|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
| **ISSUE REPORT** | | | | |
| (All fields are required to be filled, except the Owner assigned by TG and Filename which is automatically updated) | | | | |
| Spec Name | **Core Framework** | | |  | |  |
| *Use the following names:*  *Core Framework*  *Security*  *Smart Home Device*  *Smart Home Resource* | | | *If the CR applies to multiple specifications, list all.* |
| **Filename*:*** ⌘ ***(incl. Automatically assigned CR Number and Version)*** | | *CR ATG xxxx - Create, Delete, and LInkupdate Interfaces*.docx | | |

|  |  |
| --- | --- |
| ***Title:*** ⌘ | Create, Delete, and Linkupdate Inerface Definitions |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***Submitter:*** ⌘ | (Samsung) Michael Koster | | | | | |
| ***Owner:*** ⌘ | | (Samsung) Michael Koster | | | | |
| ***Certification:*** ⌘ | X | |  | ***Date:*** ⌘ | | 2018-05-6 |
| ***Category:*** ⌘ | F | |  |  | |  |
|  | *Use one of the following categories:* ***C*** *(correction)* ***F***  *(functional modification of feature)* ***E*** *(editorial modification)* | | | |  | |
| ***Summary and Reason for change:*** ⌘ | | | Add Create, Delete, and Linkupdate Interface Definitions | | | |
|  | | | | | | |
| ***Location of changes:*** ⌘ | | | 7.6.3.1 and new sections 7.6.3.9, 7.6.3.10, and 7.6.3.11 added | | | |
|  | | | | | | |
| ***Other comments:*** ⌘ | | |  | | | |

**How to propose a technical solution to the Issue submitted using this form (we are using the MS Word revision marks feature (also known as track changes) to designate the modified text in a Change Request. Please use revision marks as instructed below):**

Once the above Issue Report is filled out:

1. With “revision marks” disabled, copy the clause(s) from the Word version of the target specification subject to the Change Request into the following pages.
2. Then with “revision marks” enabled, make the changes to the clause(s).
3. If the CR adds or modifies references or includes new specification text, include all references in the CR and use bookmarks to create the references.
4. Do not denote new specification text, clauses, acronyms, references, etc. with revision marks. Instead, highlight (as appropriate) with a comment with an instruction; e.g., “Editor: New text to be inserted after Clause x.y.z”.
5. Do not try to force auto header numbering to work. When you copy in a heading, delete the auto-number and manually type in the correct number.

Note: if you are just submitting and issue and not submitting a technical solution, filling the second (and subsequent) page(s) is optional.

\*\*\*\*\*\* Paste the Change Request content here \*\*\*\*\*\*

\*\*\*\*\* First Change

### Interface methods

#### 7.6.3.1 Overview

The OCF-defined Interfaces are listed in the table below:

Table 1. OCF standard Interfaces

|  |  |  |  |
| --- | --- | --- | --- |
| Interface | Name | Applicable Operations | Description |
| baseline | “oic.if.baseline” | RETRIEVE, NOTIFY, UPDATE | The baseline Interface defines a view into all Properties of a Resource including the Meta Properties. This Interface is used to operate on the full Representation of a Resource. |
| links list | “oic.if.ll” | RETRIEVE,  NOTIFY | The ‘links list’ Interface provides a view into Links in a Collection (Resource).  Since Links represent relationships to other Resources, the links list interfaces may be used to discover Resources with respect to a context. The discovery is done by retrieving Links to these Resources. For example: the Core Resource “/oic/res” uses this Interface to allow discovery of Resource “hosted” on a Device. |
| batch | “oic.if.b” | RETRIEVE, NOTIFY, UPDATE | The batch Interface is used to interact with a collection of Resources at the same time. This also removes the need for the Client to first discover the Resources it is manipulating – the Server forwards the requests and aggregates the responses |
| read-only | “oic.if.r” | RETRIEVE NOTIFY | The read-only Interface exposes the Properties of a Resource that may be ‘read’. This Interface does not provide methods to update Properties or a Resource and so can only be used to ‘read’ Property Values. |
| read-write | “oic.if.rw” | RETRIEVE, NOTIFY, UPDATE | The read-write Interface exposes only those Properties that may be both ‘read’ and “written” and provides methods to read and write the Properties of a Resource. |
| actuator | “oic.if.a” | RETRIEVE, NOTIFY, UPDATE | The actuator Interface is used to read or write the Properties of an actuator Resource. |
| sensor | “oic.if.s” | RETRIEVE, NOTIFY | The sensor Interface is used to read the Properties of a sensor Resource. |
| create | "oic.if.create" | CREATE, NOTIFY | The create interface is used to create new resources in a collection. Both the data item and a link pointing to it are created as a single atomic operation. |
| delete | "oic.if.delete" | DELETE, NOTIFY | The delete interface is used to delete resources. If a collection is deleted, all of the items within it are recursively deleted. |
| linkupdate | "oic.if.linkupdate" | UPDATE, NOTIFY | The linkupdate interface is used to add, remove, or edit the contents of, the links in a collection. |

\*\*\*\*\* Second Change

**7.6.3.9 Create Interface**

**7.6.3.9.1 Overview**

The Create Interface is used to create resource instances in a collection. An instance of a resource and a link pointing to the resource are created together, atomically, according to a client-supplied representation.

The Create Interface supports two design patterns, one where the client provides the address for the created item, and another where the server provides the address for the created item.

The identifier "oic.if.create" is used in the "if" property of a resource or link to indicate that the resource supports the Create Interface.

A collection indicates which resource types it supports for created resources in its "rts" property. If a client attempts to create a resource type which is not supported by the collection resource, the server shall return a client error status indicating "Bad Request" (e.g. 400).

Successful CREATE operations shall return a success code indicating "Created" (e.g. 201).

**7.6.3.9.2 Data Format for CREATE**

The data format for the Create Interface is similar to the data format for the Batch Interface. The Create Interface format consists of set of link parameters and a "rep" parameter which contains a representation for the created resource.

The representation supplied for the link pointing to the newly created resource shall contain at least the "rt" and "if" link parameters, as shown in the first two elements in the example below.

The representation supplied for a created resource, in the value of the "rep" element, shall also contain at least the "if" property, the "rt" property, and shall additionally contain all mandatory properties required by the resource type to be created.

Note that the mandatory "rt" and "if" values are included in both the link parameters and the resource properties of the resource creation payload.

If the supplied representation does not contain all of the required elements, the server shall return a client error status indicating "Bad Request" (e.g. 400).

**Example Representation supplied to a CREATE operation**

|  |
| --- |
| {  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "ins": "thermostat-input",  "rep": {  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "temperature": 20  }  } |

The representation returned when a resource is successfully created shall contain the "href", "if", and "rt" link parameters, in addition to all link parameters and resource properties that were supplied by the client to the Create operation. The server may include additional link parameters and properties in the created resource as required by the application specific resource type.

**Example Response Representation returned from a successful CREATE operation**

|  |
| --- |
| {  "href": "tempinput"  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "ins": "thermostat-input",  "rep": {  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "temperature": 20  }  } |

**7.6.3.9.3 Use with CREATE**

CREATE may be performed with a client-provided URI for the created resource. The URI for the created resource may be provided as a query parameter using the identifier "href". The URI parameter supplied must not contain any "/" characters, that is it must be a relative URI from the collection resource URI.

**Example CREATE using client-supplied URI**

|  |
| --- |
| **Method:** POST  **URI:** /scenes/scene1/?href=tempinput&if=oic.if.create  **Payload:**  {  "ins": "thermostat-input",  "rep": {  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "temperature": 20  }  }  **Response:** 2.01Created  **Payload:**  {  "href": "tempinput"  "ins": "thermostat-input",  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "rep": {  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "temperature": 20  }  } |

CREATE may be performed with the server providing the URI for the created resource. The server may use any valid URI for the created resource as a relative URI from the collection resource URI.

**Example CREATE using server-supplied URI**

|  |
| --- |
| **Method:** POST  **URI:** /scenes/scene1/?if=oic.if.create  **Payload:**  {  "ins": "thermostat-input",  "rep": {  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "temperature": 20  }  }  **Response:** 2.01Created  **Payload:**  {  "href": "3755f3ac"  "ins": "thermostat-input",  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "rep": {  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "temperature": 20  }  } |

**7.6.3.9.3 Use with NOTIFY**

NOTIFY may be used with the Create Interface. Notifications sent to observers shall only be sent as a result of successful CREATE operations. Notifications on the Create Interface shall contain the required Response Representation payload for the CREATE operation, as illustrated in the example above. A successful NOTIFY shall return a status code indicating general success (e.g. HTTP 200) or a status code indicating successful creation (e.g. CoAP 2.01).

**Example NOTIFY on Create Interface**

|  |
| --- |
| **Response:** 2.01 Created  **Payload:**  {  "href": "3755f3ac"  "ins": "thermostat-input",  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "rep": {  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"],  "temperature": 20  }  } |

**7.6.3.10 Delete Interface**

**7.6.3.10.1 Overview**

The Delete Interface is used to delete resources from the server.

The identifier "oic.if.delete" is used in the "if" property of a resource or link to indicate that the resource supports the Delete Interface.

**7.6.3.10.2 Use with DELETE**

DELETE is performed using the URI of the resource to be deleted. Upon successful processing of a DELETE operation, the server shall return a response status code indicating general success (e.g. 200 for HTTP) or specific deleted status if supported (e.g. 2.02 for CoAP).

A successfully completed DELETE operation shall result in the resource pointed to by the URI being removed, an all child resources being removed. Successful removal shall result in a status code of "Resource Not Found" (e.g. 404) if any reference is attempted to a removed resource URI.

A successful DELETE operation shall result in all links in the parent collection resource pointing to the deleted resource being removed.

A successful DELETE operation shall return a response payload which includes the address of the deleted resource, relative to its parent resource.

**Example DELETE and response payload**

|  |
| --- |
| **Method:** DELETE  **URI:** /scenes/scene1/3755f3ac?if=oic.if.delete  **Payload:**  **Response:** 2.02Deleted  **Payload:**  {  "href": "3755f3ac"  } |

If a DELETE attempt is unsuccessful, the resource state shall be unchanged.

**7.6.3.10.3 Use with NOTIFY**

A client may use NOTIFY with the Delete Interface, by performing an Observe operation on the resource to be deleted. Notifications shall only be transmitted upon successful DELETE operations. The Notification payload shall contain the address of the deleted resource, relative to its parent resource.

**Example NOTIFY on Delete Interface**

|  |
| --- |
| **Response:** 2.02Deleted  **Payload:**  {  "href": "3755f3ac"  } |

**7.6.3.11 Link Update Interface**

**7.6.3.11.1 Overview**

The Link Update Interface is used to add, modify, or remove links in a collection.

The identifier "oic.if.linkupdate" is used in the "if" property of a resource or link to indicate that the resource supports the Link Update Interface.

To select links to Modify or Delete, the query parameter is used, with the same rules applied as for query filtering of links and batch items. See section TBD.

**7.6.3.11.2 Use with UPDATE**

To **Add Links** to a collection, the UPDATE operation is used with a payload containing one or more links.

When Links are added to a collection using the Link Update Interface, the server shall return a payload consisting of the added links.

**Example Add Links using the Link Update Interface**

|  |
| --- |
| **Method:** POST  **URI:** /scenes/scene1/?if=oic.if.linkupdate  **Payload:**  [  {  "anchor": "thermostat-input",  "href": "ocf://<…>/sensor/temperature",  "rel": ["boundto"],  "bind": "obs",  "rt": ["oic.r.temperature"],  "if": ["oic.if.s"]  }  ]  **Response:** 2.04Changed  **Payload:**  [  {  "anchor": "thermostat-input",  "href": "ocf://<…>/sensor/temperature",  "rel": ["boundto"],  "bind": "obs",  "rt": ["oic.r.temperature"],  "if": ["oic.if.s"]  }  ] |

To **Modify Links** in a collection, the UPDATE operation is used with a URI query parameter to select one or more links to modify. The payload is applied to each selected link according to the following rules:

1. If a key in the payload does not exist in the selected link, the key+value pair is added as a new link parameter.
2. If a key in the payload does exist in the selected link, the value is replaced with the value provided in the payload
3. If a key in the payload does exist in the selected link, and the value in the payload is null, the key+value pair is removed from the link

When one or more links are modified, the server shall return a payload consisting of the modified links.

**Example Modify Links using the Link Update Interface**

|  |
| --- |
| **Method:** POST  **URI:** /scenes/scene1/?if=oic.if.linkupdate&anchor="thermostat-input"  **Payload:**  [  {  "ins": "input-observe-link"  }  ]  **Response:** 2.04Changed  **Payload:**  [  {  "anchor": "thermostat-input",  "href": "ocf://<…>/sensor/temperature",  "rel": ["boundto"],  "bind": "obs",  "rt": ["oic.r.temperature"],  "if": ["oic.if.s"],  "ins": "input-observe-link"  }  ] |

To **Delete Links** in a collection, the UPDATE operation is used with a URI query parameter to select one or more links to delete. The payload for deleting links consists of the null value in an array, e.g. [null].

**Example Delete Links using the Link Update Interface**

|  |
| --- |
| **Method:** POST  **URI:** /scenes/scene1/?if=oic.if.linkupdate&anchor="thermostat-input"  **Payload:**  [null]  **Response:** 2.04Changed  **Payload:**  [null] |

**7.6.3.11.3 Use with NOTIFY**

A client may use NOTIFY with the Link Update Interface, by performing an Observe operation on the collection resource. Notifications shall only be transmitted upon successful UPDATE operations. The Notification payload shall contain the required response payload as specified above.

**Example NOTIFY on Modify Links using the Link Update Interface**

|  |
| --- |
| **Response:** 2.04Changed  **Payload:**  [  {  "anchor": "thermostat-input",  "href": "ocf://<…>/sensor/temperature",  "rel": ["boundto"],  "bind": "obs",  "rt": ["oic.r.temperature"],  "if": ["oic.if.s"],  "ins": "input-observe-link"  }  ] |