|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | | |
| **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 - Rules Definition*.docx | | |

|  |  |
| --- | --- |
| ***Title:*** ⌘ | Rules Definition |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***Submitter:*** ⌘ | (Samsung) Michael Koster | | | | | |
| ***Owner:*** ⌘ | | (Samsung) Michael Koster | | | | |
| ***Certification:*** ⌘ | X | |  | ***Date:*** ⌘ | | 2018-05-03 |
| ***Category:*** ⌘ | F | |  |  | |  |
|  | *Use one of the following categories:* ***C*** *(correction)* ***F***  *(functional modification of feature)* ***E*** *(editorial modification)* | | | |  | |
| ***Summary and Reason for change:*** ⌘ | | | Add Rules Definition | | | |
|  | | | | | | |
| ***Location of changes:*** ⌘ | | | 11.9 (new section) | | | |
|  | | | | | | |
| ***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

**11.9 Rules**

**11.9.1 Overview**

OCF Rules are OCF Resources that implement autonomous decision logic according to a simple input-condition-action pattern. The Rule inputs are derived from the property values of selected Resource instances, and the Rule actions consist of defined Update operations that are performed on selected Resource instances.

An OCF Rule has the following components

* One or more Rule Inputs, derived from of values of Properties of Resource instances
* One Rule Expression that defines the Rule logic in terms of the defined Rule Inputs, and which evaluates to a Boolean TRUE/FALSE value.
* One or more Rule Actions which are processed when the Rule Expression evaluates to TRUE
* One or more Dynamic Links that point to external resource instances to be used as sources for the Rule Inputs and destinations for Rule Actions.

Figure 27 shows how these components are organized with respect to the Rule resource.

**Figure 27. Components of an OCF Rule**



**11.9.2 Rule Structure and Behavior**

**11.9.2.1 Rule Structure**

A Rule is a resource that implements a collection of Links and Resources. A rule instance shall contain the following:

* One or more Rule Input Resources
* One Rule Expression Property
* One Boolean Rule Enable Property
* One Boolean Action Enable Property
* One Boolean Rule Result Property
* One or more Rule Action Resources
* Optional Dynamic Links that connect the Rule Inputs and Rule Actions to external Resources

The resource type is oic.r.rule.

Table 33 details the Properties exposed by the Rule Resource, consisting of a Rule Expression Property, a Rule Result Property, and a Links Property

**11.9.2.2 Rule Inputs**

Each Rule Input is a Resource Instance within the Rule collection that has a resource type defined by the Rule. For example, a Rule that evaluates a temperature input will include a Rule Input resource of type oic.r.temperature. Resoure Types for Rule Inputs should be compatible with, or identical to, the external Resources they are expected to be linked to.

Rule Inputs are indicated by the relation type "ruleinput" in the link pointing to the Resource instance.

**Figure 28. Example link to Rule Input Resource**

|  |
| --- |
| {  "href": "tempinput",  "rel": ["ruleinput", "item"],  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"]  } |

Whenever any Rule Input is updated, the Rule Expression is re-evaluated, potentially causing the Rule Actions to be processed.

Rule Inputs may be updated from external Resources by using Dynamic Links as described in Section 11.9.2.5.

**11.9.2.3 Rule Expression**

The Rule Expression is a string property of the Rule Resource that defines a logical expression over the Rule Inputs, and which evaluates to a Boolean Rule Result value. The expression shall conform to the syntax defined in Section 11.9.3.

Rule Inputs are specified by including the local URI name of the Rule Input Resource and including the desired Property name using the colon-delimited syntax described in Section 11.9.3. For example, for a Rule Input Resource named "setpoint" and a Property name "temperature", the name "setpoint:temperature" would be used in the Rule Expression to refer to this Rule Input.

**Figure 29. Example Rule Expression**

|  |
| --- |
| "tempinput:temperature >= setpoint:temperature" |

The Rule Expression shall be evaluated and the Rule Result shall be updated each time any Rule Input is updated, if the Rule Enable Property is set to TRUE. When the Rule Result changes from FALSE to TRUE, as the result of a Rule Input update and Rule Expression evaluation, all present Rule Actions shall be processed, if the Action Enable Property is set to TRUE.

All Rule Actions resulting from a rule evaluation shall be processed before any subsequent Rule Input updates are processed.

Rule Inputs may be obtained from external Resources by using Dynamic Links as described in Section 11.9.2.5.

**11.9.2.4 Rule Actions**

Each Rule Action is a Resource Instance within the Rule collection that has a resource type defined by the Rule. For example, a Rule meant to trigger a Scene would have a Rule Action Resource Type of oic.wk.scenecontrol with a lastScene property. Resource Types of Rue Actions should be compatible with, or identical to, any external resources they are expected to be linked to.

Rule Actions are indicated by the relation type "ruleaction" in the link pointing to the resource instance.

**Figure 30. Example link to Rule Action Resource**

|  |
| --- |
| {  "href": "scenecontrol",  "rel": ["ruleaction", "item"],  "rt": ["oic.r.scenecollection"],  "if": ["oic.if.rw","oic.if.baseline"]  } |

Rule Actions shall be processed when the result of evaluating the Rule Expression changes from FALSE to TRUE, if the Action Enable Property is set to TRUE.

Rule Actions may be used to update external Resources by using Dynamic Links as described in Section 11.9.2.5.

Processing a Rule Action results in an update operation being performed on the Rule Action resource. If the link pointing to the Rule Action does not define a "source' attribute, then the contents of the Rule Action resource shall be updated without changing the value. This shall result in a notification with the contents of the Rule Action Resource being sent to any observers, as well as updates being sent to any external Resources that are linked using push Dynamic Links.

If a link pointing to a Rule Action defines a "source" attribute, the contents of the resource pointed to by the "source" URI shall be transferred to the Rule Action Resource when the Rule Action is processed. This feature can be used to push Rule Input data (or other dynamically changing data) to an external URI using a Rule Action and a Dynamic Link.

**Figure 31. Example link to Rule Action with Rule Input source**

|  |
| --- |
| {  "href": "tempout",  "rel": ["ruleaction", "item"],  "source": "tempinput",  "rt": ["oic.r.temperature"],  "if": ["oic.if.a","oic.if.baseline"]  } |

**11.9.2.5 Rule Enable and Action Enable**

The Rule Enable Property controls whether the Rule Result Property is updated upon processing of the Rule Expression. If the Rule Enable Property is set to TRUE, then Rule Result shall be set according to evaluation of the Rule Expression as described in section 11.9.2.3 (this section). If the Rule Enable Property is set to FALSE, the Rule Result Property may be updated to force its value.

The Action Enable Property controls whether the Rule Actions are processed when the Rule Result state changes from FALSE to TRUE. If the Action Enable Property is set to TRUE, the Rule Actions will be processed when the Rule Result state is changed from FALSE to TRUE. If the Action Enable Property is set to FALSE, the Rule Actions will not be processed.

Setting the Rule Enable and Action Enable Properties can place the rule into one of four modes.

* Rule Enable FALSE, Action Enable FALSE = **Disable Mode**, Rule Expressions are not processed and Rule Actions are not processed. This is the recommended initial mode for a newly created rule.
* Rule Enable TRUE, Action Enable FALSE = **Rule Test Mode**. Rule Inputs may be updated and the Rule Result may be observed to test the logic and processing of the Rule Expression.
* Rule Enable FALSE, Action Enable TRUE = **Action Force Mode**. The Rule Result resource may be updated from FALSE to TRUE in order to manually trigger the processing of Rule Actions.
* Rule Enable TRUE, Action Enable TRUE = **Normal Operation**. The Rule Expression is processed and Rule Result is updated when Rule Inputs are updated, and Rule Actions are processed when the Rule Result value changes from FALSE to TRUE.

**11.9.2.6 Dynamic Links**

Dynamic Links are hyperlinks that define communication of state transitions from one Resource instance to another. A Dynamic Link has the relation type "boundto" and defines a data transfer from the link target resource to its context Resource, or anchor. Dynamic links also contain a target attribute "bind" which can have the value "obs" or "push".

A Dynamic link with the target attribute "bind": "obs" results in an observe operation being performed on the target resource of the link, with the context of the link being updated whenever a notification takes place.

A Rule Input may have one or more associated Dynamic Links that bind to external resources, using "bind": "obs" to observe the external Resource and update the Rule Input upon notifications from the external Resource.

**Figure 32. Example Dynamic Link from external Resource to Rule Input**

|  |
| --- |
| {  "href": "coap://[fec0::13]/temp",  "rel": ["boundto"],  "anchor": "tempinput",  "bind": "obs",  "rt": ["oic.r.temperature"],  "if": ["oic.if.a"]  } |

A Dynamic Link with a target attribute "bind": "push" results in an update operation being performed on the context resource of the link whenever the target resource is updated.

A Rule Action may have one or more associated Dynamic Links that bind to an external Resources, using "bind": "push" to update the external Resource when the Rule Action is processed.

**Figure 33. Example Dynamic Link from Rule Action to external Resource**

|  |
| --- |
| {  "href": "scenecontrol",  "rel": ["boundto"],  "anchor": "coap://[ 2001:db8:2::1]/scenes/scene1/lastScene",  "bind": "push",  "rt": ["oic.r.scenecollection"],  "if": ["oic.if.a"]  } |

**11.9.3 Rule Life Cycle**

Rules may be created and managed by a client acting in the role of an application level configuration tool or application orchestration tool.

It is expected that Rules are composed by creating templates for the Rule Resource and its Rule Input and Rule Action Resources, and creating the Rule Expression with the appropriate resource:property references.

This template can then be used to create an instance of the Rule on an appropriate device in a system. The template would be configured with appropriate initial values, or these values could be set after creating the Rule instance.

Separately, appropriate Resources are discovered or created for the Rule Input sources and Rule Action targets. Dynamic Links are added to the Rule to connect these external resources, and the Rule is now processing inputs as updates occur.

When the Rule is no longer needed, it can simply be deleted.

The life cycle of a Rule is shown in Figures 37, from creation through operation and removal.

**11.9.3.1 Create and activate a new Rule**

Rules are created dynamically by clients in order to orchestrate application logic. The ability to create rules is implemented by one or more top level Rules Collection Resource instances that include the "rts" value "oic.r.rule" and an "if" value of "oic.if.create". Creating and activating a new rule is described below. See Figures 34-36 for example representations.

1. An instance of a Rule is created in the top level Rule collection by using the oic.if.create interface and supplying a representation for the desired Rule, including the mandatory properties. Initial Values should be included in the created representation. The rule should be created with its rule processing initially disabled (ruleenable = false and actionenable = false).
2. The Rule Input Resources and Rule Action resources are created in the newly-created Rule instance using the oic.if.create interface. Initial values should be included for these resources upon creation. Figures 35 and 36 show examples of representations used to create these resource instances.
3. Any necessary Dynamic Links are added to include external resources in the rule. New links are added using the oic.if.linkupdate interface.
4. After all resources and dynamic links are created, the rule should be enabled by setting the Rule Enable Property and Action Enable Property to TRUE.
5. After the Rule Input resources are created, and a valid Rule Expression is included, and the Rule Enable Property is set to true, the Rule shall be evaluated and, when the Rule Result changes from FALSE to TRUE, and the Action Enable Property is set to TRUE, any Rule Actions that are present shall be processed. If the initial value of the Rule Result property is FALSE, and the first evaluation of the Rule yields a TRUE result, then the Rule Actions shall be processed.

**Figure 34. Example representation used to create a new Rule instance**

|  |
| --- |
| {  "rep":  {  "rt": ["oic.r.rule"],  "if": ["oic.if.rw", "oic.if.create", "oic.if.ll", "oic.if.linkupdate", "oic.if.baseline"],  "rts": ["oic.r.temperature", "oic.r.scenevalue"],  "rule": "tempinput:temperature >= tempsetpoint:temperature",  "ruleenable": false,  "ruleresult": false,  "actionenable": false,  "links": []  }  } |

**Figure 35. Example representation used to create a Rule Input Resource**

|  |
| --- |
| {  "rep":  {  "rt": ["oic.r.temperature"],  "if": ["oic.if.a", "oic.if.baseline"],  "temperature": 25,  }  } |

**Figure 36. Example representation used to create a Rule Action Resource**

|  |
| --- |
| {  "rep":  {  "rt": ["oic.r.scenecollection"],  "if": ["oic.if.a", "oic.if.baseline"],  "lastScene": "heat-off"  }  } |

**11.9.3.2 Evaluate the Rule Expression on updates to Rule Inputs and process Rule Actions**

Rule Inputs may be updated from external sources using UPDATE operations, or they may be configured with Dynamic Links to observe and autonomously receive updates from external resources. For example, a thermostat Rule may observe an external temperature sensor resource, and may also include Rule Inputs for setpoint and mode that are updated from a client application.

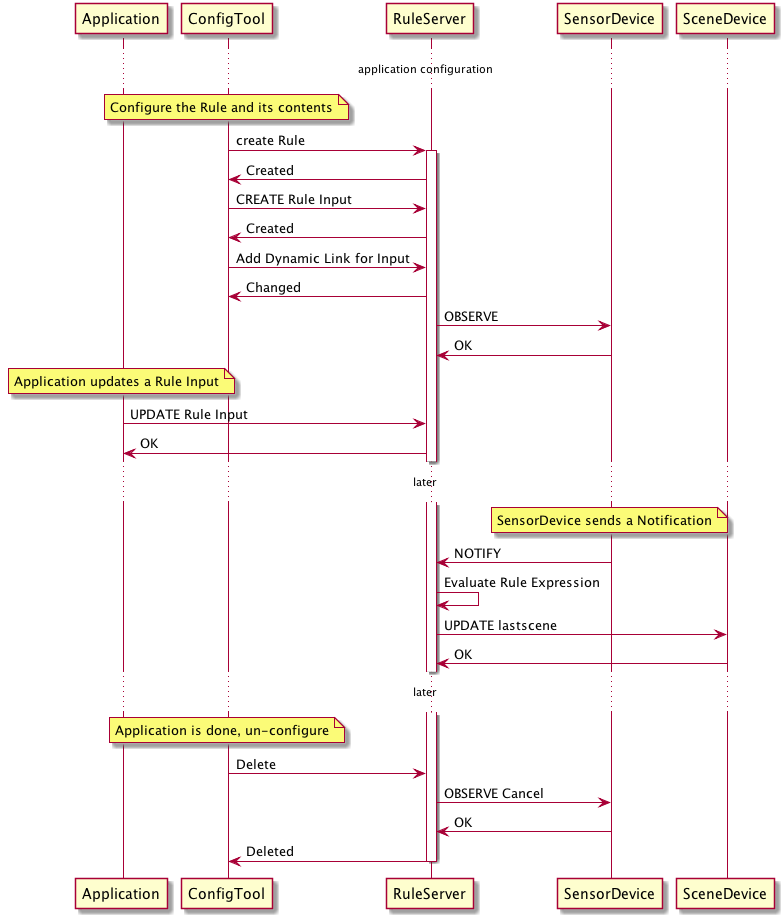
Whenever any Rule Input resource is updated, the Rule Expression shall be evaluated and the Rule Result shall be updated. If the Rule Result changes from FALSE to TRUE, the Rule Actions shall be processed.

**11.9.3.3 Remove a Rule when the application is reconfigured**

When the application is de-configured or re-configured, any rules that are no longer needed may be deleted. Performing a DELETE operation on the Rule instance resource shall immediately deactivate the rule and recursively remove all Rule Input and Rule Action Resources.

Upon deletion of a rule, any outstanding OBSERVE requests shall be cancelled.

**Figure 37. Example Life Cycle of a Rule**

****

**11.9.4 Rule Expression Syntax**

A Rule Expression consists of a string that conforms to the following syntax for "rule:

rule ::= ruleExp

ruleExp ::= relExp| ruleExp wChar+ logOp wChar+ ruleExp| '(' wChar\* ruleExp wChar\* ')'

logOp ::= 'and'|'or'

relExp ::= resourceproperty wChar+ binOp wChar+ quotedVal| resourceproperty wChar+ existsOp wChar+ boolVal

binOp ::= relOp|stringOp

relOp ::= '='|'!='|'<'|'<='|'>'|'>='

stringOp ::= 'contains'|'doesNotContain'|'startsWith'

existsOp ::= 'exists'

boolVal ::= 'true'|'false'

quotedVal ::= dQuote string dQuote

wChar ::= space|hTab|lineFeed|vTab|formFeed|return

resourceproperty ::= dQuote path-rootless dQuote

hTab ::= (\* UTF-8 code 0x09, horizontal tab character \*)

lineFeed ::= (\* UTF-8 code 0x0A, line feed character \*)

vTab ::= (\* UTF-8 code 0x0B, vertical tab character \*)

formFeed ::= (\* UTF-8 code 0x0C, form feed character \*)

return ::= (\* UTF-8 code 0x0D, carriage return character \*)

space ::= ' ' (\* UTF-8 code 0x20, space character \*)

dQuote ::= '"' (\* UTF-8 code 0x22, double quote character \*)

path-rootless ::= (see RFC3986)

**11.9.5 Rule Resource Properties**

Table 33 lists the defined Properties of the Rule Resource Type

Table 33. Rule Resource Property definition

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Property title | Property name | Value type | Value rule | Unit | Access mode | Mandatory | Description |
| Rule Expression | rule | string | ABNF Section 11.9.4 |  | RW | yes | Property that contains the logical expression that implements the rule logic |
| Rule Enable | ruleenable | boolean |  |  | RW | yes | Determines whether the Rule Result is updated from the rule expression |
| Action Enable | actionenable | boolean |  |  | RW | yes | Determines whether Rule Actions are processed |
| Rule Result | ruleresult | boolean |  |  | RW | yes | The Boolean result of the most recent evaluation of the rule |
| Link list | links | array | OCF Link |  | RW | yes | List containing all of the links to Rule Inputs, Rule Actions, and Dynamic Links |