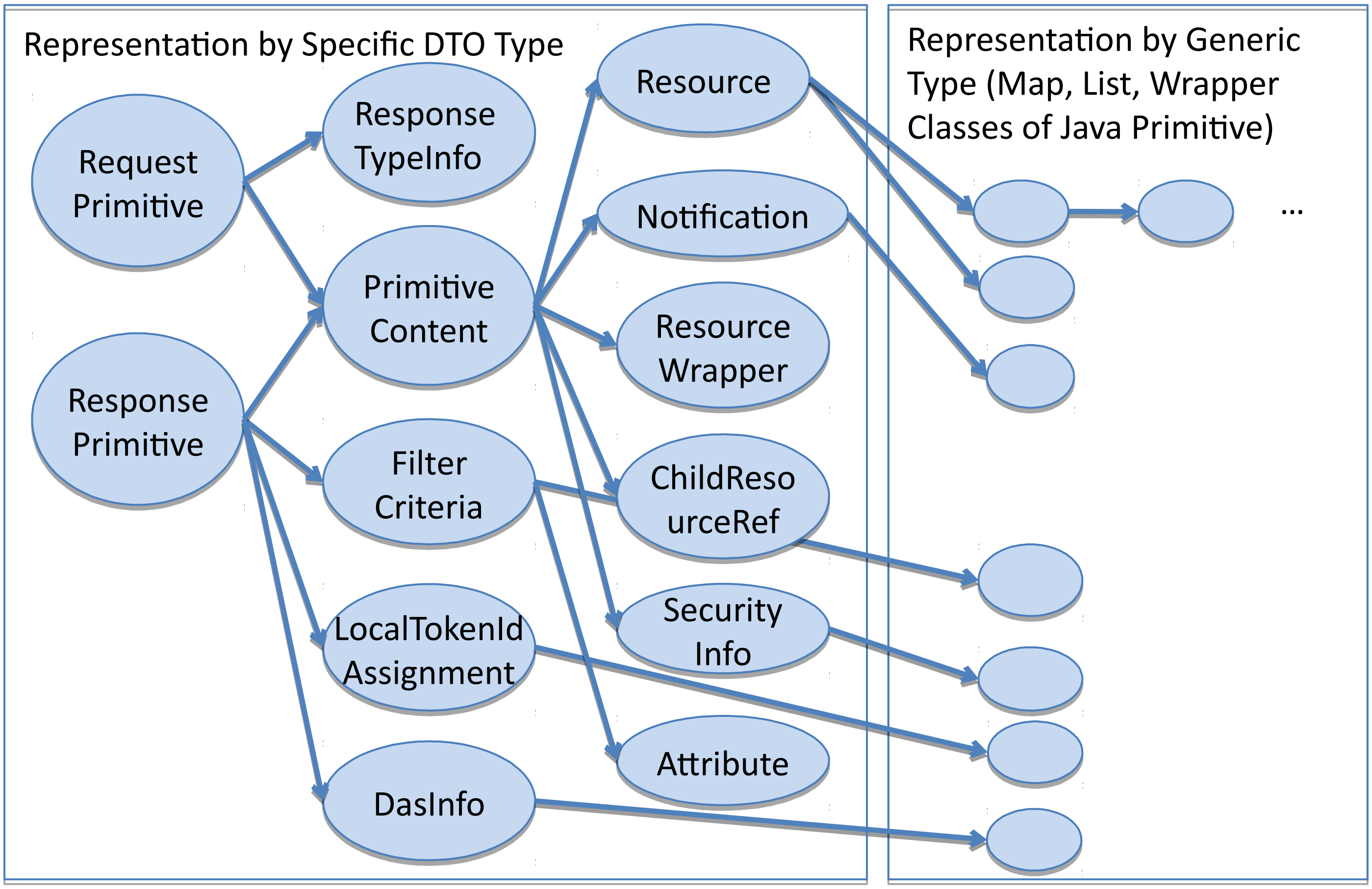
The DTOs for the design in this RFC should be described here and if there are no DTOs being defined an explanation should be given explaining why this is not applicable in this case.

This section is optional and could also be provided in a separate RFC.

6.1 Design Policy of DTOs

Data transfer Object was chosen as data object passing by the interfaces, following OSGi's design convention. Though data structure near root structures are designed specific DTO, deeper data types are to be stored as generic types, such as Map, List and Wrapper classes of Java Primitives. (See also the considered Alternatives)

In the object definition, some data types are shown as Object, but the assigned value shall be a type that is allowed for OSGi DTO.



6.2 RequestPrimitiveDTO

RequestPrimitiveDTO holds a Request Information used for oneM2M communication.

```
package org.osgi.service.onem2m.dto;

import java.util.*;

public class RequestPrimitiveDTO extends org.osgi.dto.DTO {
	@javax.xml.bind.annotation.XmlElement(required = true)
	public Operation operation;
	@javax.xml.bind.annotation.XmlElement(required = true)
	public String to;
	public String from;
	@javax.xml.bind.annotation.XmlElement(required = true)
	public String requestIdentifier;
	@javax.xml.bind.annotation.XmlElement(required = false)
	public Integer resourceType;
	public PrimitiveContentDTO content;
	public List<String> roleIDs;
```

```
    public String originatingTimestamp;
    public String requestExpirationTimestamp;
    public String resultExpirationTimestamp;
    public String operationExecutionTime;

    public ResponseTypeInfoDTO responseType;
    public String resultPersistence;
    @javax.xml.bind.annotation.XmlElement(required = false)
    public ResultContent resultContent;
    public String eventCategory;
    @javax.xml.bind.annotation.XmlElement(required = false)
    public Boolean deliveryAggregation;
    public String groupRequestIdentifier;
    public FilterCriteriaDTO filterCriteria;
    @javax.xml.bind.annotation.XmlElement(required = false)
    public DiscoveryResultType discoveryResultType;
    public String tokens;
    public List<String> tokenIDs;
    public List<String> localTokenIDs;
    @javax.xml.bind.annotation.XmlElement(required = false)
    public Boolean tokenReqIndicator;

    // Added at R3.0
    public List<String> groupRequestTargetMembers;
    public Boolean authSignatureIndicator;
    public List<String> authSignature;
    public Boolean authRelationshipIndicator;
    public Boolean semanticQueryIndicator;
    public ReleaseVersion releaseVersion;
    public String verndorInformation;

    public static enum DiscoveryResultType {
        structured(1), unstructured(2);
        // omitted
    }

    public static enum ResultContent {
        nothing(1), attributes(2), hierarchicalAddress(3),
        hierarchicalAddressAndAttributes(4),
attributesAndChildResources(5),
attributesAndChildResourceReferences(6),
childResourceReferences(7), originalResource(8), childResources(9);

        // omitted
    }
    public static enum Operation {
        Create(1), Retrieve(2), Update(3), Delete(4), Notify(5);
        // omitted
    }

}
```

6.3 ResponsePrimitiveDTO

ResponsePrimitiveDTO holds a Response Information used for oneM2M communication.

```
package org.osgi.service.onem2m.dto;
import java.util.*;

public class ResponsePrimitiveDTO extends org.osgi.dto.DTO{
    @javax.xml.bind.annotation.XmlElement( required  = true)
    public Integer responseStatusCode;
    @javax.xml.bind.annotation.XmlElement( required  = true)
    public String requestIdentifier;
    public PrimitiveContentDTO content;
    public String to;
    public String from;
    public String originatingTimestamp;
    public String resultExpirationTimestamp;
    public String eventCategory;
    @javax.xml.bind.annotation.XmlElement( required  = false)
    public ContentStatus contentStatus;
    @javax.xml.bind.annotation.XmlElement( required  = false)
    public Integer contentOffset;
    public List<LocalTokenIdAssignmentDTO>
assignedTokenIdentifiers;//Map<String,Object>
    public List<DasInfoDTO> tokenReqInfo;//DynAuthTokenReqInfoDTO

    // Added R3.0
    public Boolean AuthSignatureReqInfo;
    public ReleaseVersion releaseVersionIndicator;
    public String vendorInformation;

    public static enum ContentStatus{
        PARTIAL_CONTENT, // 1
        FULL_CONTENT;  //2

    }
}
```

6.4 ResponseTypeInfoDTO

```
package org.osgi.service.onem2m.dto;

import java.util.*;

public class ResponseTypeInfoDTO extends org.osgi.dto.DTO {
      @javax.xml.bind.annotation.XmlElement(required = true)
      public java.lang.Integer responseTypeValue;
      @javax.xml.bind.annotation.XmlElement(required = true)
      public List<java.lang.String> notificationURI;

      public static enum ResponseType {
            nonBlockingRequestSynch(1), nonBlockingRequestAsynch(2),
blockingRequest(3), flexBlocking(4);
            // omitted.
      }
}
```

6.5 FilterCriteriaDTO

```
package org.osgi.service.onem2m.dto;
import java.util.*;

public class FilterCriteriaDTO extends org.osgi.dto.DTO{
      public String createdBefore;
      public String createdAfter;
      public String modifiedSince;
      public String unmodifiedSince;
      @javax.xml.bind.annotation.XmlElement( required  = false)
      public Integer stateTagSmaller;
      @javax.xml.bind.annotation.XmlElement( required  = false)
      public Integer stateTagBigger;
      public String expireBefore;
      public String expireAfter;
      public List<String> labels;
      public List<Integer> resourceType;
      @javax.xml.bind.annotation.XmlElement( required  = false)
      public Integer sizeAbove;
      @javax.xml.bind.annotation.XmlElement( required  = false)
      public Integer sizeBelow;
      public List<String> contentType;
      public AttributeDTO attribute;
```

```
    @javax.xml.bind.annotation.XmlElement( required  = false)
    public FilterUsage filterUsage;
    @javax.xml.bind.annotation.XmlElement( required  = false)
    public Integer limit;
    public String semanticsFilter;
    @javax.xml.bind.annotation.XmlElement( required  = false)
    public FilterOperation filterOperation;
    @javax.xml.bind.annotation.XmlElement( required  = false)
    public Integer contentFilterSyntax;
    public String contentFilterQuery;
    @javax.xml.bind.annotation.XmlElement( required  = false)
    public Integer level;
    @javax.xml.bind.annotation.XmlElement( required  = false)
    public Integer offset;

    // added in R3
    public List<String> childLabels;
    public List<String> parentLabels;
    public String labelsQuery;
    public Integer childResourceType;
    public Integer parentResourceType;
    public AttributeDTO childAttribute;
    public AttributeDTO parentAttribute;
    public String applyRelativePath;


    public static enum FilterOperation {
        AND(1), OR(2);
        // omitted...
    }

    public static enum FilterUsage {
        DiscoveryCriteria(1), ConditionalRetrival(2), IPEOndemandDiscovery(3);
        // omitted...

    }
}
```

6.6 ResourceDTO

```
package org.osgi.service.onem2m.dto;
import java.util.*;

public class ResourceDTO extends org.osgi.dto.DTO{
	// Universal Attribute, which can be held by all resources.
	@javax.xml.bind.annotation.XmlElement( required  = true)
	public Integer resourceType;
	@javax.xml.bind.annotation.XmlElement( required  = true)
	public String resourceID;
	@javax.xml.bind.annotation.XmlElement( required  = true)
	public String parentID;
	@javax.xml.bind.annotation.XmlElement( required  = true)
	public String creationTime;
	@javax.xml.bind.annotation.XmlElement( required  = true)
	public String lastModifiedTime;

	public String resourceName;

	// optional, Universal Attributes
	public List<String> labels;

	/**
	 * Non Universal Attribute.
	 * Value Part must be the types that are allowed for OSGi DTO.
	 */
	public Map<String, Object> attribute;
}
```

6.7 NotificationDTO

NotificationDTO has information of notification.

```
package org.osgi.service.onem2m.dto;
import java.util.*;

public class NotificationDTO extends org.osgi.dto.DTO{
	public Map<String,Object> notificationEvent;//NotificationEventDTO
	@javax.xml.bind.annotation.XmlElement( required  = false)
	public Boolean verificationRequest;
	@javax.xml.bind.annotation.XmlElement( required  = false)
	public Boolean subscriptionDeletion;
	public String subscriptionReference;
	public String creator;
	public String notificationForwardingURI;
	@javax.xml.bind.annotation.XmlElement( required  = false)
	public Map<String,Object> ipeDiscoveryRequest;//IPEDiscoveryRequestDTO
}
```

```
There are some other DTOs, please refer Javadoc section for them.
```

7 Javadoc

Please include Javadoc of any new APIs here, once the design has matured. Instructions on how to export Javadoc for inclusion in the RFC can be found here: https://www.osgi.org/members/RFC/Javadoc

Demo Documentation

18/06/24 12:07

Package Summary		Page
org.osgi.service.onem2m		27
org.osgi.service.onem2m.dto		42
org.osgi.service.onem2m.introspection		96

Package org.osgi.service.onem2m

Interface Summary		Page
AECore		28
CSECore		29
OperationIF		33
ProtocolBinding		36
ServiceLayer		41

Enum Summary		Page
CSEType		30
ProtocolBindingType		36
SerializationType		39

Exception Summary		Page
OneM2MException		32

Interface AECore

org.osgi.service.onem2m

All Superinterfaces:

ServiceLayer

```
public interface AECore
extends ServiceLayer
```

Methods inherited from interface org.osgi.service.onem2m.ServiceLayer
request

Interface CSECore

org.osgi.service.onem2m

All Superinterfaces:

OperationIF, ServiceLayer

```
public interface CSECore
extends ServiceLayer, OperationIF
```

Methods inherited from interface org.osgi.service.onem2m.ServiceLayer
request

Methods inherited from interface org.osgi.service.onem2m.OperationIF
create, **delete**, **discovery**, **notify**, **retrieve**, **retrieve**, **update**

Enum CSEType

org.osgi.service.onem2m

```
java.lang.Object
 └java.lang.Enum<CSEType>
   └org.osgi.service.onem2m.CSEType
```

All Implemented Interfaces:

Comparable<**CSEType**>, Serializable

```
public enum CSEType
extends Enum<CSEType>
```

Enum Constant Summary	*Page*
ASN_CSE	31
IN_CSE	31
MN_CSE	31

Method Summary		*Page*
static CSEType	**valueOf**(String name)	31
static CSEType[]	**values**()	31

Enum Constant Detail

IN_CSE

```
public static final CSEType IN_CSE
```

MN_CSE

```
public static final CSEType MN_CSE
```

ASN_CSE

```
public static final CSEType ASN_CSE
```

Method Detail

values

```
public static CSEType[] values()
```

valueOf

```
public static CSEType valueOf(String name)
```

Class OneM2MException

org.osgi.service.onem2m

```
java.lang.Object
 └java.lang.Throwable
   └java.lang.Exception
     └java.io.IOException
       └org.osgi.service.onem2m.OneM2MException
```

All Implemented Interfaces:

Serializable

```
public class OneM2MException
extends IOException
```

Field Summary		Page
String	cause	32
int	errorCode	32

Constructor Summary	Page
OneM2MException()	32

Field Detail

errorCode

```
public int errorCode
```

---

cause

```
public String cause
```

Constructor Detail

OneM2MException

```
public OneM2MException()
```

Interface OperationIF

org.osgi.service.onem2m

All Known Subinterfaces:
CSECore, ProtocolBinding

```
public interface OperationIF
```

Method Summary		Page
org.osgi.util.promise.Promise<ResourceDTO>	`create(String uri, ResourceDTO resource)` create resource	33
org.osgi.util.promise.Promise<Boolean>	`delete(String uri)` delete resource	34
org.osgi.util.promise.Promise<List<String>>	`discovery(String uri, FilterCriteriaDTO fc)` find resources	34
org.osgi.util.promise.Promise<Boolean>	`notify(String uri, NotificationDTO notification)` send notification	35
org.osgi.util.promise.Promise<ResourceDTO>	`retrieve(String uri, List<String> targetAttributes)` retrieve subset of attributes.	34
org.osgi.util.promise.Promise<ResourceDTO>	`retrieve(String uri, ResourceDTO resource)` retrieve resource	34
org.osgi.util.promise.Promise<ResourceDTO>	`update(String uri, ResourceDTO resource)` update resource	34

Method Detail

create

```
org.osgi.util.promise.Promise<ResourceDTO> create(String uri,
                                                  ResourceDTO resource)
```

create resource

Parameters:
- `uri` - URI for parent resource
- `resource` - resource data

Returns:
Promise of created resource

retrieve

```
org.osgi.util.promise.Promise<ResourceDTO> retrieve(String uri,
                                                    ResourceDTO resource)
```

retrieve resource

Parameters:

`uri` - URI for retrieving resource

Returns:

retrieved resource data

retrieve

```
org.osgi.util.promise.Promise<ResourceDTO> retrieve(String uri,
                                                    List<String> targetAttributes)
```

retrieve subset of attributes.

Parameters:

`uri` - URI for retrieving resource

`targetAttributes` - names of the target attribute

Returns:

retrieved resource data

update

```
org.osgi.util.promise.Promise<ResourceDTO> update(String uri,
                                                  ResourceDTO resource)
```

update resource

Parameters:

`uri` - URI for updating resource

`resource` - data resource

Returns:

updated resource

delete

```
org.osgi.util.promise.Promise<Boolean> delete(String uri)
```

delete resource

Parameters:

`uri` - target URI for deleting resource

discovery

```
org.osgi.util.promise.Promise<List<String>> discovery(String uri,
                                                      FilterCriteriaDTO fc)
```

find resources

Parameters:

`uri` - URI for top of search

`fc` - filter criteria

Returns:

list of URIs matching the condition specified in fc

notify

```
org.osgi.util.promise.Promise<Boolean> notify(String uri,
                                              NotificationDTO notification)
```

send notification

Interface ProtocolBinding

org.osgi.service.onem2m

All Superinterfaces:

OperationIF, ServiceLayer

```
public interface ProtocolBinding
extends ServiceLayer, OperationIF
```

Methods inherited from interface org.osgi.service.onem2m.ServiceLayer
request

Methods inherited from interface org.osgi.service.onem2m.OperationIF
create, delete, discovery, notify, retrieve, retrieve, update

Enum ProtocolBindingType

org.osgi.service.onem2m

```
java.lang.Object
  └java.lang.Enum<ProtocolBindingType>
      └org.osgi.service.onem2m.ProtocolBindingType
```

All Implemented Interfaces:

Comparable<ProtocolBindingType>, Serializable

```
public enum ProtocolBindingType
extends Enum<ProtocolBindingType>
```

Enum Constant Summary	Page
COAP	37
HTTP	37
MQTT	37
reserve1	38
reserve2	38
WebService	37

Method Summary		Page
static ProtocolBindingType	valueOf(String name)	38
static ProtocolBindingType[]	values()	38

Enum Constant Detail

HTTP

```
public static final ProtocolBindingType HTTP
```

MQTT

```
public static final ProtocolBindingType MQTT
```

COAP

```
public static final ProtocolBindingType COAP
```

WebService

```
public static final ProtocolBindingType WebService
```

reserve1

```
public static final ProtocolBindingType reserve1
```

reserve2

```
public static final ProtocolBindingType reserve2
```

Method Detail

values

```
public static ProtocolBindingType[] values()
```

valueOf

```
public static ProtocolBindingType valueOf(String name)
```

Enum SerializationType

org.osgi.service.onem2m

```
java.lang.Object
 └java.lang.Enum<SerializationType>
   └org.osgi.service.onem2m.SerializationType
```

All Implemented Interfaces:

Comparable<**SerializationType**>, Serializable

```
public enum SerializationType
extends Enum<SerializationType>
```

Enum Constant Summary	Page
CBOR	39
JSON	39
XML	39

Method Summary		Page
static SerializationType	**valueOf**(String name)	40
static SerializationType[]	**values**()	39

Enum Constant Detail

XML

`public static final SerializationType XML`

JSON

`public static final SerializationType JSON`

CBOR

`public static final SerializationType CBOR`

Method Detail

values

`public static SerializationType[] values()`

valueOf

```
public static SerializationType valueOf(String name)
```

Interface ServiceLayer

org.osgi.service.onem2m

All Known Subinterfaces:

AECore, **CSECore**, **ProtocolBinding**

```
public interface ServiceLayer
```

Method Summary		*Page*
`org.osgi.util.promise.Promise<ResponsePrimitiveDTO>`	**`request`**`(RequestPrimitiveDTO request)` send a request.	41

Method Detail

request

```
org.osgi.util.promise.Promise<ResponsePrimitiveDTO> request(RequestPrimitiveDTO request)
```

send a request.

Parameters:

`request` - request

Returns:

promise for ResponseDTO.

Package org.osgi.service.onem2m.dto

Class Summary		Page
AttributeDTO		43
ChildResourceRefDTO		45
DasInfoDTO		47
DynAuthTokenReqInfoDTO		48
FilterCriteriaDTO		49
LocalTokenIdAssignmentDTO		58
NotificationDTO		59
PrimitiveContentDTO		61
RequestPrimitiveDTO		64
ResourceDTO		75
ResourceWrapperDTO		85
ResponsePrimitiveDTO		86
ResponseTypeInfoDTO		91
SecurityInfoDTO		94

Enum Summary		Page
FilterCriteriaDTO.FilterOperation		54
FilterCriteriaDTO.FilterUsage		56
ReleaseVersion		62
RequestPrimitiveDTO.DiscoveryResultType		69
RequestPrimitiveDTO.Operation		71
RequestPrimitiveDTO.ResultContent		73
ResourceType		77

ResponsePrimitiveDTO.ContentStatus		89
ResponseTypeInfoDTO.ResponseType		92

Class AttributeDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.AttributeDTO
```

```
public class AttributeDTO
extends org.osgi.dto.DTO
```

Field Summary		*Page*
String	**name**	44
Object	**value**	44

Constructor Summary	*Page*
AttributeDTO()	44

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

name

`public String name`

value

`public Object value`

Constructor Detail

AttributeDTO

`public AttributeDTO()`

Class ChildResourceRefDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.ChildResourceRefDTO
```

```
public class ChildResourceRefDTO
extends org.osgi.dto.DTO
```

Field Summary		*Page*
String	name	45
String	specializationID	45
Integer	type	45
String	uri	45

Constructor Summary	*Page*
ChildResourceRefDTO()	46

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

uri

`public String uri`

name

`public String name`

type

`public Integer type`

specializationID

`public String specializationID`

Constructor Detail

ChildResourceRefDTO

```
public ChildResourceRefDTO()
```

Class DasInfoDTO

[*org.osgi.service.onem2m.dto*]

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.DasInfoDTO
```

```
public class DasInfoDTO
extends org.osgi.dto.DTO
```

Field Summary		Page
Map<String ,Object>	dasRequest	47
String	securedDasRequest	47
String	uri	47

Constructor Summary	Page
DasInfoDTO()	47

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

uri

`public String uri`

---

dasRequest

`public Map<String,Object> dasRequest`

---

securedDasRequest

`public String securedDasRequest`

Constructor Detail

DasInfoDTO

`public DasInfoDTO()`

Class DynAuthTokenReqInfoDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.DynAuthTokenReqInfoDTO
```

```
public class DynAuthTokenReqInfoDTO
extends org.osgi.dto.DTO
```

Field Summary		***Page***
List<Map<String,Object>>	dasInfo	48

Constructor Summary	***Page***
DynAuthTokenReqInfoDTO()	48

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

dasInfo

```
public List<Map<String,Object>> dasInfo
```

Constructor Detail

DynAuthTokenReqInfoDTO

```
public DynAuthTokenReqInfoDTO()
```

Class FilterCriteriaDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.FilterCriteriaDTO
```

```
public class FilterCriteriaDTO
extends org.osgi.dto.DTO
```

Nested Class Summary		Page
static enum	FilterCriteriaDTO.FilterOperation	54
static enum	FilterCriteriaDTO.FilterUsage	56

Field Summary		Page
String	applyRelativePath	53
AttributeDTO	attribute	51
AttributeDTO	childAttribute	53
List<String>	childLabels	52
Integer	childResourceType	52
String	contentFilterQuery	52
Integer	contentFilterSyntax	52
List<String>	contentType	51
String	createdAfter	50
String	createdBefore	50
String	expireAfter	51
String	expireBefore	51
FilterCriteriaDTO.FilterOperation	filterOperation	52
FilterCriteriaDTO.FilterUsage	filterUsage	51
List<String>	labels	51
String	labelsQuery	52
Integer	level	52
Integer	limit	51
String	modifiedSince	50
Integer	offset	52
AttributeDTO	parentAttribute	53
List<String>	parentLabels	52
Integer	parentResourceType	53

List<Integer>	**resourceType**	51
String	**semanticsFilter**	52
Integer	**sizeAbove**	51
Integer	**sizeBelow**	51
Integer	**stateTagBigger**	50
Integer	**stateTagSmaller**	50
String	**unmodifiedSince**	50

Constructor Summary	*Page*
FilterCriteriaDTO()	53

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

createdBefore

```
public String createdBefore
```

createdAfter

```
public String createdAfter
```

modifiedSince

```
public String modifiedSince
```

unmodifiedSince

```
public String unmodifiedSince
```

stateTagSmaller

```
public Integer stateTagSmaller
```

stateTagBigger

```
public Integer stateTagBigger
```

expireBefore

```
public String expireBefore
```

expireAfter

```
public String expireAfter
```

labels

```
public List<String> labels
```

resourceType

```
public List<Integer> resourceType
```

sizeAbove

```
public Integer sizeAbove
```

sizeBelow

```
public Integer sizeBelow
```

contentType

```
public List<String> contentType
```

attribute

```
public AttributeDTO attribute
```

filterUsage

```
public FilterCriteriaDTO.FilterUsage filterUsage
```

limit

```
public Integer limit
```

semanticsFilter

```
public String semanticsFilter
```

filterOperation

```
public FilterCriteriaDTO.FilterOperation filterOperation
```

contentFilterSyntax

```
public Integer contentFilterSyntax
```

contentFilterQuery

```
public String contentFilterQuery
```

level

```
public Integer level
```

offset

```
public Integer offset
```

childLabels

```
public List<String> childLabels
```

parentLabels

```
public List<String> parentLabels
```

labelsQuery

```
public String labelsQuery
```

childResourceType

```
public Integer childResourceType
```

parentResourceType

```
public Integer parentResourceType
```

childAttribute

```
public AttributeDTO childAttribute
```

parentAttribute

```
public AttributeDTO parentAttribute
```

applyRelativePath

```
public String applyRelativePath
```

Constructor Detail

FilterCriteriaDTO

```
public FilterCriteriaDTO()
```

Enum FilterCriteriaDTO.FilterOperation

org.osgi.service.onem2m.dto

```
java.lang.Object
 └java.lang.Enum<FilterCriteriaDTO.FilterOperation>
   └org.osgi.service.onem2m.dto.FilterCriteriaDTO.FilterOperation
```

All Implemented Interfaces:

Comparable<FilterCriteriaDTO.FilterOperation>, Serializable

Enclosing class:

FilterCriteriaDTO

```
public static enum FilterCriteriaDTO.FilterOperation
extends Enum<FilterCriteriaDTO.FilterOperation>
```

Enum Constant Summary	Page
AND	54
OR	54

Method Summary		Page
int	getValue()	55
static FilterCriteriaDTO.FilterOperation	valueOf(String name)	55
static FilterCriteriaDTO.FilterOperation[]	values()	54

Enum Constant Detail

AND

```
public static final FilterCriteriaDTO.FilterOperation AND
```

OR

```
public static final FilterCriteriaDTO.FilterOperation OR
```

Method Detail

values

```
public static FilterCriteriaDTO.FilterOperation[] values()
```

valueOf

```
public static FilterCriteriaDTO.FilterOperation valueOf(String name)
```

getValue

```
public int getValue()
```

Enum FilterCriteriaDTO.FilterUsage

org.osgi.service.onem2m.dto

```
java.lang.Object
 └java.lang.Enum<FilterCriteriaDTO.FilterUsage>
   └org.osgi.service.onem2m.dto.FilterCriteriaDTO.FilterUsage
```

All Implemented Interfaces:

Comparable<FilterCriteriaDTO.FilterUsage>, Serializable

Enclosing class:

FilterCriteriaDTO

```
public static enum FilterCriteriaDTO.FilterUsage
extends Enum<FilterCriteriaDTO.FilterUsage>
```

Enum Constant Summary	*Page*
ConditionalRetrival	56
DiscoveryCriteria	56
IPEOndemandDiscovery	56

Method Summary		*Page*
int	**getValue**()	57
static FilterCriteriaDTO.FilterUsage	**valueOf**(String name)	57
static FilterCriteriaDTO.FilterUsage[]	**values**()	57

Enum Constant Detail

DiscoveryCriteria

```
public static final FilterCriteriaDTO.FilterUsage DiscoveryCriteria
```

ConditionalRetrival

```
public static final FilterCriteriaDTO.FilterUsage ConditionalRetrival
```

IPEOndemandDiscovery

```
public static final FilterCriteriaDTO.FilterUsage IPEOndemandDiscovery
```

Method Detail

values

```
public static FilterCriteriaDTO.FilterUsage[] values()
```

valueOf

```
public static FilterCriteriaDTO.FilterUsage valueOf(String name)
```

getValue

```
public int getValue()
```

Class LocalTokenIdAssignmentDTO

[org.osgi.service.onem2m.dto]

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.LocalTokenIdAssignmentDTO
```

```
public class LocalTokenIdAssignmentDTO
extends org.osgi.dto.DTO
```

Field Summary		Page
String	localTokenID	58
String	tokenID	58

Constructor Summary	Page
LocalTokenIdAssignmentDTO()	58

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

localTokenID

```
public String localTokenID
```

tokenID

```
public String tokenID
```

Constructor Detail

LocalTokenIdAssignmentDTO

```
public LocalTokenIdAssignmentDTO()
```

Class NotificationDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.NotificationDTO
```

```
public class NotificationDTO
extends org.osgi.dto.DTO
```

Field Summary		*Page*
String	**creator**	60
Map<String,Object>	**ipeDiscoveryRequest**	60
Map<String,Object>	**notificationEvent**	59
String	**notificationForwardingURI**	60
Boolean	**subscriptionDeletion**	59
String	**subscriptionReference**	59
Boolean	**verificationRequest**	59

Constructor Summary	*Page*
NotificationDTO()	60

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

notificationEvent

`public Map<String,Object> notificationEvent`

verificationRequest

`public Boolean verificationRequest`

subscriptionDeletion

`public Boolean subscriptionDeletion`

subscriptionReference

`public String subscriptionReference`

creator

```
public String creator
```

notificationForwardingURI

```
public String notificationForwardingURI
```

ipeDiscoveryRequest

```
public Map<String,Object> ipeDiscoveryRequest
```

Constructor Detail

NotificationDTO

```
public NotificationDTO()
```

Class PrimitiveContentDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.PrimitiveContentDTO
```

```
public class PrimitiveContentDTO
extends org.osgi.dto.DTO
```

Constructor Summary	*Page*
PrimitiveContentDTO()	61

Methods inherited from class org.osgi.dto.DTO
toString

Constructor Detail

PrimitiveContentDTO

```
public PrimitiveContentDTO()
```

Enum ReleaseVersion

org.osgi.service.onem2m.dto

```
java.lang.Object
 └java.lang.Enum<ReleaseVersion>
   └org.osgi.service.onem2m.dto.ReleaseVersion
```

All Implemented Interfaces:

Comparable<**ReleaseVersion**>, Serializable

```
public enum ReleaseVersion
extends Enum<ReleaseVersion>
```

Enum Constant Summary	*Page*
R1_0	62
R1_1	62
R2_0	62
R2A	62
R3_0	63

Method Summary		*Page*
static ReleaseVersion	**valueOf**(String name)	63
static ReleaseVersion[]	**values**()	63

Enum Constant Detail

R1_0

`public static final ReleaseVersion R1_0`

R1_1

`public static final ReleaseVersion R1_1`

R2_0

`public static final ReleaseVersion R2_0`

R2A

`public static final ReleaseVersion R2A`

R3_0

```
public static final ReleaseVersion R3_0
```

Method Detail

values

```
public static ReleaseVersion[] values()
```

valueOf

```
public static ReleaseVersion valueOf(String name)
```

Class RequestPrimitiveDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.RequestPrimitiveDTO
```

```
public class RequestPrimitiveDTO
extends org.osgi.dto.DTO
```

Nested Class Summary		Page
static enum	RequestPrimitiveDTO.DiscoveryResultType	69
static enum	RequestPrimitiveDTO.Operation	71
static enum	RequestPrimitiveDTO.ResultContent	73

Field Summary		Page
Boolean	authRelationshipIndicator	68
List<String>	authSignature	67
Boolean	authSignatureIndicator	67
PrimitiveContentDTO	content	65
Boolean	deliveryAggregation	66
RequestPrimitiveDTO.DiscoveryResultType	discoveryResultType	67
String	eventCategory	66
FilterCriteriaDTO	filterCriteria	67
String	from	65
String	groupRequestIdentifier	67
List<String>	groupRequestTargetMembers	67
List<String>	localTokenIDs	67
RequestPrimitiveDTO.Operation	operation	65
String	operationExecutionTime	66
String	originatingTimestamp	66
ReleaseVersion	releaseVersion	68
String	requestExpirationTimestamp	66
String	requestIdentifier	65
Integer	resourceType	65
ResponseTypeInfoDTO	responseType	66
RequestPrimitiveDTO.ResultContent	resultContent	66

String	**resultExpirationTimestamp**	66
String	**resultPersistence**	66
List<String>	**roleIDs**	66
Boolean	**semanticQueryIndicator**	68
String	**to**	65
List<String>	**tokenIDs**	67
Boolean	**tokenReqIndicator**	67
String	**tokens**	67
String	**verndorInformation**	68

Constructor Summary	*Page*
RequestPrimitiveDTO()	68

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

operation

`public RequestPrimitiveDTO.Operation` **`operation`**

---

to

`public String` **`to`**

---

from

`public String` **`from`**

---

requestIdentifier

`public String` **`requestIdentifier`**

---

resourceType

`public Integer` **`resourceType`**

---

content

`public PrimitiveContentDTO` **`content`**

---

roleIDs

```
public List<String> roleIDs
```

originatingTimestamp

```
public String originatingTimestamp
```

requestExpirationTimestamp

```
public String requestExpirationTimestamp
```

resultExpirationTimestamp

```
public String resultExpirationTimestamp
```

operationExecutionTime

```
public String operationExecutionTime
```

responseType

```
public ResponseTypeInfoDTO responseType
```

resultPersistence

```
public String resultPersistence
```

resultContent

```
public RequestPrimitiveDTO.ResultContent resultContent
```

eventCategory

```
public String eventCategory
```

deliveryAggregation

```
public Boolean deliveryAggregation
```

groupRequestIdentifier

```
public String groupRequestIdentifier
```

filterCriteria

```
public FilterCriteriaDTO filterCriteria
```

discoveryResultType

```
public RequestPrimitiveDTO.DiscoveryResultType discoveryResultType
```

tokens

```
public String tokens
```

tokenIDs

```
public List<String> tokenIDs
```

localTokenIDs

```
public List<String> localTokenIDs
```

tokenReqIndicator

```
public Boolean tokenReqIndicator
```

groupRequestTargetMembers

```
public List<String> groupRequestTargetMembers
```

authSignatureIndicator

```
public Boolean authSignatureIndicator
```

authSignature

```
public List<String> authSignature
```

authRelationshipIndicator

```
public Boolean authRelationshipIndicator
```

semanticQueryIndicator

```
public Boolean semanticQueryIndicator
```

releaseVersion

```
public ReleaseVersion releaseVersion
```

verndorInformation

```
public String verndorInformation
```

Constructor Detail

RequestPrimitiveDTO

```
public RequestPrimitiveDTO()
```

Enum RequestPrimitiveDTO.DiscoveryResultType

org.osgi.service.onem2m.dto

```
java.lang.Object
 └java.lang.Enum<RequestPrimitiveDTO.DiscoveryResultType>
   └org.osgi.service.onem2m.dto.RequestPrimitiveDTO.DiscoveryResultType
```

All Implemented Interfaces:

Comparable<**RequestPrimitiveDTO.DiscoveryResultType**>, Serializable

Enclosing class:

RequestPrimitiveDTO

```
public static enum RequestPrimitiveDTO.DiscoveryResultType
extends Enum<RequestPrimitiveDTO.DiscoveryResultType>
```

Enum Constant Summary	*Page*
structured	69
unstructured	69

Method Summary		*Page*
int	**getValue**()	70
static RequestPrimitiveDTO.DiscoveryResultType	**valueOf**(String name)	70
static RequestPrimitiveDTO.DiscoveryResultType[]	**values**()	69

Enum Constant Detail

structured

`public static final RequestPrimitiveDTO.DiscoveryResultType **structured**`

unstructured

`public static final RequestPrimitiveDTO.DiscoveryResultType **unstructured**`

Method Detail

values

`public static RequestPrimitiveDTO.DiscoveryResultType[] **values**()`

valueOf

```
public static RequestPrimitiveDTO.DiscoveryResultType valueOf(String name)
```

getValue

```
public int getValue()
```

Enum RequestPrimitiveDTO.Operation

org.osgi.service.onem2m.dto

```
java.lang.Object
 └java.lang.Enum<RequestPrimitiveDTO.Operation>
   └org.osgi.service.onem2m.dto.RequestPrimitiveDTO.Operation
```

All Implemented Interfaces:

Comparable<RequestPrimitiveDTO.Operation>, Serializable

Enclosing class:

RequestPrimitiveDTO

```
public static enum RequestPrimitiveDTO.Operation
extends Enum<RequestPrimitiveDTO.Operation>
```

Enum Constant Summary	Page
Create	71
Delete	72
Notify	72
Retrieve	71
Update	71

Method Summary		Page
int	getValue()	72
static RequestPrimitiveDTO.Operation	valueOf(String name)	72
static RequestPrimitiveDTO.Operation[]	values()	72

Enum Constant Detail

Create

```
public static final RequestPrimitiveDTO.Operation Create
```

Retrieve

```
public static final RequestPrimitiveDTO.Operation Retrieve
```

Update

```
public static final RequestPrimitiveDTO.Operation Update
```

Delete

```
public static final RequestPrimitiveDTO.Operation Delete
```

Notify

```
public static final RequestPrimitiveDTO.Operation Notify
```

Method Detail

values

```
public static RequestPrimitiveDTO.Operation[] values()
```

valueOf

```
public static RequestPrimitiveDTO.Operation valueOf(String name)
```

getValue

```
public int getValue()
```

Enum RequestPrimitiveDTO.ResultContent

org.osgi.service.onem2m.dto

```
java.lang.Object
 └java.lang.Enum<RequestPrimitiveDTO.ResultContent>
   └org.osgi.service.onem2m.dto.RequestPrimitiveDTO.ResultContent
```

All Implemented Interfaces:

Comparable<**RequestPrimitiveDTO.ResultContent**>, Serializable

Enclosing class:

RequestPrimitiveDTO

```
public static enum RequestPrimitiveDTO.ResultContent
extends Enum<RequestPrimitiveDTO.ResultContent>
```

Enum Constant Summary	*Page*
attributes	73
attributesAndChildResourceReferences	74
attributesAndChildResources	74
childResourceReferences	74
childResources	74
hierarchicalAddress	74
hierarchicalAddressAndAttributes	74
nothing	73
originalResource	74

Method Summary		*Page*
int	**getValue**()	74
static RequestPrimitiveDTO.ResultContent	**valueOf**(String name)	74
static RequestPrimitiveDTO.ResultContent[]	**values**()	74

Enum Constant Detail

nothing

`public static final RequestPrimitiveDTO.ResultContent` **`nothing`**

attributes

`public static final RequestPrimitiveDTO.ResultContent` **`attributes`**

hierarchicalAddress

```
public static final RequestPrimitiveDTO.ResultContent hierarchicalAddress
```

hierarchicalAddressAndAttributes

```
public static final RequestPrimitiveDTO.ResultContent hierarchicalAddressAndAttributes
```

attributesAndChildResources

```
public static final RequestPrimitiveDTO.ResultContent attributesAndChildResources
```

attributesAndChildResourceReferences

```
public static final RequestPrimitiveDTO.ResultContent attributesAndChildResourceReferences
```

childResourceReferences

```
public static final RequestPrimitiveDTO.ResultContent childResourceReferences
```

originalResource

```
public static final RequestPrimitiveDTO.ResultContent originalResource
```

childResources

```
public static final RequestPrimitiveDTO.ResultContent childResources
```

Method Detail

values

```
public static RequestPrimitiveDTO.ResultContent[] values()
```

valueOf

```
public static RequestPrimitiveDTO.ResultContent valueOf(String name)
```

getValue

```
public int getValue()
```

Class ResourceDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.ResourceDTO
```

```
public class ResourceDTO
extends org.osgi.dto.DTO
```

Field Summary		*Page*
Map<String,Object>	**attribute** Non Universal Attribute.	76
String	**creationTime**	76
List<String>	**labels**	76
String	**lastModifiedTime**	76
String	**parentID**	75
String	**resourceID**	75
String	**resourceName**	76
Integer	**resourceType**	75

Constructor Summary	*Page*
ResourceDTO()	76

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

resourceType

```
public Integer resourceType
```

resourceID

```
public String resourceID
```

parentID

```
public String parentID
```

creationTime

```
public String creationTime
```

lastModifiedTime

```
public String lastModifiedTime
```

resourceName

```
public String resourceName
```

labels

```
public List<String> labels
```

attribute

```
public Map<String,Object> attribute
```

Non Universal Attribute. Value Part must be the types that are allowed for OSGi DTO.

Constructor Detail

ResourceDTO

```
public ResourceDTO()
```

Enum ResourceType

org.osgi.service.onem2m.dto

```
java.lang.Object
 └java.lang.Enum<ResourceType>
   └org.osgi.service.onem2m.dto.ResourceType
```

All Implemented Interfaces:

Comparable<**ResourceType**>, Serializable

```
public enum ResourceType
extends Enum<ResourceType>
```

Enum Constant Summary	*Page*
accessControlPolicy	78
accessControlPolicyAnnc	82
AE	78
AEAnnc	82
container	78
containerAnnc	82
contentInstance	79
contentInstanceAnnc	82
CSEBase	79
delivery	79
dynamicAuthorizationConsultation	82
dynamicAuthorizationConsultationAnnc	83
eventConfig	79
execInstance	79
fanOutPoint	83
flexContainer	81
flexContainerAnnc	83
group	79
groupAnnc	82
latest	83
locationPolicy	79
locationPolicyAnnc	82
m2mServiceSubscriptionProfile	79
mgmtCmd	79
mgmtObj	79
mgmtObjAnnc	82
node	80
nodeAnnc	82
notificationTargetMgmtPolicyRef	81
notificationTargetPolicy	81
oldest	83

policyDeletionRules	81
pollingChannel	80
pollingChannelURI	84
remoteCSE	80
remoteCSEAnnc	82
request	80
role	81
schedule	80
scheduleAnnc	83
semanticDescriptor	81
semanticDescriptorAnnc	83
serviceSubscribedAppRule	80
serviceSubscribedNode	80
statsCollect	80
statsConfig	80
subscription	80
timeSeries	81
timeSeriesAnnc	83
timeSeriesInstance	81
timeSeriesInstanceAnnc	83
token	81
trafficPattern	81
trafficPatternAnnc	83

Method Summary		***Page***
int	**getValue**()	84
static ResourceType	**valueOf**(String name)	84
static ResourceType[]	**values**()	84

Enum Constant Detail

accessControlPolicy

`public static final ResourceType` **accessControlPolicy**

AE

`public static final ResourceType` **AE**

---

container

`public static final ResourceType` **container**

---

contentInstance

```
public static final ResourceType contentInstance
```

CSEBase

```
public static final ResourceType CSEBase
```

delivery

```
public static final ResourceType delivery
```

eventConfig

```
public static final ResourceType eventConfig
```

execInstance

```
public static final ResourceType execInstance
```

group

```
public static final ResourceType group
```

locationPolicy

```
public static final ResourceType locationPolicy
```

m2mServiceSubscriptionProfile

```
public static final ResourceType m2mServiceSubscriptionProfile
```

mgmtCmd

```
public static final ResourceType mgmtCmd
```

mgmtObj

```
public static final ResourceType mgmtObj
```

node

```
public static final ResourceType node
```

pollingChannel

```
public static final ResourceType pollingChannel
```

remoteCSE

```
public static final ResourceType remoteCSE
```

request

```
public static final ResourceType request
```

schedule

```
public static final ResourceType schedule
```

serviceSubscribedAppRule

```
public static final ResourceType serviceSubscribedAppRule
```

serviceSubscribedNode

```
public static final ResourceType serviceSubscribedNode
```

statsCollect

```
public static final ResourceType statsCollect
```

statsConfig

```
public static final ResourceType statsConfig
```

subscription

```
public static final ResourceType subscription
```

semanticDescriptor

```
public static final ResourceType semanticDescriptor
```

notificationTargetMgmtPolicyRef

```
public static final ResourceType notificationTargetMgmtPolicyRef
```

notificationTargetPolicy

```
public static final ResourceType notificationTargetPolicy
```

policyDeletionRules

```
public static final ResourceType policyDeletionRules
```

flexContainer

```
public static final ResourceType flexContainer
```

timeSeries

```
public static final ResourceType timeSeries
```

timeSeriesInstance

```
public static final ResourceType timeSeriesInstance
```

role

```
public static final ResourceType role
```

token

```
public static final ResourceType token
```

trafficPattern

```
public static final ResourceType trafficPattern
```

dynamicAuthorizationConsultation

```
public static final ResourceType dynamicAuthorizationConsultation
```

accessControlPolicyAnnc

```
public static final ResourceType accessControlPolicyAnnc
```

AEAnnc

```
public static final ResourceType AEAnnc
```

containerAnnc

```
public static final ResourceType containerAnnc
```

contentInstanceAnnc

```
public static final ResourceType contentInstanceAnnc
```

groupAnnc

```
public static final ResourceType groupAnnc
```

locationPolicyAnnc

```
public static final ResourceType locationPolicyAnnc
```

mgmtObjAnnc

```
public static final ResourceType mgmtObjAnnc
```

nodeAnnc

```
public static final ResourceType nodeAnnc
```

remoteCSEAnnc

```
public static final ResourceType remoteCSEAnnc
```

scheduleAnnc

`public static final ResourceType scheduleAnnc`

semanticDescriptorAnnc

`public static final ResourceType semanticDescriptorAnnc`

flexContainerAnnc

`public static final ResourceType flexContainerAnnc`

timeSeriesAnnc

`public static final ResourceType timeSeriesAnnc`

timeSeriesInstanceAnnc

`public static final ResourceType timeSeriesInstanceAnnc`

trafficPatternAnnc

`public static final ResourceType trafficPatternAnnc`

dynamicAuthorizationConsultationAnnc

`public static final ResourceType dynamicAuthorizationConsultationAnnc`

latest

`public static final ResourceType latest`

oldest

`public static final ResourceType oldest`

fanOutPoint

`public static final ResourceType fanOutPoint`

pollingChannelURI

```
public static final ResourceType pollingChannelURI
```

Method Detail

values

```
public static ResourceType[] values()
```

valueOf

```
public static ResourceType valueOf(String name)
```

getValue

```
public int getValue()
```

Class ResourceWrapperDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.ResourceWrapperDTO
```

```
public class ResourceWrapperDTO
extends org.osgi.dto.DTO
```

Field Summary		*Page*
String	uri	85

Constructor Summary	*Page*
ResourceWrapperDTO()	85

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

uri

`public String uri`

Constructor Detail

ResourceWrapperDTO

`public ResourceWrapperDTO()`

Class ResponsePrimitiveDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.ResponsePrimitiveDTO
```

```
public class ResponsePrimitiveDTO
extends org.osgi.dto.DTO
```

Nested Class Summary		Page
static enum	ResponsePrimitiveDTO.ContentStatus	89

Field Summary		Page
List<LocalTokenIdAssignmentDTO>	assignedTokenIdentifiers	88
Boolean	AuthSignatureReqInfo	88
PrimitiveContentDTO	content	87
Integer	contentOffset	87
ResponsePrimitiveDTO.ContentStatus	contentStatus	87
String	eventCategory	87
String	from	87
String	originatingTimestamp	87
ReleaseVersion	releaseVersionIndicator	88
String	requestIdentifier	87
Integer	responseStatusCode	87
String	resultExpirationTimestamp	87
String	to	87
List<DasInfoDTO>	tokenReqInfo	88
String	vendorInformation	88

Constructor Summary	Page
ResponsePrimitiveDTO()	88

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

responseStatusCode

`public Integer responseStatusCode`

---

requestIdentifier

`public String requestIdentifier`

---

content

`public PrimitiveContentDTO content`

---

to

`public String to`

---

from

`public String from`

---

originatingTimestamp

`public String originatingTimestamp`

---

resultExpirationTimestamp

`public String resultExpirationTimestamp`

---

eventCategory

`public String eventCategory`

---

contentStatus

`public ResponsePrimitiveDTO.ContentStatus contentStatus`

---

contentOffset

`public Integer contentOffset`

assignedTokenIdentifiers

```
public List<LocalTokenIdAssignmentDTO> assignedTokenIdentifiers
```

tokenReqInfo

```
public List<DasInfoDTO> tokenReqInfo
```

AuthSignatureReqInfo

```
public Boolean AuthSignatureReqInfo
```

releaseVersionIndicator

```
public ReleaseVersion releaseVersionIndicator
```

vendorInformation

```
public String vendorInformation
```

Constructor Detail

ResponsePrimitiveDTO

```
public ResponsePrimitiveDTO()
```

Enum ResponsePrimitiveDTO.ContentStatus

org.osgi.service.onem2m.dto

```
java.lang.Object
 └java.lang.Enum<ResponsePrimitiveDTO.ContentStatus>
   └org.osgi.service.onem2m.dto.ResponsePrimitiveDTO.ContentStatus
```

All Implemented Interfaces:

Comparable<ResponsePrimitiveDTO.ContentStatus>, Serializable

Enclosing class:

ResponsePrimitiveDTO

```
public static enum ResponsePrimitiveDTO.ContentStatus
extends Enum<ResponsePrimitiveDTO.ContentStatus>
```

Enum Constant Summary	*Page*
FULL_CONTENT	89
PARTIAL_CONTENT	89

Method Summary		*Page*
static ResponsePrimitiveDTO.ContentStatus	**valueOf**(String name)	90
static ResponsePrimitiveDTO.ContentStatus[]	**values**()	89

Enum Constant Detail

PARTIAL_CONTENT

```
public static final ResponsePrimitiveDTO.ContentStatus PARTIAL_CONTENT
```

FULL_CONTENT

```
public static final ResponsePrimitiveDTO.ContentStatus FULL_CONTENT
```

Method Detail

values

```
public static ResponsePrimitiveDTO.ContentStatus[] values()
```

valueOf

```
public static ResponsePrimitiveDTO.ContentStatus valueOf(String name)
```

Class ResponseTypeInfoDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.ResponseTypeInfoDTO
```

```
public class ResponseTypeInfoDTO
extends org.osgi.dto.DTO
```

Nested Class Summary		*Page*
static enum	**ResponseTypeInfoDTO.ResponseType**	92

Field Summary		*Page*
List<String>	**notificationURI**	91
ResponseTypeInfoDTO.ResponseType	**responseTypeValue**	91

Constructor Summary	*Page*
ResponseTypeInfoDTO()	91

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

responseTypeValue

```
public ResponseTypeInfoDTO.ResponseType responseTypeValue
```

notificationURI

```
public List<String> notificationURI
```

Constructor Detail

ResponseTypeInfoDTO

```
public ResponseTypeInfoDTO()
```

Enum ResponseTypeInfoDTO.ResponseType

org.osgi.service.onem2m.dto

```
java.lang.Object
 └java.lang.Enum<ResponseTypeInfoDTO.ResponseType>
   └org.osgi.service.onem2m.dto.ResponseTypeInfoDTO.ResponseType
```

All Implemented Interfaces:

Comparable<**ResponseTypeInfoDTO.ResponseType**>, Serializable

Enclosing class:

ResponseTypeInfoDTO

```
public static enum ResponseTypeInfoDTO.ResponseType
extends Enum<ResponseTypeInfoDTO.ResponseType>
```

Enum Constant Summary	*Page*
blockingRequest	92
flexBlocking	93
nonBlockingRequestAsynch	92
nonBlockingRequestSynch	92

Method Summary		*Page*
int	**getValue**()	93
static ResponseTypeInfoDTO.ResponseType	**valueOf**(String name)	93
static ResponseTypeInfoDTO.ResponseType[]	**values**()	93

Enum Constant Detail

nonBlockingRequestSynch

```
public static final ResponseTypeInfoDTO.ResponseType nonBlockingRequestSynch
```

nonBlockingRequestAsynch

```
public static final ResponseTypeInfoDTO.ResponseType nonBlockingRequestAsynch
```

blockingRequest

```
public static final ResponseTypeInfoDTO.ResponseType blockingRequest
```

flexBlocking

```
public static final ResponseTypeInfoDTO.ResponseType flexBlocking
```

Method Detail

values

```
public static ResponseTypeInfoDTO.ResponseType[] values()
```

valueOf

```
public static ResponseTypeInfoDTO.ResponseType valueOf(String name)
```

getValue

```
public int getValue()
```

Class SecurityInfoDTO

org.osgi.service.onem2m.dto

```
java.lang.Object
 └org.osgi.dto.DTO
   └org.osgi.service.onem2m.dto.SecurityInfoDTO
```

```
public class SecurityInfoDTO
extends org.osgi.dto.DTO
```

Field Summary		*Page*
Map<String,Object>	**dasRequest**	94
Map<String,Object>	**dasResponse**	94
byte[]	**escertkeMessage**	95
String	**esprimObject**	95
Map<String,Object>	**esprimRandObject**	94
Integer	**securityInfoType**	94

Constructor Summary	*Page*
SecurityInfoDTO()	95

Methods inherited from class org.osgi.dto.DTO
toString

Field Detail

securityInfoType

`public Integer securityInfoType`

dasRequest

`public Map<String,Object> dasRequest`

dasResponse

`public Map<String,Object> dasResponse`

esprimRandObject

`public Map<String,Object> esprimRandObject`

esprimObject

```
public String esprimObject
```

escertkeMessage

```
public byte[] escertkeMessage
```

Constructor Detail

SecurityInfoDTO

```
public SecurityInfoDTO()
```

Package org.osgi.service.onem2m.introspection

Interface Summary		*Page*
FlexContainerIntrospector	FlexContainerInspector	96
Introspector		99

Enum Summary		*Page*
Optionality		102

Interface FlexContainerIntrospector

org.osgi.service.onem2m.introspection

```
public interface FlexContainerIntrospector
```

FlexContainerInspector

Method Summary		Page
`String[]`	`findValidationProblems(ResourceDTO resource)` Execute Validation of Data Structure	97
`String[]`	`getCustomeAttributeNames(String containerDefinition)` get Possible Attributes for the given resouceType.	97
`Optionality`	`getOptionality(String containerDefinition, String attributeName)` get optionality of specified attribute	97
`Object`	`getTemplateObject(String containerDefinition, String customAttributeName)` return Typical Data Structure for Type for given attribute of resource type	98
`Class`	`getType(String containerDefinition, String attributeName)` return Java Type for given attribute of resource type	98

Method Detail

findValidationProblems

```
String[] findValidationProblems(ResourceDTO resource)
```

Execute Validation of Data Structure

Returns:

array of problems

---

getCustomeAttributeNames

```
String[] getCustomeAttributeNames(String containerDefinition)
```

get Possible Attributes for the given resouceType.

Returns:

array of Possible Attributes

---

getOptionality

```
Optionality getOptionality(String containerDefinition,
                           String attributeName)
```

get optionality of specified attribute

Parameters:

`containerDefinition` - container definition
`attributeName` - attribute name

getType

```
Class getType(String containerDefinition,
              String attributeName)
```

return Java Type for given attribute of resource type

Parameters:

`containerDefinition` - container definition of flexContainer

`attributeName` - attribute name

Returns:

expected class for the specified attribute

getTemplateObject

```
Object getTemplateObject(String containerDefinition,
                         String customAttributeName)
```

return Typical Data Structure for Type for given attribute of resource type

Parameters:

`containerDefinition` - container definition of flexContainer

Returns:

Template Object

Interface Introspector

org.osgi.service.onem2m.introspection

```
public interface Introspector
```

Method Summary		Page
String[]	`findValidationProblems(RequestPrimitiveDTO req)` execute Validation of Data Structure	99
String[]	`findValidationProblems(ResponsePrimitiveDTO resp)` execute Validation of Data Structure	99
String[]	`findValidationProblemsForCreate(ResourceDTO resource)` execute Validation of Data Structure	100
String[]	`findValidationProblemsForUpdate(ResourceDTO resource)` execute Validation of Data Structure	100
String[]	`getAttributeNames(int resourceType)` get Possible Attributes for specified resouceType.	100
Optionality	`getOptionalityForCreate(int resourceType, String attributeName)` get Optionality of specified attribute in create operation.	100
Optionality	`getOptionalityForUpdate(int resourceType, String attributeName)` get Optionality of specified attribute in update operation.	100
Object	`getTemplateObject(int resouceType, String Attribute)` return Typical Data Structure for Type of specified attribute of resource type	101
Class	`getType(int resouceType, String attributeName)` return Java Type for given attribute of resource type	101

Method Detail

findValidationProblems

```
String[] findValidationProblems(ResponsePrimitiveDTO resp)
```

execute Validation of Data Structure

Returns:

array of problems

findValidationProblems

```
String[] findValidationProblems(RequestPrimitiveDTO req)
```

execute Validation of Data Structure

Returns:

array of problems

findValidationProblemsForCreate

```
String[] findValidationProblemsForCreate(ResourceDTO resource)
```

execute Validation of Data Structure

Returns:
array of problems

findValidationProblemsForUpdate

```
String[] findValidationProblemsForUpdate(ResourceDTO resource)
```

execute Validation of Data Structure

Returns:
array of problems

getAttributeNames

```
String[] getAttributeNames(int resourceType)
```

get Possible Attributes for specified resouceType.

Returns:
array of Possible Attribute Names

getOptionalityForCreate

```
Optionality getOptionalityForCreate(int resourceType,
                                    String attributeName)
```

get Optionality of specified attribute in create operation.

Parameters:
`resourceType` - resource type
`attributeName` - attribute Name
Returns:
Optionality

getOptionalityForUpdate

```
Optionality getOptionalityForUpdate(int resourceType,
                                    String attributeName)
```

get Optionality of specified attribute in update operation.

Parameters:
`resourceType` - resource type
`attributeName` - attribute Name

Returns:

Optionality

getType

```
Class getType(int resouceType,
              String attributeName)
```

return Java Type for given attribute of resource type

Parameters:

`resouceType` - resource type

`attributeName` - attribute Name

Returns:

expected class for the specified attribute

getTemplateObject

```
Object getTemplateObject(int resouceType,
                         String Attribute)
```

return Typical Data Structure for Type of specified attribute of resource type

Returns:

Template Object

Enum Optionality

org.osgi.service.onem2m.introspection

```
java.lang.Object
 └java.lang.Enum<Optionality>
   └org.osgi.service.onem2m.introspection.Optionality
```

All Implemented Interfaces:

Comparable<**Optionality**>, Serializable

```
public enum Optionality
extends Enum<Optionality>
```

Enum Constant Summary	***Page***
Mandatory	102
NotPresent	102
Optional	102

Method Summary		***Page***
static Optionality	**valueOf**(String name)	103
static Optionality[]	**values**()	102

Enum Constant Detail

Mandatory

`public static final Optionality` **`Mandatory`**

Optional

`public static final Optionality` **`Optional`**

NotPresent

`public static final Optionality` **`NotPresent`**

Method Detail

values

`public static Optionality[]` **`values`**`()`

valueOf

```
public static Optionality valueOf(String name)
```

8 Considered Alternatives

For posterity, record the design alternatives that were considered but rejected along with the reason for rejection. This is especially important for external/earlier solutions that were deemed not applicable.

8.1 Representation of DTO

8.1.1 JAXB generated Class

As alternative solution, utilization of generated Java classes by JAXB has been considered, since oneM2M provides well defined XSD for defining data format. With the following aspects, this approach is not applied.

Many classes: Currently 65 XSD files are defined in oneM2M specification and JAXB tool (xjc) generates more than 140 Java classes. Using many classes as interface could make specification more complicated than its nature.

No Uniqueness: Generated classes by xjc are not unique, because it is possible to customize generation processes.

Changeability: Depending on the version of oneM2M, XSD files differ. It is preferable to choose version independent API, as much as possible. oneM2M ensures any data can be converted to JSON and CBOR, so proposed approach can be used with out modification, even if XSD file would be changed.

8.1.2 GenericDTO

GenricDTO, which has Map<String, Obj> in the top, has been discussed in Gent meeting. But it seems bad usage of defining DTO.

8.1.3 SpecificDTO

SpecificDTO definitions have been generated from XSD generated classes. The number of DTO exceeds 170 and Java doc pages are getting 300 pages. It is apparently too much to express data formats. So middle approach of generic DTO and specific DTO has been chosen.

8.2 Resource Types Expression

In DTO, enum was eagerly used for clear candidates of possible values. But resource types seems more fragile because new resource types could be easily added. So Integer was chosen for resource types.

8.3 Use of Annotation defined by JAXB in DTO

Currently annotations defined by JAXB was used in DTO. It was pointed out as confusing because it might give impression that it only support XML serialization. But it was kept in the definitions by following reasons.

1. Removing the annotations are easier than inserting.

2. It is informative to specify optionality.

New OSGI annotation specifying optionality could be possible, but it might take time because it should be published as Core specification and R7 just has released.

9 Security Considerations

Description of all known vulnerabilities this may either introduce or address as well as scenarios of how the weaknesses could be circumvented.

9.1 ProtocolBinding Service with secure protocol configuration

In case that ProtocolBinding Service uses secure protocols, it is expected to handle pre-shared key or certificate and other parameters. Those configuration could be very diverse. This is out of scope of this RFC and it is responsibility of bundle developer that provides ProtocolBindingService.

9.2 Binding of AE Core and Protocol Binding

Protocol Binding has identity information, such as a key or certificate, which represents AE, and AE core is bound to the service and use the identity. In case that unexpected AE Core is bind to the protocol binding, it would cause of spoofing. It is deployers responsibility to deploy only trustable AE core bundles, and to configure them properly.

Honest implementation of AE could solve situation.

1. AE Core knows it’s APP-ID, letting it as “MYAPP1”

2. AE Core searches Protocol binding service with property “APP-ID” is “MYAPP1” and bind it.

10 Document Support

10.1 References

[1]. Bradner, S., Key words for use in RFCs to Indicate Requirement Levels, RFC2119, March 1997.

[2]. oneM2M TS-0001 Functional Architecture, http://onem2m.org/images/files/deliverables/Release2/TS-0001-%20Functional_Architecture-V2_10_0.pdf

[3]. oneM2M TS-0004 Service Layer Core Protocol, http://onem2m.org/images/files/deliverables/Release2/TS-0004_Service_Layer_Core_Protocol_V2_7_1.zip

[4]. Software Requirements & Specifications. Michael Jackson. ISBN 0-201-87712-0 (NOTE:Is this needed?)

Add references simply by adding new items. You can then cross-refer to them by chosing <Insert><Cross Reference><Numbered Item> and then selecting the paragraph. ***STATIC REFERENCES (I.E. BODGED) ARE NOT ACCEPTABLE, SOMEONE WILL HAVE TO UPDATE THEM LATER, SO DO IT PROPERLY NOW.***

10.2 Author's Address

Name	Hiroyuki Maeomichi
Company	NTT
Address	Midorimachi 3-9-11, Musashino, Tokyo, Japan
Voice	+81 422 59 4072
e-mail	maeomichi.hiroyuki@lab.ntt.co.jp

10.3 Acronyms and Abbreviations

CSE: Common Services Entity

AE: Application Entity

CBOR: Concise Binary Object Representation

10.4 End of Document