**Demeter Server C2C API**

**V1.5**

|  |  |  |
| --- | --- | --- |
| Revision History | | |
| Version | Release Date | Description |
| 1.0 | 2019-12-16 | Initial release. |
| 1.1 | 2019-12-25 | 1. Add "Install App" API. 2. Add "error" notification when installing/patching IPK file of App to device failed. 3. Add "publisher" column to App information. 4. Modify desired arguments when requesting install/patch API. |
| 1.2 | 2020-01-22 | 1. Update URI of "install/patch App" API 2. Add "Get containers information" API 3. Add "Get a specific container information" API 4. Add "Uninstall App" API |
| 1.3 | 2020-06-11 | 1. Update device ID rule. 2. Add API to query all of the devices. 3. Add API to query the specific installed App information of specific device. 4. Add API to query all of the installed Apps information of specific device. 5. Update method to PUT when updating App for specific device. |
| 1.4 | 2021-09-09 | 1. Add error codes  2. {model\_name} moved to parameters in p.5.3.1  3. Correct application statuses for p. 3.3.1 |
| 1.5 | 2021-10-12 | 1. Add description for returned codes reasons  2. Add token authorization description |

**Table of contents**

[1. Introduction 5](#_Toc84948506)

[Generic Rule Declaration 5](#_Toc84948507)

[2. Demeter Server Notification 7](#_Toc84948508)

[2.1. Device Enroll and Delete Notification 7](#_Toc84948509)

[2.1.1. Flow Chart 7](#_Toc84948510)

[2.1.2. Notification Details 7](#_Toc84948511)

[2.2. App Installation or Update Progress Notification 8](#_Toc84948512)

[2.2.1. Flow Chart 8](#_Toc84948513)

[2.2.2. Notification Details 8](#_Toc84948514)

[3. Demeter Server API 10](#_Toc84948515)

[3.1. Get Devices 10](#_Toc84948516)

[3.1.1. API Details 10](#_Toc84948517)

[3.2. Get Device Information 13](#_Toc84948518)

[3.2.1. API Details 13](#_Toc84948519)

[3.3. Get Installed Apps Information of Specific Device 15](#_Toc84948520)

[3.3.1. API Details 15](#_Toc84948521)

[3.4. Get Installed App Information of Specific Device 18](#_Toc84948522)

[3.4.1. API Details 18](#_Toc84948523)

[3.5. Get Installable Apps Information of Device Model 20](#_Toc84948524)

[3.5.1. API Details 20](#_Toc84948525)

[3.6. Install or Update App 23](#_Toc84948526)

[3.6.1. API Details 23](#_Toc84948527)

[3.7. Uninstall (Delete) App 25](#_Toc84948528)

[3.7.1. API Details 25](#_Toc84948529)

[3.8. Start App 27](#_Toc84948530)

[3.8.1. API Details 27](#_Toc84948531)

[3.9. Stop App 29](#_Toc84948532)

[3.9.1. API Details 29](#_Toc84948533)

[3.10. Get All Containers Information of Device 31](#_Toc84948534)

[3.10.1. API Details 31](#_Toc84948535)

[3.11. Get Specific Container Information of Device 34](#_Toc84948536)

[3.11.1. API Details 34](#_Toc84948537)

[4. Appendix 36](#_Toc84948538)

[4.1. Unified Message Exchange Interface (UMEi) 36](#_Toc84948539)

[4.1.1. Objective 36](#_Toc84948540)

[4.1.2. Prerequisite 36](#_Toc84948541)

[4.1.3. Terminology 36](#_Toc84948542)

[4.1.4. Definition 36](#_Toc84948543)

[4.1.5. CRUDN Action 37](#_Toc84948544)

[4.1.6. Request Payload 39](#_Toc84948545)

[4.1.7. Response Payload 39](#_Toc84948546)

[4.1.8. Status Code 40](#_Toc84948547)

# Introduction

The purpose of Demeter **A**pplication **P**rotocol **I**nterface is for partners’ cloud services to integrate feature of OpenWRT compatible devices efficiently. Therefore, the partners’ cloud services can obtain the necessary functions to setup the devices without considering the complex handshaking or communicating flow charts.

## Generic Rule Declaration

Each of the API method follows rules listed as below:

1. The generic rules of Cloud2Cloud (C2C) API document were extended from [Unified Message Exchange Interface (UMEi) Design Guideline](#_Unified_Message_Exchange) defined by SERCOMM.
2. The "[Request ID](#_Request_Payload)" and "[Originator ID](#_Request_Payload)" are contained within each HTTP response header, named "**X-Request-ID**" and "**X-Originator-ID**".
3. "[Receiver ID](#_Response_Payload)" is contained within each HTTP response header, named "**X-Receiver-ID**".

**Token authentication**

1. On first launch Openfire generates a secret token.

2. If token authorization is disabled (sercomm.demeter.api.key parameter is empty) in the Openfire settings, token verification is not performed.

3. If token authorization is enabled (sercomm.demeter.api.key parameter contains token) in the Openfire settings, the server expects the parameter **“Authorization: Bearer <token>”** in HTTP request header.

4. Openfire verifies the received token: if it matches the generated token, the authorization was successful.

If the tokens do not match, the server sends a response with a HTTP code 401 "Unauthorized".

Examples of request with invalid token and response with “401 Unauthorized” :

GET /umei/v1/devices HTTP/1.1

Host: 10.110.134.212:8090

User-Agent: curl/7.74.0

Accept: \*/\*

Content-type: application/json

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: sercomm.com

X-Receiver-ID: sercomm.com

Authorization: Bearer 112233445566

HTTP/1.1 401 Unauthorized

Date: Tue, 12 Oct 2021 11:05:59 GMT

X-Request-ID:

X-Originator-ID:

X-Receiver-ID:

Content-Type: application/json

Access-Control-Allow-Origin: \*

Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept

Access-Control-Allow-Methods: GET, POST, PUT, DELETE, OPTIONS, HEAD

Access-Control-Max-Age: 3600

Access-Control-Allow-Credentials: true

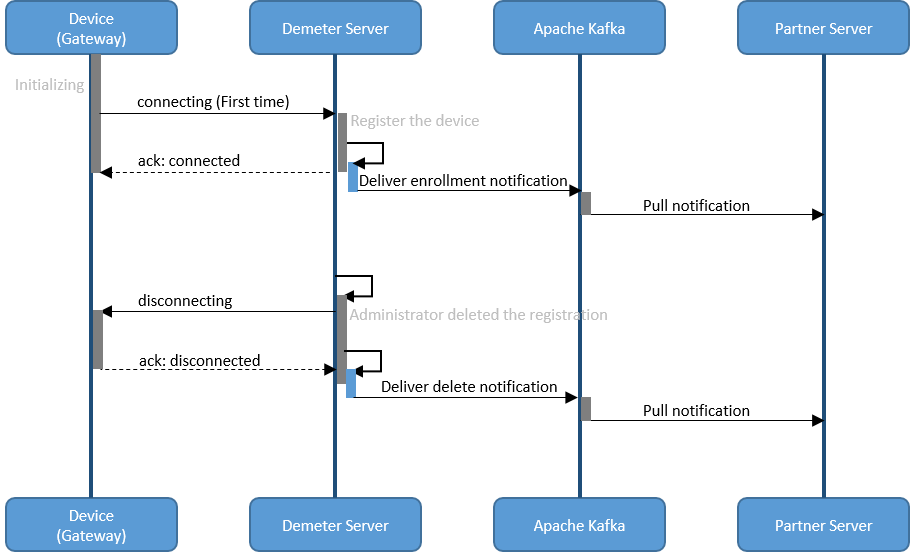
Content-Length: 64

{"errors":[{"code":401,"detail":"Unauthorized: Bad API Token"}]}

# Demeter Server Notification

## Device Enroll and Delete Notification

### Flow Chart



### Notification Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| PUT | (Specified by partner) | Send notifications to partner service when device is enrolled or deleted. |

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| serial | string | Yes | Serial number of the device. |
| mac | string | Yes | MAC address of the device. |
| model | string | Yes | Model name of the device |

* Desire Object

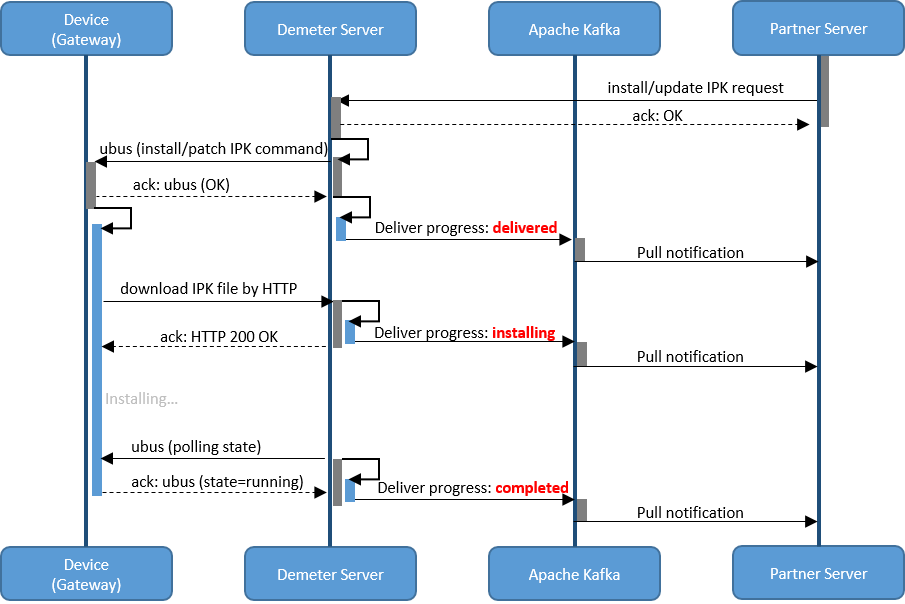
*No desire object will be provided.*

* An example of valid UMEi HTTP request:

PUT /rest/v1/example/uri HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...  
  
{  
 "data": {  
 "serial": "XX000XX00001",  
 "mac": "AABBCCDDEEFF",  
 "model": "HG4234B"  
 }  
}

## App Installation or Update Progress Notification

### Flow Chart



### Notification Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| PUT | (Specified by partner) | Send notifications to partner service for patching the IPK installation progress. |

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| serial | string | Yes | Serial number of the device. |
| mac | string | Yes | MAC address of the device. |
| taskId | string | Yes | ID of the patch task and was provided by partner server. Please refer to [section 3.3](#_Patch_Installed_App). |
| progress | string | Yes | Progress will be the following values: - **delivered**: command was delivered - **installing**: IPK is downloading and installing - **completed**: upgrade completely - **timeout**: upgrade task timeout - **fail**: error occurred when interacting with device. The details of error will be described in "error" field as below. |
| error | string | No | Detail error message when installing/patching. |

* Desire Object

*No desire object will be provided.*

* An example of UMEi HTTP notification:

PUT /rest/v1/example/uri HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...  
  
{  
 "data": {  
 "serial": "XX000XX00001",  
 "mac": "AABBCCDDEEFF",  
 "task": {  
 "taskId": "1066943f-839c-4be8-9397-56284790edd4",  
 "progress": "installing"  
 }  
 }  
}

* An example of UMEi HTTP notification with error report:

PUT /rest/v1/example/uri HTTP/1.1  
…

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...  
  
{  
 "data": {  
 "serial": "XX000XX00001",  
 "mac": "AABBCCDDEEFF",  
 "task": {  
 "taskId": "1066943f-839c-4be8-9397-56284790edd4",  
 "progress": "fail"  
 },  
 "error": "DEVICE WAS OFFLINE"  
 }  
}

# Demeter Server API

## Get Devices

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| GET | /umei/v1/devices | Get devices’ information from Demeter server. |

* **{deviceId}**: Device ID consists of its serial number and MAC address.

[Action Parameter(s)](#_Action_Parameter):

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Mandatory** | **Description** |
| from | No | Please refer to [Action Parameter](#_Action_Parameter). Default value is **0**. |
| size | No | Please refer to [Action Parameter](#_Action_Parameter). Default value is **100**. The maximum value is **500**. |
| sort | No | Please refer to [Action Parameter](#_Action_Parameter). Attribute name can be: - **serial:** Sort by serial number of devices. - **mac:** Sort by MAC address of devices. Default is "**serial:asc**" |
| models | No | Expected model names of devices. Multiple model names need to be separated by comma symbols, e.g. "**HG4234B,HG5244B**". Default is all models. |
| states | No | Expected devices’ current state. Multiple states need to be separated by comma symbol, e.g. "**online,offline**". Default is all states. |

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

*No desire object needs to be provided.*

* An example of valid UMEi HTTP Request:

GET /umei/v1/devices?from=0&size=100 HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...

[Response Payload](#_Response_Payload):

* Data Object

Device Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| serial | string | Yes | Serial number of the device. |
| mac | string | Yes | MAC address of the device. |
| model | string | Yes | Model name of the device. |
| state | string | Yes | The most current state of the device. State will be as the following values: - **offline** - **online** |
| firmware | string | Yes | The most current firmware version. |
| creationTime | string | Yes | Creation timestamp of device’s registration. |

* Possible error codes:

|  |  |
| --- | --- |
| **HTTP response code** | **Cause of error** |
| 403 FORBIDDEN | - The “size” parameter is greater than 500. |
| 500 INTERNAL\_SERVER\_ERROR | - Internal server error. |

* An example of Successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{  
 "meta": {  
 "from": 0,  
 "size": 2,  
 "total": 2,  
 },  
 "data": [  
 {  
 "serial": "XX000XX00001",  
 "mac": "AABBCCDDEEFF",  
 "model": "HG4234B",  
 "state": "online",  
 "firmware": "3.2.03\_T5",  
 "creationTime": "2019-12-12T00:11:00+0000",  
 },  
 {  
 "serial": "XX000XX00002",  
 "mac": "AABBCCDDEEEF",  
 "model": "HG4234B",  
 "state": "online",  
 "firmware": "3.2.03\_T5",  
 "creationTime": "2019-12-13T00:11:00+0000",  
 },  
 ]  
}

* An example of UMEi HTTP response with error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"errors": [{  
"code": 500,

"detail": "INTERNAL SERVER ERROR"

}]

}

## Get Device Information

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| GET | /umei/v1/device/{deviceId} | Get device’s information from Demeter server. |

* **{deviceId}**: Device ID consists of its serial number and MAC address.

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

*No desire object needs to be provided.*

* An example of valid UMEi HTTP Request:

GET /umei/v1/device/XX000XX00001-AABBCCDDEEFF HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...

[Response Payload](#_Response_Payload):

* Data Object

Device Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| serial | string | Yes | Serial number of the device. |
| mac | string | Yes | MAC address of the device. |
| model | string | Yes | Model name of the device. |
| state | string | Yes | The most current state of the device. State will be as the following values: - **offline** - **online** |
| firmware | string | Yes | The most current firmware version. |
| creationTime | string | Yes | Creation timestamp of device’s registration. |

* Possible error codes:

|  |  |
| --- | --- |
| **HTTP response code** | **Cause of error** |
| 400 BAD\_REQUEST | - Device ID has wrong format. |
| 403 FORBIDDEN | - Requested device was not found. |
| 500 INTERNAL\_SERVER\_ERROR | - Internal server error. |

* An example of Successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{  
 "data": {  
 "serial": "XX000XX00001",  
 "mac": "AABBCCDDEEFF",  
 "model": "HG4234B",  
 "state": "online",  
 "firmware": "3.2.03\_T5",  
 "creationTime": "2019-12-12T00:11:00+0000",  
 }  
}

* An example of UMEi HTTP response with error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"errors": [{  
"code": 404,

"detail": "DEVICE CANNOT BE FOUND"

}]

}

## Get Installed Apps Information of Specific Device

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| GET | /umei/v1/device/{deviceId}/apps | Get device’s installed Apps information from Demeter server. |

* **{deviceId}**: Device ID consists of its serial number and MAC address.

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

*No desire object needs to be provided.*

* An example of valid UMEi HTTP Request:

GET /umei/v1/device/XX000XX00001-AABBCCDDEEFF/apps HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...

[Response Payload](#_Response_Payload):

* Data Object

Installed App Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| appName | string | Yes | Name of the App. |
| publisher | string | Yes | Publisher name of the App. |
| appId | string | Yes | ID of the App. |
| status | string | Yes | Status of the install App:   * **Installed**: The App was installed but is not executing. * **Running**: The App was installed and is executing. * **Downloading:**  The App is downloading. * **Downloaded:** App download complete. * **Installing:** App installation in progress. * **Updating:** App is updating. * **Uninstalling:** App is uninstalling. |
| version | string | Yes | Installed version of the App. Please refer to Installed Version Object. |

Installed Version Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| versionName | string | Yes | Name of the version. |
| versionId | string | Yes | ID of the version |

* Possible error codes:

|  |  |
| --- | --- |
| **HTTP response code** | **Cause of error** |
| 400 BAD\_REQUEST | - Device ID has wrong format. |
| 403 FORBIDDEN | - Device was not found.  - Device is unavailable temporarily.  - Remote gateway is not available temporarily.  - Application was not found.  - Application version was not found. |
| 500 INTERNAL\_SERVER\_ERROR | - Internal server error. |

* An example of Successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{  
 "data": [  
 {  
 "appName": "helloworld\_1\_arm",  
 "publisher": "sercomm",  
 "appId": "52CC495F818A5615289D582EEE7AC825",  
 "status": "running",  
 "version": {  
 "versionName": "1.0",  
 "versionId": "22B10A14A77A931AA1FD183D28230D17"  
 },  
 {  
 "appName": "helloworld\_2\_arm",  
 "publisher": "sercomm",  
 "appId": "52CC495F818A5615289D582EEE7AC826",  
 "status": "installed",  
 "version": {  
 "versionName": "1.0",  
 "versionId": "22B10A14A77A931AA1FD183D28230D18"  
 },  
 ]  
}

* An example of UMEi HTTP response with error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"errors": [{  
"code": 404,

"detail": "DEVICE CANNOT BE FOUND"

}]

}

## Get Installed App Information of Specific Device

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| GET | /umei/v1/device/{deviceId}/app/{appId} | Get device’s installed App information from Demeter server. |

* **{deviceId}**: Device ID consists of its serial number and MAC address.
* **{appId}**: ID of the installed App.

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

*No desire object needs to be provided.*

* An example of valid UMEi HTTP Request:

GET /umei/v1/device/XX000XX00001-AABBCCDDEEFF/app/52CC495F818A5615289D582EEE7AC825 HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...

[Response Payload](#_Response_Payload):

* Data Object

Installed App Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| appName | string | Yes | Name of the App. |
| publisher | string | Yes | Publisher name of the App. |
| appId | string | Yes | ID of the App. |
| status | string | Yes | Status of the install App:   * **installed**: The App was installed but is not executing. * **running**: The App was installed and is executing. |
| version | string | Yes | Installed version of the App. Please refer to Installed Version Object. |

Installed Version Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| versionName | string | Yes | Name of the version. |
| versionId | string | Yes | ID of the version |

* Possible error codes:

|  |  |
| --- | --- |
| **HTTP response code** | **Cause of error** |
| 400 BAD\_REQUEST | - Device ID has wrong format. |
| 403 FORBIDDEN | - Application was not found.  - Device was not found.  - Device is unavailable temporarily.  - Remote gateway is not available temporarily.  - Application version was not found. |
| 500 INTERNAL\_SERVER\_ERROR | - Internal server error. |

* An example of Successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{  
 "data": {  
 "appName": "helloworld\_1\_arm",  
 "publisher": "sercomm",  
 "appId": "52CC495F818A5615289D582EEE7AC825",  
 "status": "running",  
 "version": {  
 "versionName": "1.0",  
 "versionId": "22B10A14A77A931AA1FD183D28230D17",  
 }  
 }  
}

* An example of UMEi HTTP response with error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"errors": [{  
"code": 404,

"detail": "APP WAS NOT INSTALLED"

}] }

## Get Installable Apps Information of Device Model

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| GET | /umei/v1/apps | Get available Apps information. |

[Action Parameter(s)](#_Action_Parameter):

|  |  |  |
| --- | --- | --- |
| **Parameter Name** | **Mandatory** | **Description** |
| model | yes | Model name of a device. |
| from | No | Please refer to [Action Parameter](#_Action_Parameter). Default value is **0**. |
| size | No | Please refer to [Action Parameter](#_Action_Parameter). Default value is **100**. The maximum value is **500**. |
| sort | No | Please refer to [Action Parameter](#_Action_Parameter). Attribute name can be: - **name** : Sort by name of an Apps. - **creationTime** :Sort by App creation time. Default is "**name:asc**" |

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

*No desire object needs to be provided.*

* An example of valid UMEi HTTP request:

GET /umei/v1/apps/HG4234B HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...

[Response Payload](#_Response_Payload):

* Data Object

Installable App Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| appName | string | Yes | Name of the App. |
| publisher | string | Yes | Publisher name of the App. |
| appId | string | Yes | ID of the App. |
| creationTime | string | Yes | Creation timestamp of the App. |
| versions | string | Yes | List of available versions. Please refer to Available Version Object. |

Available Version Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| versionName | string | Yes | Name of the version. |
| versionId | string | Yes | ID of the version. |
| creationTime | string | Yes | Creation timestamp of the version. |

* Possible error codes:

|  |  |
| --- | --- |
| **HTTP response code** | **Cause of error** |
| 400 BAD\_REQUEST | - Parameter 'model' is blank. |
| 403 FORBIDDEN | - The “size” parameter is greater than 500. |
| 404 NOT\_FOUND | - Model was not found. |
| 500 INTERNAL\_SERVER\_ERROR | - Internal server error. |

* An example of successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{  
 "data": [  
 {  
 "appName": "helloworld\_1\_arm",  
 "publisher": "sercomm",  
 "appId": "52CC495F818A5615289D582EEE7AC825",  
 "creationTime": "2019-12-12T00:01:00+0000",  
 "versions": [  
 {  
 "versionName": "1.0",  
 "versionId": "22B10A14A77A931AA1FD183D28230D17",  
 "creationTime": "2019-12-12T00:01:20+0000",  
 },  
 {  
 "versionName": "1.1",  
 "versionId": "219EF635A8DF081EF0505FD4C9F7A516",  
 "creationTime": "2019-12-12T00:01:30+0000",  
 },  
 ]  
 },  
 ]  
}

* An example of UMEi HTTP response with error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"errors": [{  
"code": 500,

"detail": "INTERNAL SERVER ERROR"

}]

}

## Install or Update App

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| POST | /umei/v1/device/{deviceId}/app | Install App with specific version to specific device. |
| PUT | /umei/v1/device/{deviceId}/app | Update the installed App to specific version of specific device. |

* **{deviceId}**: Device ID consists of its serial number and MAC address.

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| appId | string | Yes | ID of the available App. |
| versionId | string | Yes | ID of the available version.. |
| taskId | string | No | An unique ID for identifying the patch progress assigned by partner server. |

* An example of valid UMEi HTTP Request:

POST /umei/v1/device/XX000XX00001-AABBCCDDEEFF/app HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNEn  
...  
  
{  
 "desire": {  
 "appId": "2C0C0C020DB3712CDE1BEA0CD11FB822",  
 "versionId": "17A29771640C794FB87F02F22664F249",  
 "taskId": "fbec031d-59c0-436a-871e-e4cdc673f467"  
 }  
}

[Response Payload](#_Response_Payload):

* Data Object

*No data object will be provided.*

* *Possible error codes:*

|  |  |
| --- | --- |
| **HTTP response code** | **Cause of error** |
| 400 BAD\_REQUEST | - Device ID has wrong format.  - Request payload is blank.  - Request payload has wrong format.  - Payload parameter 'appId', 'versionId' or 'taskId' is blank.  - Invalid X-Request-Type. |
| 403 FORBIDDEN | - Application was not found.  - Application version was not found. |
| 406 NOT\_ACCEPTABLE | - Installation error. Additional information may be provided in response message. |
| 408 REQUEST\_TIMEOUT | - Application installation error, timeout expired. |
| 500 INTERNAL\_SERVER\_ERROR | - Internal server error. |

* An example of Successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

* An example of UMEi HTTP Response with error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"errors": [{  
"code": 403,

"detail": "APP CANNOT BE FOUND"

}]

}

## Uninstall (Delete) App

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| DELETE | /umei/v1/device/{deviceId}/app/{appId} | Uninstall (delete) App from specific device. |

* **{deviceId}**: Device ID consists of its serial number and MAC address.
* **{appId}**: ID of the installed App.

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

*No desire object needs to be provided.*

* An example of valid UMEi HTTP request:

DELETE /umei/v1/device/XX000XX00001-AABBCCDDEEFF/app/2C0C0C020DB3712CDE1BEA0CD11FB822 HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09d  
X-Originator-ID: FQDN-OF-PARTNER  
...

[Response Payload](#_Response_Payload):

* Data Object

*No data object will be provided.*

* *Possible error codes:*

|  |  |
| --- | --- |
| **HTTP response code** | **Cause of error** |
| 400 BAD\_REQUEST | - Device ID has wrong format. |
| 403 FORBIDDEN | - Device was not found.  - Device is unavailable temporarily.  - Application has not been installed.  - Database communication error.  - Remote gateway is not available temporarily, command is not executed successfully. |
| 500 INTERNAL\_SERVER\_ERROR | - Internal server error. |

* An example of successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

* An example of UMEi HTTP response with error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"errors": [{  
"code": 403,

"detail": "APP HAS NOT BEEN INSTALLED"

}]

}

## Start App

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| PUT | /umei/v1/device/{deviceId}/app/{appId}/start | Start the installed App of specific device. |

* **{deviceId}**: Device ID consists of its serial number and MAC address.
* **{appId}**: ID of the installed App.

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

*No desire object needs to be provided.*

* An example of valid UMEi HTTP request:

PUT /umei/v1/device/XX000XX00001-AABBCCDDEEFF/app/2C0C0C020DB3712CDE1BEA0CD11FB822/start HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...

[Response Payload](#_Response_Payload):

* Data Object

*No data object will be provided.*

* *Possible error codes:*

|  |  |
| --- | --- |
| **HTTP response code** | **Cause of error** |
| 400 BAD\_REQUEST | - Device ID has wrong format. |
| 403 FORBIDDEN | - Device was not found.  - Device is unavailable temporarily.  - Application was not found.  - Application was not installed.  - App already started.  - Remote gateway is not available temporarily. |
| 500 INTERNAL\_SERVER\_ERROR | - Internal server error. |

* An example of successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

* An example of UMEi HTTP response with error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"errors": [{  
"code": 403,

"detail": "APP HAS NOT BEEN INSTALLED"

}]

}

## Stop App

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| PUT | /umei/v1/device/{deviceId}/app/{appId}/stop | Stop the installed App of specific device. |

* **{deviceId}**: Device ID consists of its serial number and MAC address.
* **{appId}**: ID of the installed App.

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

*No desire object needs to be provided.*

* An example of valid UMEi HTTP request:

PUT /umei/v1/device/XX000XX00001-AABBCCDDEEFF/app/2C0C0C020DB3712CDE1BEA0CD11FB822/stop HTTP/1.1  
...  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...

[Response Payload](#_Response_Payload):

* Data Object

*No data object will be provided.*

* *Possible error codes:*

|  |  |
| --- | --- |
| **HTTP response code** | **Cause of error** |
| 400 BAD\_REQUEST | - Device ID has wrong format. |
| 403 FORBIDDEN | - Device was not found.  - Device is unavailable temporarily.  - Application was not found.  - Application was not installed.  - App already stopped.  - Remote gateway is not available temporarily. |
| 500 INTERNAL\_SERVER\_ERROR | - Internal server error. |

* An example of successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

* An example of UMEi HTTP response with error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"errors": [{  
"code": 403,

"detail": "APP HAS NOT BEEN INSTALLED"

}]

}

## Get All Containers Information of Device

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| GET | /umei/v1/device/{deviceId}/lxcs | Get all containers information of the specific device. |

* **{deviceId}**: Device ID consists of its serial number and MAC address.

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

*No desire object needs to be provided.*

* An example of valid UMEI HTTP request:

GET /umei/v1/device/XX000XX00001-AABBCCDDEEFF/lxcs HTTP/1.1  
X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc  
X-Originator-ID: FQDN-OF-PARTNER  
...

[Response Payload](#_Response_Payload):

* Data Object

List of Containers’ Information:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| id | string | Yes | ID of the container. |
| name | string | Yes | Name of the container. |
| enabled | boolean | Yes | Container was enabled or not. |
| resources | string | Yes | Hardware resources usage. Please refer to Resource Object. |

Resources Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| cpu | string | Yes | CPU usage status. Please refer to CPU Object. |
| memory | string | Yes | Memory usage status. Please refer to Memory Object. |
| storage | string | Yes | Storage usage status. Please refer to Storage Object. |

CPU Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| usage | number | Yes | CPU usage status in percentage. |

Memory Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| total | number | Yes | Total allocated memory size in byte unit. |
| free | number | Yes | Available memory size in byte unit. |

Storage Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| total | number | Yes | Total allocated storage size in byte unit. |
| free | number | Yes | Available storage size in byte unit. |

* An example of successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"data": [{  
"id": "security",  
"name": "security",  
"enabled": true,  
"resources": {  
 "cpu": {  
 "usage": 13.4,  
 },  
 "memory": {  
 "total": 442280,  
 "free": 142720  
 },  
 "storage": {  
 "total": 117760,  
 "free": 102656  
 }  
}

},

{  
"id": "webgui",  
"name": "webgui",  
"enabled": true,  
"resources": {  
 "cpu": {  
 "usage": 2.5,  
 },  
 "memory": {  
 "total": 442280,  
 "free": 256360  
 },  
 "storage": {  
 "total": 117760,  
 "free": 101520  
 }  
}

}]

}

* An example of UMEi HTTP response with error:

HTTP/1.1 200 OK

...

{

"errors": [{  
"code": 404,

"detail": "DEVICE CANNOT BE FOUND"

}]

}

## Get Specific Container Information of Device

### API Details

|  |  |  |
| --- | --- | --- |
| **HTTP Verb** | **URI Pattern** | **Description** |
| GET | /umei/v1/device/{deviceId}/lxc/{containerId} | Get specific container information of the specific device. |

* **{deviceId}**: Device ID consists of its serial number and MAC address.
* **{containerId}**: ID of target container.

[Action Parameter(s)](#_Action_Parameter):

*No action parameter will be provided.*

[Request Payload](#_Request_Payload):

* Data Object

*No data object needs to be provided.*

* Desire Object

*No desire object needs to be provided.*

* An example of valid request:

GET /rest/v1/device/XX000XX00001-AABBCCDDEEFF/lxc/security HTTP/1.1  
...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

...

[Response Payload](#_Response_Payload):

* Data Object
* Containers Information:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| id | string | Yes | ID of the container. |
| name | string | Yes | Name of the container. |
| enabled | boolean | Yes | Container was enabled or not. |
| resources | string | Yes | Hardware resources usage. Please refer to Resource Object. |

* Resources Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| cpu | string | Yes | CPU usage status. Please refer to CPU Object. |
| memory | string | Yes | Memory usage status. Please refer to Memory Object. |
| storage | string | Yes | Storage usage status. Please refer to Storage Object. |

* CPU Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| usage | number | Yes | CPU usage status in percentage. |

* Memory Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| total | number | Yes | Total allocated memory size in byte unit. |
| free | number | Yes | Available memory size in byte unit. |

* Storage Object:

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute Name** | **Data Type** | **Mandatory** | **Description** |
| total | number | Yes | Total allocated storage size in byte unit. |
| free | number | Yes | Available storage size in byte unit. |

* An example of successful UMEi HTTP response without error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"data": {  
"id": "security",  
"name": "security",  
"enabled": true,  
"resources": {  
 "cpu": {  
 "usage": 13.4,  
 },  
 "memory": {  
 "total": 442280,  
 "free": 142720  
 },  
 "storage": {  
 "total": 117760,  
 "free": 102656  
 }  
}

}

}

* An example of UMEi HTTP response with error:

HTTP/1.1 200 OK

...

X-Request-ID: 7fe79306-366a-4098-bbb8-9ea5c77c09dc

X-Originator-ID: FQDN-OF-DEMETER-SERVER

X-Receiver-ID: FQDN-OF-PARTNER

...

{

"errors": [{  
"code": 404,

"detail": "DEVICE CANNOT BE FOUND"

}]

}

# Appendix

## Unified Message Exchange Interface (UMEi)

This chapter describe the guideline of SERCOMM cloud platforms API.

### Objective

1. Define message format.
2. Keep rules be self-described and as simple as possible.
3. Rules defined here is not especially bound with any specific application layer protocol. How to adapt rules to specific protocol should be described in another documents.

### Prerequisite

1. Default naming convention should be fully compliant with lower camel case throughout all UME related documents if does not specify explicitly.
2. Variables defined in this guideline are identified within namespace “<http://www.sercomm.com/2018/ume/v1>”.

### Terminology

* **Action Parameter:** Information which are sent with a CRUDN requests to specify execution criteria.
* **Actor:** An actor who is either sending requests or replying response to corresponding requests.
* **Attribute:** An attribute is a piece of information to represent a data or controllable attributes which are applied to resource.
* **CRUDN:** CRUDN actions specify how a resource will be processed, actions manipulate resource in different ways with specified parameters, controllable and data attributes.
* **Entity:** An entity is either a physical device or a software component.
* **Resource:** A resource represents a type of object which is able to be managed by actors which invoke APIs on resources with a specific CRUDN action. Resources are described by data attributes which associates with a kind of object.

### Definition

#### Resource

Resource is composed of a set of attributes which represents a physical or logical entity. Resources could be created, read, updated, deleted or control by CRUDN actions with corresponding action parameters and action payload.

Attributes of resource may represent state of device element or controllable device elements. Attributes are categorized into three access authorities for different operation purposes.

##### Read-Only

Entities report resource with a set of read-only attributes to inform receiver current state of resource, it could be used as result of action or event notification.

##### Write-Only

A Write-only attribute should not appear in response as query result, but it is used in a request to alter state of resource or instruct resource on how to execute operation. An actor may control entities by altering the attributes of resources.

##### Read-Write

A Read-Write attribute has the features of both Read-Only and Write-Only.

Writable attributes are controllable elements, actor cannot retrieve state of resource from writable attributes but it is allowed to alter writable resource attributes to directly change the behavior of entity. A resource should contain following common attributes.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Attribute Name** | **Short Form** | **Date Type** | **Must** | **Read/Write** | **Description** |
| resourceId | r | string | Yes | Read-Only | A system-wide unique identity to recognize a resource. The fully qualified expression represents a resource could be varied and depends on the application layer protocol. It should be discussed in separate document. |
| timestamp | t | datetime | Yes | Read-Only | The latest updated time of resource. The value expression should conform with "**YYYY-MM-DDThh:mm:ss.sss±hhmm**" defined by ISO8601, ±hhmm is the zone designator representing time offset from UTC. |
| version | v | positive number | No | Read-Only | Each resource is given a version number and version number is returned as part of resource information for READ action. It should be increased for Create/Update/Delete actions. It will be used to control the version of the resource the action is intended to be executed against when version attribute is specified. |

Additional attributes for specific type of resource may be varied for representation or control purposes and it is beyond the scope of this guidline and should be defined in another documents.

#### Version System

The versioning system implements **O**ptimistic **C**oncurrency **C**ontrol transactions (aka. Optimistic locking) and involves resource manipualation in a distributed system.

By default, internal versioning is used that starts at 1 on created and increments with each update, deletes. The system checks to see if the version number passed to the request is identical to the version of the currently stored resource. If true, the document will be updated and the increase version number by 1. If the value provided is not identical to the stored resource’s version number, a version conflict error will occur and the operation will fail and status code 409 should be replied. **If no version is provided, then the operation is executed without any version checks**.

### CRUDN Action

Action indicates a entity how to manipulate specified resources with specific parameters and attributes. A action operation is compposed of a originator, a receiver, a request message and a optional response message. A originator invokes an action upon a receiver by sending a request message to the receiver, The receiver act on specified action and reply response message to receiver.

Five verbs are defined below for CRUDN actions.

|  |  |
| --- | --- |
| **CRUDN Action** | **Description** |
| CREATE | Create a resource. |
| READ | Read one or multiple resources. |
| UPDATE | Update an existing resource or create a ne resource. Update attribute value of resource may imply a device control process. |
| DELETE | Delete one or multiple resources. |
| NOTIFY | Send a notification to receiver, no response is expected. |

Following example represents a possibility to map HTTP verbs to CRUDN actions mentioned above.

|  |  |  |  |
| --- | --- | --- | --- |
| **HTTP Verb** | **CRUDN Action** | **Collection: /orders** | **Instance: /orders/{id}** |
| GET | READ | Read a list of orders. | Read the detail of a single order. |
| POST | CREATE | Create a new order. | **─** |
| PUT | UPDATE/CREATE/NOTIFY | **─** | Full update or create a specific order. |
| PATCH | UPDATE | **─** | Partial update a order. |
| DELETE | DELETE | **─** | Delete a order. |

Two actors will involved each action operation, originator and receiver. A **originator** is responsible to compose **requests**, with specific action assigned, and send it to receivers via specific application layer protocol. And a **recevier** should act on received request and reply originator either processing result or error within **response**.

#### Target URI

An **U**nified **R**esource **I**dentifier is designed for unambiguous identification of resources that specifies the means of acting upon. In order to easily differentiate between resoure URI and action parameters, the default naming convention for URI is compliant with [lower snake case](https://en.wikipedia.org/wiki/Snake_case).

#### Action Parameter

Action parameters are used to specify the conditions to execute action, entity can specifies query parameters to filter or restrict the range of replied result or other operation instructions. For example, an originator invokes READ action to retrieve information from receiver according to specifiy execution criteria.

Some common parameters defined hereunder are shared among actions, additional parameters may be defined for specific operations should be discussed individually in another document.

| **Name** | **Date Type** | **Must** | **Description** |
| --- | --- | --- | --- |
| begin | datetime | No | Specify the begin time of specific time range to restrict the number of queried resources. The value expression should conform with "**YYYY-MM-DDThh:mm:ss.sss±hhmm**" defined by ISO8601, ±hhmm is the zone designator representing time offset from UTC. |
| end | datetime | No | Specify the end time of specific time range to restrict the number of queried resources. The value expression should conform with "**YYYY-MM-DDThh:mm:ss.sss±hhmm**" defined by ISO8601, ±hhmm is the zone designator representing time offset from UTC. |
| size | number | No | The number of resources to be returned. |
| from | number | No | The offset of 1st resource to be returned. |
| sort | string | No | Sorting to perform, can be either a **attributeName** or **attributeName:asc**,**attributeName:desc** (sequence is important). default order is **asc**. |
| include | string | No | Set to false to disable retrieval of resource attributes. You can also specify attributes to retrieve part of the resource , e.g. **attributeName \*("," attributeName)**. |

#### Message Payload

Action payload is information which is mandatory for resource processing with specified action and parameters. Action payload may contain resources, metadata of resource and information to describe what kind data a payload is providing. Payload exists in both **request and response** and discussed in next sections.

### Request Payload

Request payload is an object includes information which a originator sends it to receiver in order to instruct which operation should be prcessed accordingly. Request payload may be not required for action and is divided into several categories based on the characteristics of information it contains.

The following table represents the relationshiop between categories and actions.

| **Information Category** | **Short Form** | **Data Type** | **Create** | **Read** | **Update** | **Delete** | **Notify** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| data | d | object | **+** | **─** | **+\*** | **─** | **+** |
| desire | ds | object | **─** | **─** | **+\*** | **─** | **─** |
| requestId | ri | string | **+\*** | **+\*** | **+\*** | **+\*** | **+\*** |
| originatorId | oi | string | **+\*** | **+\*** | **+\*** | **+\*** | **+\*** |

**Legends: +:**Mandatory, **+\*:** Optional, **─:** Not including

* **data:** Data is a key-value map which includes at least one or multiple resources or arrays of resources to represent readable attributes of resources. Key is the name of resource in either singular or plural form. The data type of value should be array of resources if key is naming in plural and value type is object for singular.
* **desire:** Desire is a key-value map which includes at least one or multiple resource objects or arrays of resource to represent writable attributes of resources.
* **requestId:** An ID to identify a unique request sent from same originator.
* **originatorId:** An unique ID to identify the originator.

### Response Payload

Response payload is anobject includes status code and information which a receiver replies originator to inform the result of execution of action. Response payload is divided into several categories based on the characteristics of information it contains.

The following table represents the relationshiop between categories and actions.

| **Information Category** | **Short Form** | **Data Type** | **Create** | **Read** | **Update** | **Delete** | **Notify** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| data | d | object | **+** | **─** | **+\*** | **─** | **─** |
| errors | e | array | **+\*** | **+\*** | **+\*** | **+\*** | **─** |
| meta | m | object | **─** | **+\*** | **─** | **─** | **─** |
| receiverId | rc | string | **+\*** | **+\*** | **+\*** | **+\*** | **─** |
| requestId | ri | string | **+\*** | **+\*** | **+\*** | **+\*** | **─** |
| originatorId | oi | string | **+\*** | **+\*** | **+\*** | **+\*** | **─** |

**Legends: +:**Mandatory, **+\*:** Optional, **─:** Not including

* **data:** Data is a key-value map which includes at least one or multiple resources or arrays of resources to represent readable attributes of resources. Key is the name of resource in either singular or plural form. The data type of value should be array of resources if key is naming in plural and value type is object for singular.
* **errors:** Errors is an array of error objects which represents the status of executions, an error object contains two attributes.
* ***code****:* Status code indicates the result of execution (see status code definitions in following chapters).
* ***detail****:* A description of error to make it more easy to read for human readers.

In some implementations, e.g. API gateway in Microservice, a response message might be replied by receiver who invokes multiple upstream services and merge them into a single response message. In this case, there might be existing multiple error objects in errors to represent each status of execution.

* **meta:** Meta provides information about data that are encouraged to provided by CRUDN actions for different purposes and helps originators to understand how to manipulate data well.

The following table is the example case which defines the pagination metadata of READ action.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Attribute Name | Short Form | Data Type | Must | Read/Write | Description |
| from | fr | positive number | Yes | Read-Only | The offest between 1st data item in current response and whole matched data items. |
| size | sz | positive number | Yes | Read-Only | The number of data items. |
| total | tt | positive number | Yes | Read-Only | Total number of whole matched data items. |

* **receiverId:** An unique ID to identify the receiver.
* **requestId:** Replicated from requestId of request if it is existed.
* **originatorId:** Replicated from originatorId of request if it is existed.

### Status Code

Status code are returned in the Response payload to indicate the result of request processing. The status codes are categorized as one of 6 classes as the following table.

|  |  |  |
| --- | --- | --- |
| Status Class | Code Class | Description |
| Informational | 1XX | The request is successfully received, but the request is still on process. |
| Success | 2XX | The request is successfully received, understood, and accepted. |
| Redirection | 3XX | (Not used in present release) |
| Originator Error | 4XX | The request was malformed by the originator and is rejected. |
| Receiver Error | 5XX | The requested operation cannot be performed due to an error condition at the Receiver. |

#### Information Code

|  |  |
| --- | --- |
| Numeric Code | Abbreviation |
| 100 | ACCEPTED |

#### Success Code

|  |  |
| --- | --- |
| Numeric Code | Abbreviation |
| 200 | OK |
| 201 | CREATED |
| 202 | DELETED |
| 204 | UPDATED |

#### Originator Error Code

|  |  |
| --- | --- |
| Numeric Code | Abbreviation |
| 400 | BAD\_REQUEST |
| 401 | NOT\_SUPPORTED |
| 404 | NOT\_FOUND |
| 405 | ACTION\_NOT\_ALLOWED |
| 408 | REQUEST\_TIMEOUT |
| 409 | VERSION\_CONFLICT |

#### Receiver Error Code

|  |  |
| --- | --- |
| Numeric Code | Abbreviation |
| 500 | INTERNAL\_SERVER\_ERROR |
| 501 | NOT\_IMPLEMENTED |
| 502 | BAD\_GATEWAY |
| 503 | GATEWAY\_TIMEOUT |