IoT Platform API Reference
(Version 7.0_0)

<Revision History>

Version No.	Description	Date
Version 1.0	First version	2016/10/07
Version 1.1	Updates based on K5 launch on UK site	2016/11/01
Version 1.2	Error Correction Deleting "$orderby" in retrieval of resource Adding Number of JSON array elements in "2.1 Maximum/Minimum Parameter Values" Correction of error response in "3.2 Retrieving Resource_JSON Data" Adding note about Timestamp of last_modified in "7.2 Referencing Resource Metadata"	2016/12/01
Version 1.3	Correction of usable characters as Resource Path "2.1 Maximum/Minimum Parameter Values" Correction of cross reference error	2017/01/12
Version 1.4	Correction of the BODY text of API calling example "10.1 Calling APIs"	2017/02/08
Version 4_1.0	-	Internal Version (Not released)
Version 4_2.0	Function enhancements Added notes on partial acquisition functionality (Range Request) to sections 2.3.1 and 2.3.2 Added notes on "CORS (Cross-Domain Resource Sharing) support" to section 2.4 Added notes on using hierarchy_get permissions to sections 3.3 and 3.5 Added notes on client certificate-related matters to section 8.1 and 8.2 Added notes on hierarchy_get and hierarchy_put permissions to section 8.1 Added notes on system restrictions concerning the frequency of connections in section 2.2.7 Added section on reference data stored by the system in Chapter 11	2017/08/01
Version 5.0_0	Function enhancements Added notes on registering data of csv/text/binary format, and compressed transmission at registering data to section 3.1 and 3.4 Added notes on searching data for array format to section 3.3	2017/11/15
Version 5.1_0	Function enhancements Added notes on relation between API request and error information to section 2.1, 2.4.1, 3.1, 3.4, 3.5, 4.1 and 11.3 Modified minimum value of "Number of skip items" at section 2.1 Added notes on sorting of retrieving data to section 3.3 Deleted TLS 1.1 of SSL support at section 2.3.1 and 2.5.1 Addition of adding Added notes on adding for query to section 3.1, 3.2, 3.3, 3.4 and 3.5	2018/3/17
Version 5.1_1	Error Correction in section 2.1, 11.3.	2018/4/11
Version 5.1_2	Descriptions add Modified chapter title in chapter 10 Added notes on Requests and required specifications of Response to section 10.1 Added description of API call specification of Resource(Transfer)_JSON to section 10.3	2018/6/13

Version 7.0_0	Function enhancements Added notes on retrieving data with data compression to section 2.3.1, 2.3.2, 2.4.1 and 2.4.2 Added notes on relation between API request and error information to section 2.4.1, 5.1 and 5.5 Deleted restriction of 'reason-phrase' requirement in 'status-line' at section 10.1 Added new error_factor_message at section 11.3	2018/9/15

Preface

Thank you for considering the "IoT Platform Service (hereafter, this service)".
This "IoT Platform API Reference (hereafter, this manual)" is intended for customers considering or implementing this service. We ask for your understanding regarding the following matters.

1. Customers considering this service are kindly requested to utilize this manual to assist in making their decision to implement this service only.
2. This manual and the contents therein are not to be disclosed or provided to any third parties.
3. Copying or reproducing the contents of this manual without the permission of the provider is prohibited.

This manual contains important information to be used in implementing this service.
Customers signing up to a service contract are asked to thoroughly read this manual prior to using this service.
Please handle this manual with care and store it in a safe place.
Customers opting to not use this service are responsible for promptly disposing of this manual.

While we have striven to prepare this manual with the utmost of care in describing tasks in the most easy- to-understand manner as possible, we cannot be held responsible for any errors or omissions in the content of this manual. This manual and the contents therein may change at any time without notice.

The contents of this manual cannot be copied, reproduced or modified, in part or in full, without prior permission to do so.

Disclaimers

- We do not accept any responsibility for unexpected malfunctions or for unforeseen charges occurring due to the user performing operations not listed in this manual of this service.
- We do not accept any responsibility in the unlikely event that the use, or inability to use, this service causes damage to the user (including, but not limited to, damages caused by a suspension of work, damage to/loss of data, or accident- related damages, and including the potential for liability claims from a third party).

<Terms>

Term	Description	Notes
IoT	An abbreviation of "Internet of Things" This refers to "things" connected to a network via an Internet protocol (Internet language)".	
REST	An abbreviation of REpresentational State Transfer This refers to a software design format where design principles optimized for linking multiple software instances are adapted for the web. *A caller interface (referred to as a "RESTful API") sends messages written in XML to a specific URL via HTTP(s). When using this service replies are sent back in JSON format, not XML.	
MQTT	An abbreviation of Message Queuing Telemetry Transport MQTT is a light communications protocol on the TCP/IP network suited to frequently sending and receiving short messages between multiple subjects. The use of MQTT is now very common in M2M networks and the IoT (Internet of Things) field. At a minimum the header is only two bytes in size, significantly reducing the amount of communications traffic, CPU load and energy consumption required, compared to sending the same communications via HTTP.	
Dynamic resource controller (DRC)	Dynamic Resource Controller Provides distributed control processing based on proprietary wide- area distribution technologies. This function assists in the optimal collection of data from a limited number of resources based on traffic fluctuation during data collection.	
Resources	The collection unit for IoT data	
Resource data	One piece of data	
Access code	Authorization information for resources	

- Contents -

Chapter 1 Introduction

1.1. Purpose of this Manual

This manual is an "API Reference" intended for persons developing applications using APIs based on the use of the IoT Platform (hereafter, this service).

1.2. Available Documents

The following manuals have been prepared to support customers using this service.

Manual name	Description
IoT Platform Service Details Instruction Manual	Describes service specifications.
IoT Platform User Guide	A manual used to support API usage during application design and using this service, including specific examples.
IoT Platform API Reference	An API reference manual used for application design and using this services. (This manual)
IoT Platform Service Portal Operating Manual	A manual describing web interface functionality (hereafter, the Service Portal).

Memo

Refer to Chapter 3 of the “IoT Platform Service Portal Operating Manual” for more information about resources, access codes and other general service definitions and concepts
For Dynamic Resource Controller (DRC) please refer to “IoT Platform User Guide”.

Chapter 2 Common Items

Data maintained within this service can be accessed with REST (HTTP) and MQTT protocols. This chapter covers common matters for such.

2.1. Maximum/Minimum Parameter Values

Table 1: The maximum and minimum parameter values specified for REST (HTTP) headers, URI and MQTT Topics are described in this table.

Table 2: The maximum and minimum values for filter condition parameters when performing searches with REST (HTTP) are described in this table.

Table 3: The maximum and minimum values for filter condition parameters when registering, referencing, updating and deleting access codes and events with API operations are described in this table.

Table 4: The maximum and minimum values for resource data control interfaces for select condition details are described in this table.

Table 5: The maximum and minimum values for parameters contained in resource data unpacked to the REST BODY or MQTT Payload are described in this table.

Table 1: List of maximum and minimum values for external interfaces

Parameter name	Description	Usable characters	Minimum	Maximum
Resource path	Unique path given to represent a resource. Express different levels by separating them with a / (slash). *The initial "/" is not included in the resource path.	- Single byte alphanumeric characters - "- ", "_", "/" *As with "/- " and "/_", a "- " or "_" cannot be specified after a "/" or as the initial character. *Consecutive "/" symbols, such as "//", cannot be specified.	2 characters	128 characters (Please note that, for resource (transfer), "_fwd" is included, and for Resource_Binary, "_bin" is included in the maximum of 128 characters)
Extension	A filename extension showing the resource data format	json, csv, text, bin	3 characters	4 characters
compression type	Type of compression when sending compressed data	gz only	2 characters	2 characters
Access code	An identifier used to show external IF access permissions	Only single-byte alphanumeric characters	3 characters	48 characters
Registration timestamp	Registration timestamp given/granted to registration data. Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.).	- Single byte alphanumeric characters and, + or - - The only alphabetical characters permitted are T, when used between the date and the time, and Z, used to represent UTC.	16 characters	24 characters

	Millisecond- level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).			
RETAIN	Determines whether to retain this registration data on the MQTT broker side. - true: Retain - false: Do not retain	true or false	4 characters	5 characters
filter conditions	Objects for processing are limited to only those that match the <filter condition> set. A <filter condition> is set as the "property name operator condition value", capable of defining multiple filter conditions by "and" and "or". "_date" can be added to the property name to show the registration timestamp.	Same as the character strings available for name and value described below. *Entries surrounded by single quotation marks (') are treated as character strings, and those without are treated as numerical values.	6 characters	256 characters
Number of Top items	Limits the amount of data returned as search results.	Single byte numerals	1 item	1000 items
Number of skip items	Skips a specified number of data items returned as search results.	Single byte numerals	0 item	100,000 items
Selection key	Selection key for Select conditions $select=<Selection key> Only returns data for fields specified with the <Selection key>. The <Selection key> is equivalent to the name in JSON format, and the element name and attribute name in XML format.	- Same as the character strings available for name described below. - Multiple instances of the <Selection key> can be specified by separating each with a comma ",". E.g.) $select=data.no,data.name	8 characters	256 characters
Content- Type	Body MIME type	Same as RFC2046, RFC6838, RFC4289, RFC6657.	0 characters	128 characters
x-iotpf-request-id	Request identification	%x21- %x 7E (Printable US- ASCII characters)	0 characters	64 characters
x-iotpf-meta-data1, x-iotpf-meta-data2, x-iotpf-meta-data3	Resource data metadata	%x21- %x 7E (Printable US- ASCII characters)	1 character	128 characters
Number of JSON array elements	Number of array elements when JSON array is described to Body part	-	-	1000

Table 2: List of maximum and minimum values for resource data control interfaces for filter condition details

Parameter name	Description	Usable characters	Minimum	Maximum
Property name	JSON- format resource data names	- All Unicode characters (UTF- 8), excluding " " ", "\"(*1), " ' ", "&", "$", "(", ")" control codes - Use percent- encoding for characters other than unreserved URI characters ("Single byte alphanumeric characters", "- ", ".", "_", "~"). - Do not use "_" as the initial character - "and", "or", "eq", "ne", "lt", "le", "gt", and "ge" are not yet supported as name values. Use <name>.<name> expressions when names have a hierarchical structure. Maximum depth of 15	1 character	128 characters
Condition values	Condition values for "value" in JSON-format resource data	- All Unicode characters (UTF- 8), excluding " " ", "\"(*1), " ' ", "$", control codes - Use percent- encoding for characters other than unreserved URI characters ("Single byte alphanumeric characters", "- ", ".", "_", "~"). - Entries surrounded by single quotation marks (') are treated as character strings, and those without are treated as numerical values. - null shows that the value does not exist.	For character string, max 256 characters	
			For integer, - 999999999999999 to 999999999999999	
			For real numbers, double-precision floating-point number	
Comparative operators	eq (equals sign), ne (inequality sign), gt (greater than), ge (greater than or equal), lt (less than), le (less than or equal)	Either eq, ne, gt, ge, lt, le	1 time	8 times (Number of incidences)
Logical operators	and (logical AND), or (logical OR). Use "()" for logical operators to use expressions like (A eq 1 and B eq 1)or(A eq 2 and B eq 2). However, in these instances () cannot be defined when inside (). For example, ((A eq 1 and B eq 1)or(A eq 2 and B eq 2))and(C eq 1) would return an error.	Either and, or	0 times	8 times (Number of incidences)

(*1) U+005C(REVERSE SOLIDUS)

Table 3: List of maximum and minimum values for access code control and event control interfaces for filter condition details

Parameters	Description	Usable characters	Minimum	Maximum
Property name	Filter Property name	_resource_path	1 time each (Number of incidences)	1 time each (Number of incidences)
Condition values	Condition values for each property name. Prefix matches with startwith() can be used.	- All Unicode characters (UTF- 8), excluding " " ", "\"(*1), " ' ", "$" control codes - Use percent- encoding for characters other than unreserved URI characters ("Single byte alphanumeric characters", "- ", ".", "_", "~"). - Only target character strings, and enclose entries with single quotation marks (').	For character strings, max 256 characters	
Comparative operators	eq (equality)	eq only	-	3 times (Number of incidences)
Logical operators	and (logical product) *"()" cannot be used as the only logical operator is and	and only	-	2 times (Number of incidences)

(*1) U+005C(REVERSE SOLIDUS)

Table 4: List of maximum and minimum values for resource data control interfaces for select condition details

Parameters	Description	Usable characters	Minimum	Maximum
Selection key	JSON- format resource data names	- All Unicode characters (UTF- 8*1), excluding " " ", "\"(*1), " ' ", "&", "$", "(", ")" control codes - Use percent- encoding for characters other than unreserved URI characters ("Single byte alphanumeric characters", "- ", ".", "_", "~") - Do not use "_" as the initial character - If names have a hierarchical structure, express this using <name>.<name> Maximum depth of 15	1 character	128 characters
Specifying multiple selection keys	Multiple instances of the <Selection key> can be specified by separating each with a comma ",".	","	-	10 times (Number of incidences)

(*1) U+005C(REVERSE SOLIDUS)

Table 5: List of maximum and minimum values for Body resource data control interfaces

Parameter name	Description	Usable characters	Minimum	Maximum
-	Characters that can be contained within the BODY (When using Bulk Insert function)	Must be in JSON format array	0 Bytes	16 Mbytes (Size when converting from JSON to BSON format)
-	Characters that can be contained within the BODY	Must be data form corresponding to the filename extention - json or no extention JSON format - csv CSV file format - txt Plain text file format - bin Any byte stream	0 Bytes	256 Kbytes (Size after decompression at compressed transmission)
key	JSON- format resource data keys	- All Unicode characters (UTF- 8), excluding " " ", "\"(*1), " ' ", "&", "$", "(", ")" control codes - Do not use "_" as the initial character - "and", "or", "eq", "ne", "lt", "le", "gt", "ge" not supported as names. - Maximum depth of key hierarchies is 15	1 character	128 characters
value	JSON- format resource data values	- All Unicode characters (UTF- 8), excluding " " ", "\"(*1), " ' ", "$" control codes - Entries surrounded by double quotation marks (") are treated as character strings, and those without are treated as numerical values - null shows that the value does not exist. - true/false shows boolean value.	For character string, max 256 charactors	
			For whole numbers, - 999999999999999 to 999999999999999	
			For real numbers, double-precision floating-point number. (However, the display form is not necessarily guaranteed.)	

(*1) U+005C(REVERSE SOLIDUS)

2.2. Notes

2.2.1. About API Charge

In this service, charge will be calculated according to the usage frequency of number of resource ID, API, SSL and evet on pay per use basis.

2.2.2. About Controlling resources, access codes, and events with API

In order to add, reference, update and delete resource, access code or event with the API, you will need create, delete and list permissions for those resources and access codes. API control is possible only of those resource paths which have access codes with create, delete and list permissions defined.

Please register higher-layer resources and access codes from the Service Portal to control resources and access codes beneath them with the API.

2.2.3. About REST/MQTT compatibility for the same Resource

In "Resource _ JSON", the resource data registered with REST can be referred by MQTT with MQTT SUBSCRIBE.

However, in "Resource _ Binary", the resource data registered with REST cannot be referred by MQTT with MQTT SUBSCRIBE.

2.2.4. About JSON Key when registering or updating Resource/Access code/Event control interface

Keys other than the JSON key described in this document can not be used. If used then it is responded as 400 Bad Request.

And, if a JSON key is set which is necessary for other control interfaces, also then it is responded as 400 Bad Request. For example, if "fwd_info" is set as the key in Request BODY when registering resource data to "Resource_JSON", then it is responded as "400 Bad Request" as "fwd_info" is necessary for Resource(transfer)_JSON.

2.2.5. About Resource path

Resource path is defined as follows.

- "Resource_JSON" : <arbitrary>

- "Resource(transfer)_JSON" : _fwd/<arbitrary>

  Resource path includes the prefix control character "_fwd/" too.

- "Resource_Binary" : _bin/<arbitrary>

  Resource path includes the prefix control character "_bin/" too.

2.2.6. About reflection time of Resource, Access code or Event setting

The system takes around 5 minutes (maximum) to reflect the change when addition/change/deletion of Resource, Access code or Event is executed from Service Portal or by control API operation. (However, this time is subject to change by future enhancement etc.).

2.2.7. System Restrictions Concerning Connection Frequency

All requests will return an error or data will be discarded if connection frequency exceeds a set time guideline. The occurrence of such events can be confirmed through the following. If this occurs, consider altering the frequency of connections, or change the payment plan.

Connection method	Event occurring	Confirmation method
REST (HTTP)	Error returned	Confirm via HTTP response. Occurs when 429 Too Many Requests appears.
MQTT	Discarding of data	Confirm via the error log storage function. Occurs when error log storage shows "trigger":"MP", "error_factor_message":"Failed to execute publishing resource data, in the system internal processing."

2.3. REST (HTTP) Common Items

2.3.1. Request

Parameters	Value	Additional notes
Base URL	https://<zone>.fujitsu.com	Follow the notification received during contract

	or http://<zone>.fujitsu.com	initiation for <zone> etc., Base URL value

- Use the 80/tcp port for http
- Use the 443/tcp port for https (TLS 1.2 supported)
- [Note]
  - Percentage encoding is not required for the URL path (part before the "?") (Percentage encoding not available)

Headers	Value	Additional notes	M/O(*1)
Authorization	Bearer <Access code>	The access code is the value set on the Service Portal	M
Accept-Encoding	gzip	Set this if you want to retrieve compressed data (payload) by gzip	O(*2)
Range	bytes=<lead position>- bytes=<lead position>-<end position> bytes=-<size> *You can specify multiple ranges using a comma to separate them	Set this if you only want to acquire a certain part of the data *A multi-part response will be returned when multiple ranges are specified.	O(*3)

(*1) M: Mandatory, O: Option
(*2) Only available for 3.2Retrieving Resource_JSON Data and 3.3Searching Resource_JSON Data
(*3) Only available for 5.2 Referencing Resource_Binary Data

2.3.2. Response

Status- Code	Reason- Phrase	Description
200	OK	Successfully created resource data
201	Created	Successfully created resource, access code or event
204	No Content	Either of the following case: -If specified resource exists, but target data does not exist when referencing resource data., -When successfully deleting a resource -If requested information does not exist when referencing metadata, access code or event. (this might be changed in future enhancement) -When successfully deleting metadata, access code or event.
206	Partial content	Partial acquisition successful
400	Bad Request	Invalid value found in request data
401	Unauthorized	Resource access permission not found
403	Forbidden	No access permission
404	Not Found	Resource not found
405	Method Not Allowed	The method not allowed
408	Request Time-out	The request is time-out
409	Conflict	Conflicted with another resource
411	Length Required	Server access denied (content-length not specified)
412	Precondition Failed	Server access denied (request condition incorrect)
413	Payload Too Large	Server access denied (requested body size exceeds capacity)
414	URI Too Long	Server access denied (URI is too long)

415	Unsupported Media Type	Server access denied (content-type not supported)
416	Requested Range Not Satisfiable	Server access denied (range request value incorrect)
421	Misdirected Request	Transmitted to the server that cannot generate response
423	Locked	Resource is locked.
429	Too Many Requests	Exceeds contracted traffic capacity
495	SSL Certificate Error	Invalid client certificate received
496	SSL Certificate Required	Client certificate was not sent from the client
497	HTTP Request Sent to HTTPS Port	HTTP request was received by HTTPS Port.
500	Internal Server Error	Failure due to server error
501	Not Implemented	Request method not supported in the server transmitted
502	Bad Gateway	Gateway server not activated
503	Service Unavailable	Service temporarily unavailable
504	Gateway Time-out	Gateway server could not return response in time

Headers	Value	Additional notes
Content-Encoding	gzip	Assigned when retrieved data (payload) is compressed by gzip (*1)
Content- Length	Body size (bytes)	-
Content-Range	<Start position>-<End position>-<Overall length>	Assigned by 206 response. Assigned for each part when multiple range specified in range request.

(*1) This header is assigned only if Accept-Encoding is set in request of 3.2 Retrieving Resource_JSON Data or 3.3 Searching Resource_JSON Data, and the size of retrived data before compression is larger than 1024 bytes.

✧ Expect detailed information in the following format when an error occurs

Headers	Value	Additional notes
Content- Type	application/json or application/json; charset=UTF-8	Fixed

✧ Body

```
{"errors": [{
        "message": "<Message>",
        "<Enter name here>": "<Enter value here>"
}]}
```

Parameter name	Value	Additional notes	M/O(*1)
Message	Detailed error information	-	M
name used	name containing additional information on the error	Specify a name based on the error	O
value used	value based on the <name used>	-	O

(*1) M: Mandatory, O: Option

Details are set in the message body when an error occurs. Refer to Appendix 1 for further details.

2.4. CORS (Cross-Origin Resource Sharing) Support

Generally, the Same-Origin Policy prevents REST (HTTP) requests being sent to domains other than those generating web pages in a web browser. However, IoT Platform provides support for CORS(*1) (Cross-Origin Resource Sharing) as a means of allowing secure cross-domain access.

(*1) As regulated by the W3C. (http://www.w3.org/TR/cors/)

Two connection protocols apply when performing cross-domain access based on CORS.

- One involves directly sending a cross-domain access request (access request).
- The other involves sending a preflight request (a request to confirm whether cross-domain access is possible) and, after a response has been received, sending a cross-domain access request (access request).

Note the CORS support is only provided for REST (HTTP) access. CORS support is not available for accessing the Service Portal.

2.4.1. Preflight Request

Request

Method	Value	Additional notes
Options	-	-

Headers	Value	Additional notes	M/O(*1)
Origin	<Origin server domain>	-	M
Access-Control-Request-Method	Either POST, PUT, GET or DELETE	Specify the method used by REST (HTTP) for the access request.	M
Access-Control-Request-Headers	Authorization	-	M
	Content-Type	Reference the following.	O
	Range	Reference the following.	O
	x-iotpf-meta-data1 x-iotpf-meta-data2 x-iotpf-meta-data3	Reference the following.	O
	x-iotpf-request-id	Reference the following.	O
	Accept-Encoding	Reference the following	O

(*1) M: Mandatory, O: Option

- Supplementary items on Access-Control-Request-Headers
  - Separate each header with a comma "," when specifying multiple headers.
  - Content-Type is required when the access request is as follows.
    1. Register resource (Section 7.1)
    2. Updating Resource Metadata (Section 7.3)
    3. Registering Access Codes (Section 8.1)
    4. Updating Access Codes (Section 8.3)
    5. Register event (Section 9.1)
    6. Updating Event Information (Section 9.3)
    7. Registering Data to Resource_JSON/Transferring Data with Resource (Transfer)_JSON (Section 3.1)

8. Updating Resource_JSON Data (Section 3.4)
9. Registering Data to Resource_Binary (Section 5.1)

✧ x-iotpf-meta-data1, x-iotpf-meta-data2, x-iotpf-meta-data3 and Range are required when the access request is as follows.

1. Controlling Resource_Binary Data(REST) (Chapter 5)

✧ x-iotpf-request-id is required when the access request is as follows.

1. Registering Data to Resource_JSON/Transferring Data with Resource (Transfer)_JSON (Section 3.1)
2. Updating Resource_JSON Data (Section 3.4)
3. Deleting Resource_JSON Data (Section 3.5)
4. Registering Data to Resource_Binary (Section 5.1)
5. Deleting Resource_Binary Data (Section 5.5)

✧ Accept-Encoding is required when the access request is as follows

1. Retrieving Resource_JSON Data (Section 3.2)
2. Searching Resource_JSON Data (Section 3.3)

➢ An error response (400 Bad Request) will be returned if the Origin header and other required headers are not added when CORS has been enabled in the Service Portal.

Body	Value	Additional notes
None	-	-

Response

Status-Code	Reason-Phrase	Description
Refer to Section 2.3.2	-	-

Headers	Value	Additional notes
Access-Control-Allow-Origin	*	Fixed (All origins allowed)
Access-Control-Allow-Methods	OPTIONS,POST,PUT,GET,DELETE	Fixed
Access-Control-Allow-Headers	Authorization	Fixed
	Content-Type	Fixed
	Range	When Controlling Resource_Binary Data (REST)
	x-iotpf-meta-data1 x-iotpf-meta-data2 x-iotpf-meta-data3	When Controlling Resource_Binary Data (REST)
	x-iotpf-request-id	When Controlling Resource_JSON/Resource (Transfer)_JSON Data (REST) and Controlling Resource_Binary Data (REST)
	Accept-Encoding	When Controlling Resource_JSON/Resource (Transfer)_JSON Data (REST)
Access-Control-Max-Age	86400	Fixed (24 hours)

➢ Separate each header with a comma "," when specifying multiple headers to Access-Control-Request-Headers.

Body	Value	Additional notes
Body	Success: None	-

	Failure: Found (Failure error information)	

2.4.2. Access Requests

Differences when using CORS compared to regular requests are detailed below. Refer to Section 2.3 for notes on other common matters.

Request

Add the following headers to the Request for cross-domain access.

Headers	Value	Additional notes	M/O(*1)
Origin	<Origin server domain>	-	M

(*1) M: Mandatory, O: Option

Response

Headers	Value	Additional notes
Access-Control-Allow-Origin	*	Fixed (All origins allowed) Only add when approved for CORS request. * Set whether the CORS function can be used from the Service Portal. CORS requests are only approved when the CORS function is in use.
Access-Control-Expose-Headers	Location Content-Length Content-Range x-iotpf-meta-data1 x-iotpf-meta-data2 x-iotpf-meta-data3 Content-Encoding	Fixed Only add when approved for CORS request.

2.5. MQTT Common Items

This Service supports MQTT v3.1.

2.5.1. Request

- MQTT broker address

MQTT broker address	Port number	SSL/TLS
<zone>.fujitsu.com	1883/TCP	
<zone>.fujitsu.com	8883/TCP	Yes

Follow the notification received during contract initiation for the value to use in place of the above sample MQTT broker address.

For SSL/TLS, TLS1.2 is supported

- Available message types

Message type	Value	Description	Availability
Reserved	0	Reserved	No
CONNECT	1	Connection request from the client to the server	Yes
CONNACK	2	Connection test response	Y&N (Cannot send)
PUBLISH	3	Issue message	Yes
PUBACK	4	PUBLISH confirmation response (QoS=1)	Yes

PUBREC	5	PUBLISH confirmation response (QoS=2, part1)	Yes
PUBREL	6	PUBLISH state cancel request (QoS=2, part 2)	Yes
PUBCOMP	7	PUBLISH state cancel request (QoS=2, part 3)	Yes
SUBSCRIBE	8	Subscription request	Yes
SUBACK	9	SUBSCRIBE confirmation response	Y&N (Cannot send)
UNSUBSCRIBE	10	Cancel subscription request	Yes
UNSUBACK	11	UNSUBSCRIBE confirmation response	Y&N (Cannot send)
PINGREQ	12	PING request	Yes
PINGRESP	13	PING response	Y&N (Cannot send)
DISCONNECT	14	Disconnection notification	Yes
Reserved	15	Reserved	No

- Fixed header settings

Parameters	Value	Additional notes	Applicable message type
DUP flag	Can be set to 0 or 1.	0: Initial message 1: Resent message	PUBLISH PUBREL SUBSCRIBE UNSUBSCRIBE
QoS flag	Can be set to 0 or 1 or 2. However, if session break occurs then setting 1 or 2 will have equivalent result of setting 0. * This represents QoS between the MQTT client and the broker, not from end to end.	0: Highest once 1: Lowest once 2: Accurately once	PUBLISH SUBSCRIBE
RETAIN flag	Can be set to 0 or 1.	0: Do not retain latest information with MQTT broker 1: Retain latest information with broker	PUBLISH

- Variable header settings

Parameters	Value	Additional notes	Applicable message type
Protocol name	MQIsdp	Fixed	CONNECT
Protocol version	3	Fixed	CONNECT
User name flag	1	Fixed	CONNECT
Password flag	1	Fixed	CONNECT
Will RETAIN	0	Fixed	CONNECT
Will QoS	0	Fixed	CONNECT
Will flag	0	Fixed	CONNECT
Clean Session flag	1	Fixed	CONNECT
Keep- alive	1 second or more,	Will disconnect with	CONNECT

timer	1800 seconds or less	an error (0x05:Connection denied) when a value over 1800 seconds, or 0 seconds, is set Disconnected due to error (0x05: Connection denied)	
Topic name	<Accesscode>/v1/<TenantID>/<Resourcepath>(*1)	-	PUBLISH SUBSCRIBE UNSUBSCRIBE

(*1) The <access code> part is used to judge permit/deny of the access. In case of appropriate <access code> is set, MQTT clients can exchange message between them when /v1/<Tenant ID>/<Resource path> is same.

- Payload settings
  - CONNECT

Parameters	Value	Maximum length	Additional notes
Client Identifier	String to identify client uniquely	23 characters	Set for each client without duplication.
User Name	User name	10 characters	This is the tenant ID received during contract initiation. (Tenant ID is fixed)
Password	Password	12 characters	Set from the Service Portal

  - PUBLISH
    - Resource data registration message
  - SUBSCRIBE
    - List of topics available to subscribe to
      - Topic name set as "<Access code>/v1/<Tenant ID>/<Resource path>".
  - UNSUBSCRIBE
    - List of topics to cancel subscriptions for
      - Topic name set as "<Access code>/v1/<Tenant ID>/<Resource path>".

- MQTT topic name wild cards

  Supports the following patterns of wild card expressions. An error will appear and you will be disconnected when receiving a SUBSCRIBE/UNSUBSCRIBE message that uses an unsupported wild card expression.

No.	Pattern	Description	Example of matching resource path
1	<Access code>/v1/<Tenant ID>/#	All resource paths within a <Tenant ID> tenant All resource paths	DC/ DC/aaa/Tokyo DC/bbb/Tokyo/1 DC/aaa/Fukuoka/1 DC/bbb/Fukuoka All of the above and more.
2	<Access code>/v1/<Tenant ID>/DC/aaa/#	Resource path prefix match	DC/aaa DC/aaa/Tokyo DC/aaa/Fukuoka/1
3	<Access code>/v1/<Tenant ID>/DC/+/Tokyo	Resource path partial match	DC/aaa/Tokyo

    - The "<Access code>/v1/<Tenant ID>/" part is required.
    - The following wild card patterns are not supported.
      - Wild card patterns where "+" appears multiple times

- Combinations of "#" and "+"
- When "+" appears at the end

2.5.2. Response

- Fixed header
  - None in particular

- Variable header

Parameters	Value	Additional notes	Applicable message type
Connection return code	0: Connection permitted 1: Connection denied (protocol version not accepted) 2: Connection denied (identifier denied) 3: Connection denied (server use not allowed) 4: Connection denied (invalid user name/password) 5: Connection denied (no permissions)	-	CONNACK

- Payload
  - None in particular

Chapter 3 Controlling Resource_JSON/Resource (Transfer)_JSON Data (REST)

A table showing the REST interface used to control Resource_JSON and Resource (Transfer)_JSON data is provided below.

Table 6: Data storing/reference interface list

Protocol	Type	Purpose	Chapter
REST	Registering data to Resource_JSON/ Transferring data with Resource (Transfer)_JSON	Registers (transfers) new data to a resource. [Option] Resource_JSON: - None: Registers data based on the timestamp when data is received by this service. - Specify time/date: Registers data based on a set timestamp. - Bulk Insert flag set: Registers data with bulk insert * 'Bulk insert' refers to registering multiple data items with a single API call. Resource (Transfer)_JSON: - None: Data is transferred to URLs restricted within this service. This functions as an MQTT broker when a URL is not set.	Chapter 3.1
	Referencing resource data	References data registered to a resource. [Option] - None: Returns the latest data. - Specify time/date: Returns data with the timestamp specified.	Chapter 3.2
	Retrieving resource data	Set search conditions and retrieve data registered to a resource. Registered data that matches the search conditions is returned as search results, along with additional registration timestamp information maintained by this service. [Option] - You can specify search conditions with QUERY.	Chapter 3.3
	Updating resource data	Updates data registered to a resource. [Option] - Specifying new registration timestamp: Changes the registration timestamp when updating data. - None: Only the data is updated and the registration timestamp will remain unchanged.	Chapter 3.4
	Deleting resource data	Deletes data registered to a resource. [Option] - You can specify deletion conditions with QUERY.	Chapter 3.5

3.1. Registering Data to Resource_JSON/Transferring Data with Resource (Transfer)_JSON

- Request

Parameters		Value
Method		PUT
URI	Data storing	<Base URL>/v1/<Tenant ID>/<Resource path>.<Extension>?$date=<Registration timestamp>&$retain=<RETAIN>&$bulk=<Bulk Insert flag>&$charset=<Character set of Body data>&$skip=<deleting lines of Body data>&$numconv=<Numerical conversion>
	Transferring data	<Base URL>/v1/<Tenant ID>/_fwd/<Resource path>.<Extension>?$date=<Registration timestamp>&$retain=<RETAIN>&$charset=<Character set of Body data>&$skip=<lines of deleting Body data>&$numconv=<Numerical conversion>

Headers	Refer to Section 2.3.1
Body	Data for registering

Memo

It is a specification that uses PUT Method in the registering data of IoT Platform, so from the idea "storing data to the existing resource" . POST Method to the resource is used by register resource(7.1 Register resource).

- URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for registering data	M
Extension	Specifies data format from either of following. When omitted (including .) will consider this as json. - json - csv - txt - bin	O
Compression type	Specifies compression type from following when transmitting data is compressed. When omitted (including .) will consider that data is not compressed. - gz	O
Registration timestamp	Registration timestamp given to registration data(*2) - This can be omitted (including $date=). When omitted, this service will use the request received timestamp. - This will be added without checking whether there is data with a pre-specified registration timestamp.	O
RETAIN	Determines whether to retain this registration data on the MQTT broker side. - true: Retain - false: Do not retain Can be omitted (including $retain=) when false *In case of Bulk Insert, RETAIN specification will be neglected (data will not be retained).	O
Bulk Insert flag	Specifies whether or not to run a bulk insert (sending multiple requests at once.). It is effective when the extension is only json or an omission. ● none: Do not perform bulk insert ● single_resource_path: Perform a bulk insert for a single resource Can be omitted when none (include $bulk=)	O
Character set of Body data	Specifies character set of Body data from either of following. It is effective when the extension is only csv or txt. - utf-8 : UTF-8 - shift_jis : Shift-JIS When omitted (including $charset=) will consider this as utf-8.	O
Deleting lines of Body data	Specifies number of lines to delete top of Body data. It is effective when the extention is only csv. When omitted (including $skip=), no lines are deleted.	O
Numerical conversion	Specifies whether or not to convert numerical figure into numerical value. It is effective when the extension is only csv. - true : numerical figure will convert into numerical value. - false : numerical figure will consider as string.	O

(*1) M: Mandatory, O: Option
(*2) Conforms to ISO8601 standards (standard millisecond expressions). (20141225T103612.001Z, etc.)
"Registration timestamps" all have the same specifications for millisecond-level precision (when omitting milliseconds, the system will read this as 0 milliseconds) and on.

*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

1. In queries of URI parameter, request does not become error if queries are not described in above but are described in any of the other request in this chapter (Retrieving, Searching, Updating, Deleting). Request with other queries become error.

- HTTP Header

Headers	Value	Additional notes	M/O(*1)
Content-Type	- When the extention is json or omitted application/json; charset=UTF-8	Fixed	O
	- When the extention is csv text/csv; charset=<character set> <character set> is either of following - UTF-8 - Shift_JIS		
	- When the extention is txt text/plain; charset=<character set> <character set> is either of following - UTF-8 - Shift_JIS		
	- When the extention is bin MIME type of Body		
	- When the compression type is gz (the extention is any of then) application/gzip or application/x-gzip		
x-iotpf-request-id	Request identification (any string)(*2)	It is used for error information output when the request is lost by error in this system or forwarding system. Please Refer to section 11.3 for error information.	O

(*1) M: Mandatory, O: Option
(*2) It operates assuming that x-iotpf-reqeust-id is not specified when only "-" is set to the value

- Body (When the extention is json or omitted)
  1. Common
     Member order not guaranteed. The order at registration may change when referencing.
     A member who has duplicated name cannot be stored.
     Maximum element of JSON array is 1,000.
  2. If <Bulk Insert flag> is not selected or none
     For JSON, starting with an array ([]) is not allowed.
  3. If <Bulk Insert flag> is single_resource_path

Parameters	Format	Description	Maximum length	M/O(*1)

_date	Timestamp (*2)	Registration timestamp - If not set, set the <registration timestamp> for the query string. If the <registration timestamp> for the query string also is not set, set the timestamp when data is received.	24 characters	O
_data	Character string	Resource data body to be registered (JSON format)	Follow the maximum overall size for the below	M

Each resource data shall have the following JSON object, and multiple values can be stored in a JSON array format.

(*1) M: Mandatory, O: Option
(*2) Conforms to ISO8601 standards (standard millisecond expressions). (20141225T103612.001Z, etc.) "Registration timestamps" all have the same specifications for millisecond-level precision (when omitting milliseconds, the system will read this as 0 milliseconds) and on.
*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

Example:

```
[
  {"_date":" 20160717T131520Z",
    " _data": {"key1":"value1"}},
  {"_date":" 20160717T131521Z",
   "_data":{"key1":"value1"}}
]
```

- API with bulk insert flag is more expensive than API without bulk insert. Please refer the service menu when contracted.
- Events are not processed for data registered with bulk insert.
- Data registered with bulk insert is not notified to the MQTT Subscriber.

- Body (When the extention is csv)
  1. The data of CSV form is converted into the JSON form and stores it. The data form after conversion is shown below.
     name : "csv"
     value : Two dementional array format
  2. Comforms to RFC4180 for CSV form.
     - Double quotation (") just before/after delimiter character (,) is deleted.
     - The line feed code of CSV form corresponds to CRLF and LF.
  3. Each values of CSV form is interpreted as follows:
     - Numerical value(the decimal point is included) or exponent notation, and $numconv=false
       It is interpreted as the numerical value. The sign between mantissa and exponent part is "e" or "E" at exponent notation.
     - 'true' or 'false'
       It is interpreted as the boolian value.
     - Case other than the above
       It is interpreted as string. When $charset=shift_jis is pointed out, this string is converted into UTF-8 as Shift_JIS character (regardless of an actual character-code).
  4. Byte-Order-Mark(BOM) that shows UTF-8 at the Body head is deleted.
  5. In the following case, the data converted into the JSON form is notified.
     - Transfeing data by Resource(Transfer)_JSON

- Additional data when event is generated.
- Transfering data to MQTT Subscriber.

6. The conversion example is shown as follows.

Body data(CSV form : before convertion)

```
node-a, true, 10.1
node-b, false, 20.0
```

stored data(JSON form : after conversion)

```
{
  "csv": [
    [
      "node-a",
      true,
      10.1
    ],
    [
      "node-b",
      false,
      20
    ]
  ]
}
```

- Body (When the extention is txt)

1. The data of Plain text form is converted into the JSON form and stores it. The data form after conversion is shown below.
   name : "txt"
   value : string
2. When $charset=shift_jis is pointed out, this string is converted into UTF-8 as Shift_JIS character (regardless of an actual character-code).
3. The conversion example is shown as follows.

Body data(Plain text form: before conversion)

```
#include <stdio.h>

int main(int argc, char **argv)
{
        printf("Hello, World!!\n";
}
```

Stored data(JSON form: after conversion)

```
{
  "txt":"#include <stdio.h>\r\n\r\nint main(int argc, char
**argv)\r\n{\r\n\tpirntf(\"Hello,world!!\\n\";\r\n)\r\n"
}
```

- Body (When the extension is bin)

1. The data of binary form is converted into the JSON from and sotres it. The data form after conversion is shown below.
   name : "bin"
   value : string
2. In the following case, the data converted into the JSON form is notified.
   - Transferring data by Resource(Transfer)_JSON
   - Additional data when event is generated.
   - Transferrring data to MQTT Subscriber.
3. The conversion example is shown as follows.

Body data(binary form: before conversion)

```
1234567890
```

Stored data(JSOM form: after conversion)

```
{
   "bin":"MTIzNDU2Nzg5Ma=="
}
```

- Response
  - When there is a normal response

Parameters	Value
Status- Code	200 OK
Headers	Refer to Section 2.3.2
Body	None

  - Refer to Section 2.3.2 for when there is an error response

3.2. Retrieving Resource_JSON Data

- Request

Parameters		Value
Method		GET
URI	Referencing the latest data	<Base URL>/v1/<Tenant ID>/<Resource path>/_present.<Extension>?<QUERY>
	Referencing past data(*1)	<Base URL>/v1/<Tenant ID>/<Resource path>/_past(<Registration timestamp>).<Extension>?<QUERY>
Headers		Refer to Section 2.3.1
Body		None

(*1) If multiple <registration timestamp> data exists, all data will be returned.

  - URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for referencing data	M
Registration timestamp	Target reference data registration timestamp(*2)	M
Extension	JSON only. When omitted (including .) will consider this as json.	O
QUERY	Reference conditions (mentioned below). When omitted (including ?), it is treated as no condition set (all).	O

(*1) M: Mandatory, O: Option

(*2) Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).

*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

1. In queries of URI parameter, request does not become error if queries are not described in above but are described in any of the other request in ths chapter (Registering, Searching, Updating, Deleting). Request with other queries become error.

  - QUERY list

Format	Description
$select=<Selection key>	Only returns data for fields specified with the <Selection key>. - The <Selection key> is equivalent to the name in JSON format, and the element name and attribute name in XML format. Any key in the

	registered data can be specified. The field level is expressed with a ".". - Multiple instances of the <Selection key> can be specified by separating each with a comma ",". - _date/_resource_path/_data, the management data for this service, cannot be used for the <Selection key>. [Example] $select=sensor.id,sensor.name,sensor.data.temp

- Response
  - When there is a normal response

Parameters		Value
Status- Code		200 OK
Headers	Content- Type	MIME type set according to the <extension>. - For json: application/json; charset=UTF- 8
Body		Matching resource data

  - Body

    Returns multiple registered data entries in the following format.

Name	Value
_resource_path	Resource path for the target data
_date	Target data registration timestamp(*1)
_data	Data registered in the JSON format.

(*1) Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).

*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

  - For error responses
    1. A "404 Not Found" message will be returned if the resource being referenced does not exist.
    2. A "204 No Content" message will be returned if the resource data matching the conditions set does not exist.
    3. A "400 Bad Request" is returned if two or more resource data entries match the conditions set, and the total data size exceeds 16 MBs, or the number of matching resource data items exceeds 1,000 items.
    4. Refer to Section 2.3.2 for information about other errors

3.3. Searching Resource_JSON Data

- Request

Parameters		Value
Method		GET
URI	Acquiring the data body(*1)	<Base URL>/v1/<Tenant ID>/<Resource path(/$all usable)>/_past.<Extension>?<QUERY>
	Acquiring the number of hits	<Base URL>/v1/<Tenant ID>/<Resource path(/$all usable)>/_past/_count?<QUERY>
Headers		Refer to Section 2.3.1
Body		None

(*1) The order of data returned as search results is sorted by <resource path> and <registration timestamp>.

  - URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path(/$all usable)	Resource path to be searched. The two description methods described below are used.	M
Extension	JSON only. Omitting this (including .) will consider this as json.	O
QUERY	Search conditions (described later). When omitted (including ?), it is treated as no condition set (all).	O

(*1) M: Mandatory, O: Option

1. In queries of URI parameter, request does not become error if queries are not described in above but are described in any of the other request in ths chapter (Registering, Retrieving, Updating, Deleting). Request with other queries become error.

✧ <Resource path(/$all usable)> description methods

1. Specify the full resource path:
   - Returns the resource data for the resource path specified.
2. Specify the resource path in part and add "/$all" at the end:
   - Returns the resource data for all resource paths under the path specified.
   - Example

     Assuming that three resources exist, "AX", "A/B", and "A/B/C", and "A/$all" is specified, resources "A/B" and "A/B/C" would be targeted.

Memo

To target all resource_JSON data within the target tenant, configure the following settings. But this is not recommended so request will be timed out by scanning too much data.

- <Tenant ID>/$all/_past

✧ When targeting multiple resource paths with $all, one of the following must be used for access codes stored to the Authorization header for the request.

1. Use an access code with read permissions for all target resource paths.
   - If there is even one resource path without a corresponding read permission this request will return an error response (401 Unauthorized).
2. Use an access code with hierarchy_get permissions for all upper target resource paths.
   - To specify $all, specify the resource path in part and add "$all" to the end when using hierarchy_get permissions.
   - Only read permissions will be used without hierarchy_get permissions if "$all" is specified by itself. If there is even one resource path without a corresponding read permission this request will return an error response (401 Unauthorized).

✧ QUERY list

Format	Description
$filter=<filter condition>	Results are limited to only those that match the <filter condition> set. A <filter condition> is set as the "property name operator condition value", capable of defining multiple filter conditions by "and" and "or". Usable operators and property names are described below.
$top=n	Limits the amount of data returned as search results to n items. The maximum number of acquirable items is 1,000.
$skip=n	Skips n items of data returned as search results. The order of data returned as search results is sorted by <resource path> for the first sort key and <registration timestamp> for the second sort key.

$select=<Selection key>	Returns data for fields specified with the <Selection key> only. - The <Selection key> is equivalent to the name in JSON format. Any name in the registered data can be specified. The field level is expressed with a ".". - Multiple instances of the <Selection key> can be specified by separating each with a comma ",". - _date/_resource_path/_data cannot be used for the <Selection key>. [Example] $select=sensor.no,sensor.name,sensor.data.temp
$orderby=<sort key> <sort order>	Rules for Sorting order of results. - <sort key> specifies name for sorting . Either of the following can be specified. _resource_path _date - <sort order> specifies sorting order. Either of the following can be specified. asc : ascending order desc : descending order - Requires a space between <sort key> and <sort order>. - Two or more pairs of <sort key> <sort order> can be specified by separating the pairs with a comma ",". - Sorting order of the <sort key> which is not specified is arbitrary. - When $orderby is not specified, it operates assuming that the following are specified. $orderby=_resource_path asc,_date desc

1. $top, $skip, $select and $orderby are ignored when /_count is specified. (Only $filter can be used in conjunction with this)

✧ Filter condition operators

Operator	Description	Example
eq	Equal sign	/temperature?$filter=Owner eq 'Tom'
ne	Inequality sign	/temperature?$filter=Owner ne null
gt	Greater than	/temperature?$filter=Floor1.Value gt 1000
ge	Equal or more	/temperature?&$filter=Floor1.Value ge 1000
lt	Less than	/temperature?$filter=Floor1.Value lt 1000
le	Equal or less	/temperature?$filter=Floor1.Value le 1000
and	Logical product	/temperature?$filter=Floor1.Value ge 1000 and Owner eq 'Tom'
or	Logical sum	/temperature?$filter=Id eq 2 or Id eq 1

1. null shows that the value does not exist.
2. Use <name>.<name> expressions when data names have a hierarchical structure.
3. Expressions like (A eq 1 and B eq 1)or(A eq 2 and B eq 2) are also allowed.
   - However, in these instances () cannot be defined when inside (). For example, ((A eq 1 and B eq 1)or(A eq 2 and B eq 2))and(C eq 1) would return an error.
4. Matching by regular expressions is not supported.
5. Surround character strings in single quotations. Values not surrounded in single quotations are interpreted as numerical values.
6. When specifying the content of the JSON array, it is expressed with <name>.<array index>. Array index is numeric value.
   Example:

data for searching	{ "Owners":["Taro",

	"Jiro"] }
Ex. Of filter condition	$filter=Owners.0 eq 'Taro'

- When there is an object structure in the array and name is composed only of the numeric character, array Index and name might not be able to be identified. In this case, specify both array index and name referring to the example of the following.

example data for searching:

(1) { "data": [{ "0":"Taro" }, { "0":"Jiro" }] } (2) { "data": [{ "0":"Jiro"}, { "0":"Taro" }] } (3) { "data": { "0":"Taro" } }

example result of searching

filter condition	Matching datas
$filter=data.0 eq 'Taro'	(1), (2), (3)
$filter=data.1.0 eq 'Taro'	(2)

✧ Property names usable for filter conditions

Property name	Description	Notes
_date	Registration timestamp	Search target data registration timestamp(*1) *Do not enclose registration timestamps used to specify conditions with _date in single quotations.
name used	name used included in registration data	Do not encode characters other than URI non-reserved characters ("Single byte alphanumeric characters", "- ", ".", "_", "~") using percentage marks.

(*1) Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).

*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

- Response

[Acquiring data body (resource data)]

- When there is a normal response

Parameters		Value
Status- Code		200 OK
Headers	Content- Type	application/json; charset=UTF- 8
Body		Matching resource data

1. Body

Returns multiple registered data entries in the following format. The format is based on the <extension>.

Key	Value
resource_path	Resource path for the target data
_date	Target data registration timestamp(*1)
_data	Data registered in a format according to the <extension>.

(*1) Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).

*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

- If data matching search conditions for the target resource does not exist, data related to the resource will not be returned.

- For error responses
  1. A "404 Not Found" message will be returned if the target resource does not exist.
  2. A "204 No Content" message will be returned if the resource data matching the conditions set does not exist.
  3. A "400 Bad Request" is returned if two or more resource data entries match the conditions set, and the total data size exceeds 16 MBs, or the number of matching resource data items exceeds 1,000 items.
     - In this case, acceptable_top=n is stored to the Body as the top number of data items that can be received for the error response.
  4. Refer to Section 2.3.2 for information about other errors

[Acquiring the number of hits]

- When there is a normal response

Parameters		Value
Status- Code		200 OK
Headers	Content- Type	text/plain
Body		Amount of matching resource data

1. Body
   - Returns the amount of matching resource data as is.

- Refer to Section 2.3.2 for when there is an error response

3.4. Updating Resource_JSON Data

- Request

Parameters		Value
Method		PUT
URI	Updating past data(*1)	<Base URL>/v1/<Tenant ID>/<Resource path>/_past(<Registration timestamp (Old)>).<Extension>.<compression type>?$newdate=<Registration timestamp (New)> &$charset=<Character set of Body data>&$skip=<deleting lines of Body data>&$numconv=<Numerical conversion>
Headers		Refer to Section 2.3.1
Body		Data for updating

(*1) When multiple data with the same registration date exists, only one (randomly chosen) is updated.

- URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for updating data	M
Extension	Specifies data format from either of following. When omitted (including .) will consider this as json. - json - csv - txt	O

	- bin	
Compression type	Specifies compression type from following when transmitting data is compressed. When omitted (including .) will consider that data is not compressed. - gz	O
Registration timestamp (old)	Target update data registration timestamp(*2)	M
Registration timestamp (new)	Registration timestamp(*2) after update - This can be omitted (including $newdate=). When omitted, this service will not update the registration timestamp. - This will be overwritten without checking whether there is data with a pre- specified registration timestamp.	O
Character set of Body data	Specifies character set of Body data from either of following. It is effective only when the extension is either csv or txt. - utf-8 : UTF-8 - shift_jis : Shift-JIS When omitted (including $charset=) will consider this as utf-8.	O
Deleting lines of Body data	Specifies number of lines to delete from top of Body data. It is effective only when the extention is csv. When omitted (including $skip=), no lines are deleted.	O
Numerical conversion	Specifies whether or not to convert numerical figure into numerical value. It is effective when the extension is only csv. - true : numerical figure will be converted into numerical value. - false : numerical figure will be considered as string. Can be omitted when it is true($numconv=including).	O

(*1) M: Mandatory, O: Option
(*2) Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).
*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

1. In queries of URI parameter, request does not become error if queries are not described in above but are described in any of the other request in ths chapter (Registering, Retrieving, Searching, Deleting). Request with other queries become error.

✧ HTTP header

Headers	Value	Additional notes	M/O(*1)
Content-Type	- When the extention is json or omitted application/json; charset=UTF-8	Fixed	O
	- When the extention is csv text/csv; charset=<character set> <character set> is either of following - UTF-8 - Shift_JIS		
	- When the extention is txt text/plain; charset=<character set> <character set> is either of following - UTF-8 - Shift_JIS		
	- When the extention is bin MIME type of Body		

	- When the compression type is gz (the extention is anonymous) application/gzip or application/x-gzip		
x-iotpf-request-id	Request identification (any string)(*2)	It is used for error information output when the request is lost by error in this system or forwarding system. Please Refer to section 11.3 for error information.	O

(*1) M: Mandatory, O: Option

(*2) It operates assuming that x-iotpf-reqeust-id is not specified when only "-" is set to the value

- ✧ Body

Data to be registered in a format according to the extension. Refer to section 3.1 for details.

- ➢ For JSON, starting with an array ([]) is not allowed.

- Response
  - ✧ When there is a normal response

Parameters	Value
Status- Code	200 OK
Headers	Refer to 2.3.2
Body	None

- ✧ Refer to Section 2.3.2 for when there is an error response

3.5. Deleting Resource_JSON Data

- Request

Parameters		Value
Method		DELETE
URL	Deleting past data	<Base URL>/v1/<Tenant ID>/<Resource path>/_past?<QUERY>
Headers		Refer to Section 2.3.1, and following
Body		None

- ✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for deleting data	M
QUERY	Deletion conditions (described later).	M

(*1) M: Mandatory, O: Option

- ✧ HTTP header

Headers	Value	Additional notes	M/O(*1)
x-iotpf-request-id	Request identification (any string)(*2)	It is used for error information output when the request is	O

		lost by error in this system or forwarding system. Please Refer to section 11.3 for error information.	

(*1) M: Mandatory, O: Option

(*2) It operates assuming that x-iotpf-reqeust-id is not specified when only "-" is set to the value

- ✧ QUERY list

Format	Description
$filter=<filter condition>	Results are limited to deleting only those that match the <filter condition> set. A <filter condition> is set as the "property name operator condition value", capable of defining multiple filter conditions by "and" and "or". Usable operators and property names are described below.

1. In queries of URI parameter, request does not become error if queries are not described in above but are described in any of the other request in ths chapter (Registering, Retrieving, Searching, Updating). Request with other queries become error.

- ✧ Filter condition operators

Refer to Section 3.3

- ✧ Property names usable for filter conditions

Refer to Section 3.3

- Response
  - ✧ When there is a normal response

Parameters	Value
Status- Code	200 OK
Headers	Refer to Section 2.3.1
Body	None

- ✧ Refer to Section 2.3.2 for when there is an error response

Chapter 4 Controlling Resource_JSON/Resource (Transfer)_JSON Data (MQTT)

Below is a table of the MQTT interface used to control Resource_JSON and Resource (Transfer)_JSON data.

Table 7: Data storing/reference interface list

Protocol	Type	Purpose	Chapter
MQTT	Registering data to a resource	Adds and registers (transfers) new data to a resource. [Option] - None: Registers data based on the timestamp when data is received by this service. - Specify time/date: Registers data based on a set timestamp.	Chapter 4.1
	Referencing resource data	References data registered to a resource	Chapter 4.2

4.1. Registering Data to Resource_JSON/Transferring Data with Resource (Transfer)_JSON

- Request

Parameters		Value
Message type		PUBLISH
Topic	Data storing	<Access code>/v1/<Tenant ID>/<Resource path>
	Transferring data	<Access code>/v1/<Tenant ID>/_fwd/<Resource path>
Other fixed headers and variable headers		Refer to 2.5.1
Payload		For headers for this service, and data to be registered The topmost "- - - IoT- PF<CRLF>" value in a payload onwards are headers for this service. Write the header name and value in "<Header name>: <Value><CRLF>" format. "<CRLF><CRLF>" tags are used to determine the end of the header for this service. [<CRLF>] includes the Header portion. The headers for this service can be omitted.

✧ Headers for this service

Headers	Value	M/O(*1)
Date	<Registration timestamp>	O
x-iotpf-request-id	<Request identification(any stirng)>(*2)	O

(*1)M: Mandatory, O: Option

(*2) It is used for error information output when the request is lost by error in this system or forwarding system. Please Refer to section 11.3 for error information.

✧ Parameters

Parameter name	Description	M/O(*1)
Access code	Character string representing the access permission. Acquire from administrator.	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for registering data	M
Registration timestamp	Registration timestamp(*2) given to registration data. - This can be omitted. When omitted, this service will use the request received timestamp.	O

	- This will be overwritten without checking whether there is data with a pre- specified registration timestamp.	
Request identification	It is used for error information output when the request is lost by error in this system or forwarding system. Please Refer to section 11.3 for error information. It operates assuming that x-iotpf-reqeust-id is not specified when only "-" is set to the value.	O

(*1)M: Mandatory, O: Option
(*2) Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).
*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

- ✧ Data targeted for registration
  1. User- selected data in the JSON format.
     - ➢ Starting with an array ([]) is not allowed.
     - ➢ Member order not guaranteed. The order at registration may change when referencing.
     - ➢ A member who has duplicated name cannot be stored.

- Response
  - ✧ Messages based on QoS level(None/PUBACK Message/PUBREC Message)

4.2. Referencing Resource_JSON/Resource (Transfer)_JSON Data

SUBSCRIBE messages are used to express an interest in resource data with the MQTT protocol.
SUBSCRIBE to a relevant Topic (resource path) to receive messages when data is registered via REST or MQTT.

Protocol	Operation	Receive with SUBSCRIBE	Retain option
REST	Registering resource data	Yes(*1)	Yes
	Referencing resource data	No	-
	Updating resource data	No	-
	Deleting resource data	No	-
MQTT	Registering resource data	Yes	Yes
	Referencing resource data	No	-

(*1) Data registered with bulk insert cannot be received to the MQTT PUBLISH.

- Request

Parameters	Value
Message type	SUBSCRIBE
Fixed headers and variable headers	Refer to Section 2.5.1
Payload	Topic name "<Access code>/v1/<Tenant ID>/<Resource path>" or "<Access code>/v1/<Tenant ID>/ fwd/<Resource path>" list

- ✧ Parameters

Parameter name	Description	M/O(*1)

Access code	Character string representing the access permission. Acquire from administrator.	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for registered data	M

(*1) M: Mandatory, O: Option

- Response
  - ✧ SUBACK messages

Chapter 5 Controlling Resource_Binary Data (REST)

A list of interfaces for storing data to Resource_Binary/referencing resource data is provided below.

Table 8: Data storing/reference interface list

Protocol	Type	Purpose	Chapter
REST	Registering data to a resource	Adds and registers new data to a resource. [Option] - None: Registers data based on the timestamp when data is received by this service. - Specify time/date: Registers data based on a set timestamp.	Chapter 5.1
	Referencing resource data	References data registered to a resource. [Option] - None: Returns the latest data - Specify time/date: Returns data with the timestamp specified	Chapter 5.2
	Retrieving resource data	Set search conditions and retrieve data registered to a resource. Information for registered data that matches the search conditions is returned as search results. (Only metadata can be searched, not the actual registered data.) [Option] - You can specify search conditions with QUERY.	Chapter 5.3
	Deleting resource data	Deletes data registered to a resource [Option] - You can specify deletion conditions with QUERY	Chapter 5.5

Note, binary data registered in Resouce_Binary by REST can not be referenced by MQTT SUBSCRIBE.

5.1. Registering Data to Resource_Binary

- Request

Parameters		Value
Method		PUT
URI	Data storing	<Base URL>/v1/<Tenant ID>/_bin/<Resource path>?$date=<registration timestamp>
Headers		Refer to Section 2.3.1 and below
Body		Data for registering

> Memo
>
> From the idea of "storing data to the existing resource", PUT Method is used as the specification of data registration in IoT Platform . POST Method to the resource is used for registering resources(7.1 Register resource).

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for registering data	M
Registration timestamp	Registration timestamp(*2) given to registration data. Can be omitted (include $date=). When omitted, this service will use	O

	the request received timestamp. - This will be overwritten and registered when there is data with a pre- specified registration timestamp.	

(*1) M: Mandatory, O: Option
(*2) Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).
*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

✧ HTTP header

Headers	Value	Additional notes	M/O(*1)
Content- Type	Body MIME type	Managed as metadata for resource data, and attached to responses	O
x-iotpf-meta-data1 x-iotpf-meta-data2 x-iotpf-meta-data3	User- defined metadata A user- defined value can be set as an item name for name.	Managed as metadata for resource data, and attached to responses	O
x-iotpf-request-id	Request identification (any string)(*2)	It is used for error information output when the request is lost by error in this system or forwarding system. Please Refer to section 11.3 for error information.	O

(*1) M: Mandatory, O: Option
(*2) It operates assuming that x-iotpf-reqeust-id is not specified when only "-" is set to the value

✧ Body

1. Data targeted for registration.
   - ➢ Maximum size: 100MB
   - ➢ Binary data registered in REST cannot be referenced by MQTT

- Response
  - ✧ When there is a normal response

Parameters	Value
Status- Code	200 OK
Headers	Refer to Section 2.3.2
Body	None

✧ Refer to Section 2.3.2 for when there is an error response

5.2. Referencing Resource_Binary Data

- Request

Parameters		Value
Method		GET
URI	Referencing the latest data	<Base URL>/v1/<Tenant ID>/_bin/<Resource path>/_present
	Referencing past data	<Base URL>/v1/<Tenant ID>/_bin/<Resource path>/_past(<registration timestamp>)
Headers		Refer to 2.3.1
Body		None

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for referencing data	M
Registration timestamp	Target reference data registration timestamp(*2).	M

(*1) M: Mandatory, O: Option
(*2) Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).
*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

- Response
  ✧ When there is a normal response

Parameters		Value	Description	
Status- Code		200 OK	OK (All)	
		206 Partial content	OK (Partially)	
Headers	field name	Value		M/O(*1)
	Content- Type	Body MIME type. Set the value specified when registering resource data Set “application/octet- stream” if not specified when registering resource data.		M
	x-iotpf-meta-data1 x-iotpf-meta-data2 x-iotpf-meta-data3	Metadata. Assigned if set when registering resource data.		O
Body		Data body		

(*1) M: Mandatory, O: Option

✧ For error responses

1. A "404 Not Found" message will be returned if the resource data being referenced does not exist.
2. A “204 No Content” message will be returned if the resource data being matched.
3. Refer to Section 2.3.2 for information about other errors

5.3. Retrieving Resource_Binary Data

- Request

Parameters		Value
Method		GET
URI	Acquiring data lists (*1)	<Base URL>/v1/<Tenant ID>/_bin/<Resource path(/$all usable)>/_past?<QUERY>
	Acquiring the number of hits	<Base URL>/v1/<Tenant ID>/_bin/<Resource path(/$all usable)>/_past/_count?<QUERY>
Headers		Refer to 2.3.1
Body		None

(*1) The order of data returned as search results is sorted by <registration timestamp>.

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M

Resource path	Resource path to be searched. The two description methods described below are used.	M
QUERY	Search conditions (described later). When omitted (include ?), it is treated as no condition set (all).	O

(*1) M: Mandatory, O: Option

✧ <Resource path(/$all usable)> description methods

1. 1) Specify the full resource path:
   - Returns the resource data information for the resource path specified.
2. 2) Specify the resource path in part and add "/$all" to the end:
   - Returns the resource data information for all resource paths under the path specified.
   - Example
   - Assuming that three resources exist, "AX", "A/B", and "A/B/C", and "A/$all" is specified, resources "A/B" and "A/B/C" would be targeted.
   - To target all Binary resources within the target tenant, configure the following settings.
   - <Tenant ID>/_bin/$all/_past

✧ When targeting multiple resource paths with $all, one of the following must be used for access codes stored to the Authorization header for the request.

1) Use an access code with read permissions for all target resource paths.
   - If there is even one resource path without a corresponding read permission this request will return an error response (401 Unauthorized).

2) Use an access code with hierarchy_get permissions for all upper target resource paths.
   - To specify $all, specify the resource path in part and add "$all" to the end when using hierarchy_get permissions.
   - Only read permissions will be used without hierarchy_get permissions if "$all" is specified by itself. If there is even one resource path without a corresponding read permission this request will return an error response (401 Unauthorized).

✧ QUERY list

Format	Description
$filter=<filter condition>	Results are limited to only those that match the <filter condition> set. A <filter condition> is set as the "property name operator condition value", capable of defining multiple filter conditions by "and". Usable operators and property names are described below.
$top=n	Limits the amount of data returned as search results to n items. The maximum number of acquirable items is 1,000.
$skip=n	Skips n items of data returned as search results. The order of data returned as search results is sorted by <resource path> for the first sort key and <registration timestamp> for the second sort key.

1. $top and $skip are ignored when /_count is specified (only $filter can be used in conjunction with this)

✧ Filter condition operators

Operator	Description	Example
eq	Equal sign	/image?$filter=_date eq 20160101T000000.001Z
gt	Greater than	/image?$filter=_date gt 20160101T000000.001Z
ge	Equal or more	/image?$filter=_date ge 20160101T000000.001Z

lt	Less than	/image?$filter=_date lt 20160101T000000.001Z
le	Equal or less	/image?$filter=_date le 20160101T000000.001Z
and	Logical product	/image?$filter=_date ge 20160101T000000.001Z and _date le 20160529T101010.001Z

1. “and” can only be used once, as shown in the example above.

✧ Property names usable for filter conditions

Property name	Description	Notes
_date	Registration timestamp	Search target data registration timestamp(*1) *Do not enclose registration timestamps used to specify conditions with _date in single quotations.

(*1) Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).

*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

- Response

[Acquiring data lists]

✧ When there is a normal response

Parameters		Value
Status- Code		200 OK
Headers	Content- Type	application/json; charset=UTF- 8
Body		Matching metadata for resource data

1. Body
   - Returns multiple registered data information items in JSON format.

Name	Value	M/O(*1)
name	Name of the resource data Expressed with the URI path. _bin/<Resource path>/_past(<registration timestamp>)	M
content_type	Resource data MIME type. Set the Content- Type specified when registering resource data. Set “application/octet- stream” if not specified when registering resource data.	M

(*1) M: Mandatory, O: Option

   - If data matching search conditions for the target resource does not exist, data related to the resource will not be returned.

✧ For error responses

1. A "404 Not Found" message will be returned if the target resource does not exist.
2. A "204 No Content" message will be returned if the resource data matching the conditions set does not exist.
3. A 400 Bad Request is returned if the number of resource data items matching the conditions set exceeds 1,000 items.
   - In this case, acceptable_top=n is stored to the Body as the top number of data items that can be received for the error response.
4. Refer to Section 2.3.2 for information about other errors.

[Acquiring the number of hits]

✧ When there is a normal response

Parameters		Value
Status- Code		200 OK
Headers	Content- Type	text/plain; charset=UTF-8
Body		Amount of matching resource data

1. Body
   - ➢ Returns the amount of matching resource data as is.

✧ Refer to Section 2.3.2 for when there is an error response

5.4. Updating Resource_Binary Data

There is no interface for updating Resource_Binary. However, data in Resouce_Binary can be overwritten by registering data as described in section 5.1, using the same registration timestamp.

5.5. Deleting Resource_Binary Data

- Request

Parameters		Value
Method		DELETE
URI	Deleting past data	<Base URL>/v1/<Tenant ID>/_bin/<Resource path>/_past?<QUERY>
Headers		Refer to 2.3.1
Body		None

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for deleting data	M
QUERY	Deletion conditions (described later).	M

(*1) M: Mandatory, O: Option

✧ HTTP header

Headers	Value	Additional notes	M/O(*1)
x-iotpf-request-id	Request identification (any string)(*2)	It is used for error information output when the request is lost by error in this system or forwarding system. Please Refer to section 11.3 for error information.	O

(*1) M: Mandatory, O: Option

(*2) It operates assuming that x-iotpf-reqeust-id is not specified when only "-" is set to the value

✧ QUERY list

Format	Description
$filter=<filter condition>	Results are limited to deleting only those that match the <filter condition> set. A <filter condition> is set as the "property name operator condition value", capable of defining multiple filter conditions by "and" . Usable operators and property names are described below.

- ✧ Filter condition operators

  Refer to Section 3.3

- ✧ Property names usable for filter conditions

  Refer to Section 3.3

- Response
  - ✧ When there is a normal response

Parameters	Value
Status- Code	200 OK
Headers	Refer to 2.3.2
Body	None

- ✧ Refer to Section 2.3.2 for when there is an error response

Chapter 6 Controlling Resource_Binary Data (MQTT)

A list of interfaces for storing data to Resource_Binary/referencing resource data is provided below.

Table 9: Data storing/reference interface list

Protocol	Type	Purpose	Chapter
MQTT	Registering data to a resource	Adds and registers new data to a resource.	Chapter 6.1
	Referencing resource data	References data registered to a resource.	Chapter 6.2

6.1. Registering Data to Resource_Binary

- Request

Parameters	Value
Message type	PUBLISH
Topic	<Access code>/v1/<Tenant ID>/_bin/<Resource path>
Other fixed headers and variable headers	Refer to 2.5.1
Payload	Data to be registered

✧ Parameters

Parameter name	Description	M/O(*1)
Access code	Character string representing the access permission. Acquire from administrator.	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for registering data	M

(*1) M: Mandatory, O: Option

✧ Data targeted for registration

1. Maximum size: 16MB

- Response
  - ✧ Messages based on QoS level (None/PUBACK Message/PUBREC Message)

6.2. Referencing Resource_Binary Data

SUBSCRIBE messages are used when referencing resource data with the MQTT protocol.
SUBSCRIBE to a relevant Topic (resource path) to receive the following messages with MQTT PUBLISH.
*Binary data registered with REST IF cannot be referenced with MQTT.

Protocol	Operation	Receive with SUBSCRIBE	Retain option
REST	Registering resource data	No	-
	Referencing resource data	No	-
	Updating resource data	No	-

	Deleting resource data	No	-
MQTT	Registering resource data	Yes	Yes
	Referencing resource data	No	-

- Request

Parameters	Value
Message type	SUBSCRIBE
Fixed headers and variable headers	Refer to 2.5.1
Payload	Topic name "<Access code>/v1/<Tenant ID>/_bin/<Resource path>" list

  - ✧ Parameters

Parameter name	Description	M/O(*1)
Access code	Character string representing the access permission. Acquire from administrator.	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for registered data	M

(*1) M: Mandatory, O: Option

- Response
  - ✧ SUBACK messages

Chapter 7 Controlling Resources (REST)

A table showing the interface used to control resources is provided below.
For using resource control API, it is necessary to set the resource and the access code beforehand from Service Portal. Please refer to Chapter 2.2

Table 10: Resource Control Interface List

Type	Purpose	Chapter
Register resource	Registers a new resource. [Option] - No Body: Creates the resource only - Store metadata in Body: Configure default metadata settings when registering resources.	Chapter 7.1
Reference resource metadata	Reference resource metadata. [Option] - Specifying part of the resource path with $all: Returns metadata for resources under the specified path.	Chapter 7.2
Update resource metadata	Updates resource metadata. [Option] None	Chapter 7.3
Delete resource	Deletes the resource. [Option] None	Chapter 7.4

7.1. Register resource

- Request

Parameters			Value
Method			POST
URI	JSON	Storing	<Base URL>/v1/<Tenant ID>/<Resource path>
		Transfer	<Base URL>/v1/<Tenant ID>/_fwd/<Resource path>
	Binary		<Base URL>/v1/<Tenant ID>/_bin/<Resource path>
Headers			Refer to 2.3.1 and following
Body			Metadata

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for registering data	M

(*1) M: Mandatory, O: Option

✧ HTTP headers

Headers	Value	Additional notes	M/O(*1)
Content-Type	application/json; charset=UTF-8	Fixed	O

(*1) M: Mandatory, O: Option

✧ Body

You can reset metadata by storing metadata in JSON format, as shown below.

Parameters	Format	M/O (*1)	Description	Maximum length
resource	-	M	Starting tag	-
retention_period	Numerical values	O(*2)	Retention period for the resource data (days) - The retention period shall be one day when this is not set. - Specify a value from 1 to 9999.	9999
fwd_info	-	O(*3)	Transfer destination information	-
http	-	M	HTTP transfer destination information	-
method	String	M	Either "GET", "POST", "PUT", "DELETE", "HEAD", "OPTIONS", or "TRACE"	7 characters
uri	String	M	URI. "http://" or "https://"	256 characters
basic_auth_id	String	O	Basic authentication ID	20 characters
basic_auth_pass	String	O	Basic authentication password	20 characters
header_fields	- (array)	O	HTTP header (Maximum of 10 elements in an array)	-
field_name	String	M	Header field name. ":" not included.	20 characters
field_value	String	M	Value stored to the header field	512 characters

(*1) M: Mandatory, O: Option

The M/O mark for each child element represents whether it is necessary to configure the element when the parent element has been configured or not.

(*2) Valid only for Resource_JSON and Resource_Binary.

(*3) This can only be set when the target resource is a JSON-format transfer resource.

- Response
  - ✧ When there is a normal response

Parameters		Value
Status- Code		201 Created
Headers	Location	<Base URL>/v1/<Tenant ID>/<Resource path> or <Base URL>/v1/<Tenant ID>/_fwd/<Resource path> or <Base URL>/v1/<Tenant ID>/_bin/<Resource path>
Body		None

✧ Refer to Section 2.3.2 for when there is an error response

7.2. Referencing Resource Metadata

- Request

Parameters	Value
Method	GET

URI	Acquiring data lists	JSON	Storing	<Base URL>/v1/<Tenant ID>/<Resource path(/$all) >/_resources?<QUERY>
			Transfer	<Base URL>/v1/<Tenant ID>/_fwd/<Resource path(/$all) >/_resources?<QUERY>
		Binary		<Base URL>/v1/<Tenant ID>/_bin/<Resource path(/$all) >/_resources?<QUERY>
	Acquiring the number of hits	JSON	Storing	<Base URL>/v1/<Tenant ID>/<Resource path(/$all) >/_resources/_count?<QUERY>
			Transfer	<Base URL>/v1/<Tenant ID>/_fwd/<Resource path(/$all)>/_resources/_count?<QUERY>
		Binary		<Base URL>/v1/<Tenant ID>/_bin/<Resource path(/$all)>/_resources/_count?<QUERY>
Headers				Refer to 2.3.1
Body				None

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
QUERY	Reference conditions (mentioned below). When omitted (including ?), it is treated as no condition set (all).	O

(*1) M: Mandatory, O: Option

✧ <Resource path(/$all)> description methods

1) Specify the full resource path:
   - Returns the metadata for the resource path specified.

2) Specify the resource path in part and add "/$all" to the end:
   - Returns the metadata for all resource paths under the path specified.
   - Example
     - Assuming that four resources exist, "A", "AX", "A/B", and "A/B/C", and "A/$all" is specified, resources "A/B" and "A/B/C" would be targeted.
   - To target all resource paths within each resource type, configure the following settings.
     - Resource_JSON: <Tenant ID>/$all/_resources
     - Resource(transfer)_JSON: <Tenant ID>/_fwd/$all/_resources
     - Resource_Binary: <Tenant ID>/_bin/$all/_resources

     *Resource paths for different types of resources cannot be referenced at the same time.

✧ QUERY list

Format	Description
$top=n	Limits the amount of data returned as search results to n items (maximum of 1,000 items).
$skip=n	Skips n items of data returned as search results. The order of data returned as search results is sorted by resource path.

1. $top and $skip are ignored when /_count is specified

- Response

[Acquiring data lists]

✧ When there is a normal response

Parameters		Value
Status- Code		200 OK
Headers	Content- Type	application/json; charset=UTF- 8
Body		Target resource metadata (multiple)

1. Body
   - The metadata for each resource are JSON objects, and multiple metadata items can be stored in a JSON array format as shown below.
   - Information for each resource is sorted by resource path (in ascending order)

Parameters	Format	M/O (*1)	Description	Maximum length
Resources	-	M	Starting tag	-
resource_path	string	M	Resource path	128 characters
retention_period	Numerical values	O	Retention period for the resource data - The retention period shall be one day when this is not set. - Set values from 1 to 9999.	9999
last_modified(*2)	Timestamp	M	Latest data <registration timestamp>(*3)	20 characters
fwd_info	-	O(*4)	Transfer destination information	-
http	-	M	HTTP transfer destination information	-
Method	string	M	Either “GET”, ”POST”, ”PUT”, ”DELETE”, ”HEAD”, ”OPTIONS”, or ”TRACE”.	6 characters
Uri	string	M	URI. “http://” or “https://”	256 characters
basic_auth_id	string	O	Basic authentication ID	20 characters
basic_auth_pass	string	O	Basic authentication password	20 characters
header_fields	- (array)	O	HTTP header (A maximum of up to 10 elements can be set in an array)	-
field_name	string	M	Header field name. “:” is not included.	20 characters
field_value	string	M	Value stored to the above-mentioned header field	512 characters

(*1) M: Mandatory, O: Option
The M/O mark for each child element represents whether it is necessary to configure the element when the parent element has been configured or not.
(*2) Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).
*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.
(*3) The latest update timestamp on the server and the latest data <registration timestamp> may not match.
(*4) This is only set when the target resource is a JSON-format transfer resource.

- Parameters are not stored in a response when no value is set for the parameter.

- The last_modified parameter is also not stored when no data is found within a resource.
- The access code specified for the Authorization header of a request cannot be referenced when list permissions have not been granted for the target resource.

✧ For error responses

1. A "404 Not Found" message will be returned if no resources matching the conditions set are found.
2. A 400 Bad Request is returned if the number of matching resource data items exceeds 1,000 items. In this case, acceptable_top=n is stored to the Body as the top number of data items that can be received for the error response.
3. Refer to Section 2.3.2 for information about other errors

[Acquiring the number of hits]

✧ When there is a normal response

Parameters		Value
Status- Code		200 OK
Headers	Content- Type	text/plain
Body		Amount of matching resources

1. Body
   - Returns the amount of matching resources as is.
   - Resources for which list permissions have not been granted for the access code stored to the Authorization header of the request are not counted.

✧ Refer to Section2.3.2 for when there is an error response

7.3. Updating Resource Metadata

- Request

Parameters			Value
Method			PUT
URI	JSON	Storing	<Base URL>/v1/<Tenant ID>/<Resource path>/_resources
		Transfer	<Base URL>/v1/<Tenant ID>/_fwd/<Resource path>/_resources
	Binary		<Base URL>/v1/<Tenant ID>/_bin/<Resource path>/_resources
Headers			Refer to 2.3.1
Body			Metadata

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for updating metadata	M

(*1) M: Mandatory, O: Option

✧ HTTP headers

Headers	Value	Additional notes	M/O(*1)
Content-Type	application/json; charset=UTF-8	Fixed	O

(*1) M: Mandatory, O: Option

- ✧ Body

1. Refer to Section7.1.
2. Overwrites all metadata for the target resource with the updated data.
   - ➢ If resource A and B are registered before the update, and if only resource A is specified when updating, the system will interpret this as a request to delete resource B and resource B will be deleted.

- Response
  - ✧ When there is a normal response

Parameters	Value
Status- Code	200 OK
Headers	Refer to 2.3.2
Body	None

- ✧ Refer to Section 2.3.2 for when there is an error response

7.4. Delete resource

- Request

Parameters			Value
Method			DELETE
URI	JSON	Storing	<Base URL>/v1/<Tenant ID>/<Resource path>
		Transfer	<Base URL>/v1/<Tenant ID>/_fwd/<Resource path>
	Binary		<Base URL>/v1/<Tenant ID>/_bin/<Resource path>
Headers			Refer to Section 2.3.1
Body			None

- ✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the resource owning tenant	M
Resource path	Resource path for deletion	M

(*1) M: Mandatory, O: Option

- Response
  - ✧ When there is a normal response

Parameters	Value
Status- Code	204 No Content
Headers	Refer to Section 2.3.2
Body	None

- ✧ For error responses

1. Resources targeted for deletion cannot be deleted when an access code is associated with the resource. A "423 Locked" message will be returned in this instance.
2. Refer to Section 2.3.2 for information about other errors

Chapter 8 Controlling Access Codes (REST)

The interface for controlling access codes is described below. REST(HTTP) is always used as the protocol.

Table 11: Access Code Control Interface List

Type	Purpose	Chapter
Register access code	Registers a new access code. [Option] None	Chapter 8.1
Reference access code information	References (retrieves) access code information. [Search option] - Specify access code - Specify resource path (allows for prefix match searches)	Chapter 8.2
Update access code information	Updates access code information. [Option] None	Chapter 8.3
Delete access code	Deletes the access code. [Option] None	Chapter 8.4

8.1. Registering Access Codes

- Request

Parameters	Value
Method	POST
URI	<Base URL>/v1/<Tenant ID>/_access_codes/<Access Code>
Headers	Refer to 2.3.1 and below
Body	Access code information

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the target tenant	M
Access Code	Access code	M

(*1) M: Mandatory, O: Option

✧ HTTP header

Headers	Value	Additional notes	M/O(*1)
Content- Type	application/json; charset=UTF- 8	Fixed	M

(*1) M: Mandatory, O: Option

✧ Body: Access code information

Registers access code information in JSON format as shown below.

Parameters	Format	M/O(*1)	Description	Maximum length
access_code	-	M	Starting tag	-
Permissions	-	M	Permissions information	-

	ip_filter	string (array)	O	Specify IP address range information that was set by access control (access code) defined in the Service Portal. An error response (400 Bad Request) will be returned when undefined IP address range information is set. - You can specify two IP addresses in the following format: ["Starting IP address", "Ending IP address"]. - When specifying one IP address, specify the same address as the starting IP address and the ending IP address. - A greater value cannot be set for the starting IP address over the ending IP address (when comparing IP addresses in 32 bit format).	35 characters
	resource_operations	- (array)	M	Permission information tied to a resource	
	resource_path	string	M	Resource path	128 characters
	operations	string (array)	M	Either of "hierarchy_get","hierarchy_put", "create", "read", "update", "delete", or "list". Multiple values can be specified in an array.	72 characters
certification_info		—	O	Client certificate information	
	certification	string	M	Client certificate ・PEM format Include "-----BEGIN CERTIFICATE-----", "-----END CERTIFICATE-----". Register line break code as "\n".	10000 characters
	certificate_usage	string	M	Client certificate application Please specify the following. ・auth: Use for client authorization.	4 characters
protocols		string (array)	O	Information to specify protocols: "http" , "https", "mqtt" or tp" , " Any combinations can be specified by array.	29 characters

(*1) M: Mandatory, O: Option
The M/O mark for each child element represents whether it is necessary to configure the element when the parent element has been configured or not.

1) Attempting to register access code information without permissions assigned will return an error.

- That means one or more resource_path and operations must be set for resource_operations. Maximum 1000 combinatins can be set.
- Configurable permission setting patterns are outlined in the table below.

	Permissions				
Pattern	create	read, hierarchy_get(*1)	update, hierarchy_put(*1)	delete	list
1	Yes	Yes	Yes	Yes	Yes
2	Yes	Yes		Yes	Yes
3	Yes		Yes	Yes	Yes
4	Yes			Yes	Yes
5		Yes	Yes		Yes
6		Yes			Yes
7			Yes		Yes
8					Yes
9		Yes	Yes		
10		Yes			
11			Yes		

(*1) Either one, or both of the permissions can be set

2) Create/delete/list/hierarchy_get/hierarchy_put permissions apply to all resource paths under the specified resource path.
3) If create/delete/list/hierarchy_get/hierarchy_put permissions are set to a different level path, permissions set for the higher path shall take precedence.
   - Example:
     If create, delete, list, hierarchy_get, and hierarchy_put permissions are set for resource path "A/B",
     and if only list permissions are set for resource path "A/B/C",
     create, delete, list, hierarchy_get, and hierarchy_put permissions shall apply for resource path "A/B/C/D".
4) If you set certification_info, both certification and certificate_usage must be set.If not, an error response (400 Bad Request) will be returned
5) Setting protocols is explained as below.

Pattern	Behavior
No member of "protocols"	Will judge all protocols have been specified.
No value while a member of " protocol" exists. ・"protocols":[ ] ・"protocols":[""]	Will return an error response (400 Bad Request) due to a format error.

- Response
  - ✧ When there is a normal response

Parameters		Value
Status- Code		201 Created
Headers	Location	<Base URL>/v1/<Tenant ID>/_access_codes/<Access Code>
Body		None

  - ✧ Refer to Section 2.3.2 for error responses.

8.2. Referencing Access Codes

- Request

Parameters		Value
Method		GET
URI	Acquiring data lists	<Base URL>/v1/<Tenant ID>/_access_codes/<Access Code>?<QUERY>
	Acquiring the number of hits	<Base URL>/v1/<Tenant ID>/_access_codes/<Access Code>/_count?<QUERY>
Headers		Refer to Section 2.3.1
Body		None

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the target tenant	M
QUERY	Search conditions (described later). When omitted (including ?), it is treated as no condition set (all).	O

(*1) M: Mandatory, O: Option

✧ QUERY list

Format	Description
$filter=<filter condition>	Results are limited to only those that match the <filter condition> set. A <filter condition> is set as the "property name operator condition value", capable of defining multiple filter conditions by "and". Usable operators and property names are described below.
$top=n	Limits the amount of data returned as search results to n items. Up to a maximum of 1,000 items can be set.
$skip=n	Skips n items of data returned as search results. The order of data returned as search results is sorted by access code.

1. $top and $skip are ignored when /_count is specified. (Only $filter can be used in conjunction with this)

✧ filter condition operators

Operator	Description	Example
eq	Equal sign	?$filter=_resource_path eq 'hoge'

✧ Property names usable for filter conditions

Property name	Description	Notes
_resource_path	Resource path	Percentage encoding not required, including "/".

✧ Support functions for filter conditions

Function	Description	Example
bool startswith(string p0, string p1)	Prefix match	?$filter=startswith(_resource_path, 'hoge') eq true

1. "false" not supported.
2. An absolute match is specified when _resource_path eq hoge is specified under $filter conditions, and a prefix match is specified when startswith() is used.

- Response

[Acquiring data lists]

✧ When there is a normal response

Parameters		Value
Status- Code		200 OK
Headers	Content- Type	application/json; charset=UTF- 8
Body		Matching access code information

1. Body
   - Information items for each access code are JSON objects, and multiple information items can be stored in a JSON array format as shown below.
   - Access code information is sorted by access code (access_code). (ascending order)

Parameters	Format	Description	Maximum length
access_codes	- (array)	Starting tag	-
access_code	string	Access code	48 characters
permissions	-	Permissions information	-
ip_filter	string (array)	Permitted IP address range information used to define IP filters. Specify two IP addresses in the following format: ["Starting IP address", "Ending IP address"].	35 characters
resource_operations	- (array)	Permission information tied to a resource	
resource_path	string	Resource path	128 characters
operations	string (array)	Either "hierarchy_get"、"hierarchy_put", "create", "read", "update", "delete", or "list". Multiple values can be specified in an array.	72 characters
certification_info	—	Client certificate information	
certification	string	Client certificate ・PEM format Include "-----BEGIN CERTIFICATE-----", "-----END CERTIFICATE-----". Register line break code as "\n".	10000 characters
certificate_usage	string	Client certificate application Please specify the following. ・auth: Use for client authorization.	4 characters
protocols	string (array)	Information to specify protocols: "http" , "https", "mqtt" or tp" , " Any conbinations can be specified by array.	29 characters

   - When no value is set for the parameter, the parameter itself is not stored in a response.
   - Access codes with permissions that exceed the permissions granted to the access code stored to the Authorization header of the request are not stored in a response.
   - "protocols" are not notified when "protocols" are not specified at access code registration.

✧ For error responses

1. A "404 Not Found" message will be returned if no access code information matching the conditions set is found.
2. A “401 Unauthorized” error will be returned when an access code has not been registered to the Authorization header.
3. A “400 Bad Request” is returned if the number of access code items matching the conditions set exceeds 1,000 items.

- In this case, acceptable_top=n is stored to the Body as the top number of data items that can be received for the error response.

4. Refer to Section 2.3.2 for information about other errors

[Acquiring the number of hits]

- When there is a normal response

Parameters		Value
Status- Code		200 OK
Headers	Content- Type	text/plain
Body		Number of matching access codes

1. Body
   - Returns the number of matching access codes as is
   - Access codes with permissions that exceed the permissions granted to the access code stored to the Authorization header are not counted.

- Refer to Section 2.3.2 for when there is an error response

8.3. Updating Access Codes

- Request

Parameters	Value
Method	PUT
URI	<Base URL>/v1/<Tenant ID>/_access_codes/<Access Code>
Headers	Refer to 2.3.1 and below
Body	Access code information

- URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the target tenant	M
Access Code	Access code	M

(*1) M: Mandatory, O: Option

- HTTP header

Headers	Value	Additional notes	M/O(*1)
Content- Type	application/json; charset=UTF- 8	Fixed	M

(*1) M: Mandatory, O: Option

- Body: Access code information
  1. Refer to Section8.1
  2. Overwrites all target access code information with the updated data.
     If A and B have been registered as a setting parameter before update, and only A is specified at update, B shall be considered to, and actually deleted.

- Response
  - ✧ When there is a normal response

Parameters	Value
Status- Code	200 OK
Headers	Refer to 2.3.2
Body	None

  - ✧ Refer to Section 2.3.2 for error responses

8.4. Delete access code

- Request

Parameters	Value
Method	DELETE
URI	<Base URL>/v1/<Tenant ID>/_access_codes/<Access Code>
Headers	Refer to Section 2.3.1
Body	None

  - ✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the target tenant	M
Access Code	Access code	M

(*1) M: Mandatory, O: Option

- Response
  - ✧ When there is a normal response

Parameters	Value
Status- Code	204 No Content
Headers	Refer to Section 2.3.2
Body	None

  - ✧ For error responses

1. Access codes subject to deletion that are associated with an event cannot be deleted. In this case, the event ID associated with the status code 423 Locked will be returned.
2. Refer to Section 2.3.2 for information about other errors

Chapter 9 Controlling Events (REST)

The interface for controlling events is described below. REST(HTTP) is always used as the protocol.

Table 12: Event Control Interface List

Type	Purpose	Chapter
Register event	Registers a new event. [Option] None	Chapter 9.1
Reference event information	References (retrieves) event information. [Search option] - Specify event ID - Specify resource path (allows for prefix match searches)	Chapter 9.2
Update event information	Updates event information. [Option] None	Chapter 9.3
Delete event	Deletes the event. [Option] None	Chapter 9.4

9.1. Register event

- Request

Parameters	Value
Method	POST
URI	<Base URL>/v1/<Tenant ID>/_events
Headers	Refer to 2.3.1 and below
Body	Event information

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the target tenant	M

(*1) M: Mandatory, O: Option

✧ HTTP header

Headers	Value	Additional notes	M/O(*1)
Content- Type	application/json; charset=UTF- 8	Fixed	M

(*1) M: Mandatory, O: Option

✧ Body: Event information

Registers event information in JSON format as shown below.

Parameters	Format	M/O (*1)	Description	Maximum length
Event	-	M	Starting tag	-
Conditions	-	M	Event conditions	-

		Targets	-	M	Available	-
		resource_path	string	M	Target resource path Use "_bin/" at the start of the target resource path to control binary data.	128 characters
		operations	string (array)	M	Controlling target resource data (This is different from access code access permissions.) - For regular resources, specify ["create", "update"]. - For Resource_Binary resources, specify "create".	33 characters
		read_access_code	string	M	Access code with a target resource path with read permission or hierarchy_get permisssion	48 characters
		notification_condition	-	O	Notification conditions	-
		start_time	The date and time or time	O (*2)	The start date and time, or time(*3)	20 characters
		end_time	The date and time or time	O (*4)	The end date and time, or time(*3)	20 characters
		body_conditions	-	O	Data body conditions Cannot be set for binary resources	-
			path_type	string	M	path format Only "JSONPath" can be specified with Step1.0	8 characters
			path	string	M	Path for specifying body elements	1,902 characters
			comparing_operator	string	M	Comparative operators Either "eq" (=), "ne" (≠)," gt" (>), "ge" (>=), "lt" (<), "le" (=<), or "substring_of" (prefix match).(*5)	12 characters
			value	string or numerical value	M	Value to be compared	For character strings, 1 to 128 characters For whole numbers, -2,147,483,648 to 2,147,483,647

						For real numbers, 10 whole number part digits and 5 decimal part digits
		awake_condition		-	0	Event suppression removal conditions. Cannot be set for binary resources. *This setting will cause the event to go into a suppressed state after the notification_condition is met and the event notification is performed. Set the conditions to release the event from the suppressed state, here.	-
			body_conditions	-	M	Data body conditions	-
				path_type	string	M	path format Only "JSONPath" can be specified.	8 characters
				path	string	M	Path for specifying body elements	1902 characters
				comparing_operator	string	M	Comparative operators. Either "eq" (=), "ne" (≠)," gt" (>), "ge" (>=), "lt" (<), "le" (=<), or "substring_of" (prefix match). (*5)	12 characters
				value	string or numerical value	M	Value to be compared	For character strings, 1 to 128 characters For whole numbers, -2,147,483,648 to 2,147,483,647 For real numbers, 10 whole number part digits and 5 decimal part digits
	notification				- (*6)	M	Content of notification	-
		http			-	0	HTTP notification settings	-
			method		string	M	Either "GET", "POST", "PUT", "DELETE", "HEAD", "OPTIONS", or "TRACE".	7 characters
			Uri		string	M	URI. "http://" or "https://"	256 characters
			basic_auth_id		string	0	Basic authentication ID	20 characters

			basic_auth_pass	string	O	Basic authentication password	20 characters
			header_fields	- (array)	O	HTTP header (Maximum of 10 elements in an array)	-
			field_name	string	M	Header field name. ":" not included.	20 characters
			field_value	string	M	Value stored to the above- mentioned header field	512 characters
			body	string	O	Value stored to the body When omitted, the resource data body, event ID, event occurrence time, or the target resource path used as the event trigger is used	1024 characters
		Smtp		-	O	SMTP notification settings	-
			send_to	string	M	Notification destination e- mail address	256 characters
			subject	string	O	Subject	256 characters
			body	string	M	Body	140 characters

(*1) M: Mandatory, O: Option
The M/O mark for each child element represents whether it is necessary to configure the element when the parent element has been configured or not.
(*2) The start_time and end_time cannot beset independtly.
Both the start_time and end_time must use the same data and time, or time format.
(*3) Dates and times must meet ISO8601 requirements (20141225T103612Z etc.). Times can be specified down to the second.
When specifying the time, follow the ISO8601 format with year, month and date removed (T103612Z etc.). As above, times can be specified down to the second. Timezones are specified in "+-hhmm" format, with a "Z" added when omitted.
(*4) The start_time and end_time cannot be specified independently. Both the start_time and end_time must use the same date and time, or time format.
(*5) eq, ne, gt, ge, lt, and le can be specified when value is a numerical value. eq, ne, and substring_of can be specified when value is a character string. Case sensitivity applies when comparing character strings.
(*6) Either http or smtp settings are required under notification.

- Response
  - ✧ When there is a normal response

Parameters		Value
Status- Code		201 Created
Headers	Location	<Base URL>/v1/<Tenant ID>/_events/<Event ID>
Body		None

1. The URI of the created event is stored to Location.

  - ✧ Refer to Section 2.3.2 for when there is an error response

9.2. Referencing Event Information

- Request

Parameters			Value
Method			GET
URI	Referencing by specifying the event ID		<Base URL>/v1/<Tenant ID>/_events/<Event ID>
	Searching with QUERY	Acquiring data lists	<Base URL>/v1/<Tenant ID>/_events?<QUERY>
		Acquiring the number of hits	<Base URL>/v1/<Tenant ID>/_events/_count?<QUERY>

Headers	Refer to 2.3.1
Body	None

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the target tenant	M
Event ID	The event ID - When omitted (“/events”) targets all events for the target tenant. (Cannot be written as “/events/”.)	O
QUERY	Search conditions (described later). When omitted (include ?), it is treated as no condition set (all).	O

(*1) M: Mandatory, O: Option

✧ QUERY list

Format	Description
$filter=<filter condition>	Results are limited to only those that match the <filter condition> set. A <filter condition> is set as the "property name operator condition value", capable of defining multiple filter conditions by "and". Usable operators and property names are described below.
$top=n	Limits the amount of data returned as search results to n items. The maximum number of acquirable items is 1,000.
$skip=n	Skips n items of data returned as search results. The order of data returned as search results is sorted by the Event ID.

1. $top and $skip are ignored when /_count is specified (only $filter can be used in conjunction with this).

✧ Filter condition operators

Operator	Description	Example
Eq	Equal sign	?$filter=_resource_path eq ‘hoge’

✧ Property names usable for filter conditions

Property name	Description	Notes
resource_path	Resource path	Percentage encoding not required, including “/”

✧ Support functions for filter conditions

Function	Description	Example
bool startswith(string p0, string p1)	Prefix match	?$filter=startswith(_resource_path, ’hoge’) eq true

1. “false” not supported.
2. An absolute match is specified when _resource_path eq hoge is specified under $filter conditions, and a prefix match is specified when startswith() is used.

- Response

[Reference by specifying the Event ID, or acquire the data list]

✧ When there is a normal response

Parameters	Value
Status- Code	200 OK

Headers	Content- Type	application/json; charset=UTF- 8
Body		Matching event information

1. Body
   - Event information items are JSON objects, and multiple information items can be stored in a JSON array format as shown below. (Bold: Difference from the time of registering the event)
   - All event information is sorted by Event ID (event_id) in ascending order.

Parameters	Format	M/O (*1)	Description	Maximum length
events	-	M	Starting tag	-
event_id	string	M	Event ID	12 characters
conditions	-	M	Event conditions	-
targets	-	M	Available	-
resource_path	string	M	Target resource path	128 characters
operations	string (array)	M	Target resource data operation - For JSON resources, specify ["create", "update"]. - For Binary resources, specify ["create"].	33 characters
read_access_code	string	M	Access code with a target resource path with read permission or hierarchy_get permission	48 characters
notification_condition	-	O	Notification conditions	-
start_time	The date and time or time	O	The start date and time, or time	20 characters
end_time	The date and time or time	O	The end date and time, or time	20 characters
body_conditions	-	O	Data body conditions	-
path_type	string	M	path format. Please specify "JSONPath".	8 characters
path	string	M	Path for specifying body elements	1902 characters
comparing_operator	string	M	Comparative operators. Either "eq" (=), "ne" (≠)," gt" (>), "ge" (>=), "lt" (<), "le" (=<), or "substring_of" (prefix match).	12 characters
value	string or numerical value	M	Value to be compared.	For character strings, 1 to 128 characters For whole numbers, -

						2,147,483,648 to 2,147,483,647 For real numbers, 10 whole number part digits and 5 decimal part digits
		awake_condition	-	O	Event suppression removal conditions *This setting will cause the event to go into a suppressed state after the notification_condition is met and the event notification is performed. Set the conditions to release the event from the suppressed state here.	-
			body_conditions	-	O	Data body conditions	-
				path_type	string	M	path format. Please specify "JSONPath".	8 characters
				path	string	M	Path for specifying body elements	1,902 characters
				comparing_operator	string	M	Comparative operators Either "eq" (=), "ne" (≠)," gt" (>), "ge" (>=), "lt" (<), "le" (=<), or "substring_of" (prefix match).	12 characters
				value	string or numerical value	M	Value to be compared.	For character strings, 1 to 128 characters For whole numbers, -2,147,483,648 to 2,147,483,647 For real numbers, 10 whole number part digits and 5 decimal part digits
	notification				-	M	Content of notification	-
		http			-	O	HTTP notification settings	-
			method		string	M	Either "GET", "POST", "PUT", "DELETE", "HEAD", "OPTIONS", or "TRACE".	7 characters

			uri	string	M	URI. “http://” or “https://”	256 characters
			basic_auth_id	string	O	Basic authentication ID	20 characters
			basic_auth_pass	string	O	Basic authentication password	20 characters
			header_fields	- (array)	O	HTTP header	-
			field_name	string	M	Header field name. “:” not included.	20 characters
			field_value	string	M	Value stored to the above- mentioned header field	512 characters
			body	string	O	Value stored to the body When omitted, the resource data body, event ID, event occurrence time, or the target resource path used as the event trigger is used.	1024 characters
		smtp		-	O	SMTP notification settings	-
			send_to	string	M	Notification destination e- mail address	256 characters
			subject	string	O	Subject	256 characters
			body	string	M	Body	140 characters

(*1) M: Mandatory, O: Option
The M/O mark for each child element represents whether it is necessary to configure the element when the parent element has been configured or not.

- When no value is set for the parameter, the parameter itself is not stored in a response.
- Event information for resources for which list permissions have not been granted for the access code stored to the Authorization header of the request are not stored in a response.

✧ For error responses

1. A “404 Not Found” message will not be returned if no event information matching the conditions set is found.
2. A “400 Bad Request” is not returned if the number of events matching the conditions set exceeds 1,000 events.
   - In this case, acceptable_top=n is stored to the Body as the top number of data items that can be received for the error response.
3. Refer to Section 2.3.2 for information about other errors

[Acquiring the number of hits]

✧ When there is a normal response

Parameters		Value
Status- Code		200 OK
Headers	Content- Type	text/plain
Body		Amount of matching resource data

1. Body
   - Returns the amount of matching events as is.
   - Events for resources for which list permissions have not been granted for the access code stored to the Authorization header of the request are not counted.

✧ Refer to Section 2.3.2 for when there is an error response

9.3. Updating Event Information

- Request

Parameters	Value
Method	PUT
URI	<Base URL>/v1/<Tenant ID>/_events/<Event ID>
Headers	Refer to 2.3.1 and below
Body	Event information

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the target tenant	M
Event ID	Event ID	M

(*1) M: Mandatory, O: Option

✧ HTTP header

Headers	Value	Additional notes	M/O(*1)
Content- Type	application/json; charset=UTF- 8	Fixed	M

(*1) M: Mandatory, O: Option

✧ Body: Event information

1. Refer to Section 2.3.2.
2. Overwrites all target event information with the updated data.
   If A and B have been registered as a setting parameter before update, and only A is specified at update, B shall be considered to, and actually deleted.

- Response

✧ When there is a normal response

Parameters	Value
Status- Code	200 OK
Headers	Refer to Section 2.3.2
Body	None

✧ Refer to Section 2.3.2 for when there is an error response

9.4. Deleting Events

- Request

Parameters	Value
Method	DELETE
URI	<Base URL>/v1/<Tenant ID>/_events/<Event ID>
Headers	Refer to Section 2.3.1
Body	None

✧ URI parameters

Parameter name	Description	M/O(*1)
Base URL	Refer to Section 2.3.1	M
Tenant ID	Identifier for the target tenant	M
Event ID	Event ID	M

(*1) M: Mandatory, O: Option

- Response
  - ✧ When there is a normal response

Parameters	Value
Status- Code	204 No Content
Headers	Refer to 2.3.2
Body	None

  - ✧ Refer to Section 2.3.2 for when there is an error response

Chapter 10 Referencing external system invoking

In following functions of this system, external systems are invoked. This chapter describes specifications of external system invoking.

- Calling APIs (event function)
- Email (event function)
- Calling APIs (Resource(Transfer)_JSON function)

10.1. Calling APIs (event function)

Details concerning the API called when "Call API" is selected as an action in an event are as follows.

- Request

Parameters	Value
Method	<Method specified for the event>
URI	<URI specified for the event>
Headers	Host: <hostname which is included in URI specified for the event>
	<Header name specified for the event>:<Header value specified for the event> (multiple)
	If an authentication ID and authentication password are included: Authorization: Basic <Authentication ID: Authentication password with Base64 encoding>
Body	If a value is set for the Body: Value set for the Body
	If a value is not set for the Body: Table 13 data is stored in JSON format

[note] This service calls APIs according to HTTP/1.1 speficications, but sometimes it may defer from that specifications due to user's settings or implementation of this service. Please contact us in case of any problem occurs.

Table 13: JSON data stored as event notification data

Name	value
event_id	Event ID
date	Time at which the event judgment was made[1]
resource_path	Resource path used for the event
operation	Controls performed for the resource path create means register, update means update
body	Body text registered or updated for the resource path

If the event ID is "123456789abc", the date and time in which the event judgment was made would be March 1, 2016, 00:00:00 Japan time.
If "Printer/1F/Printer" is specified for the resource path, and "{"temperature" : 24}" is registered for the BODY text, an example of the BODY text sent for the event would be as follows.

```
{
```

[1] Conforms to ISO8601 standards (use standard millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).
*Seconds and milliseconds are separated with a ".". Timezones are specified in "+-hhmm" format, UTC time used

```
  "event_id" : "123456789abc",
  "date" : "20160301T000000.000+0900",
  "resource_path" : "Printer/1F/Printer",
  "operation" : "create",
  "body " :
    {
      "temperature" : 24
    }
}
```

- Response

Needs to match the HTTP message format specification. Only 'status-line' needs to be included. Details process specifications related to 'status-code' are as follows.

status-code	Judgment of call accepting	Retry
2xx	successed	no
408, 503, 504, 509 no response	failed	yes
else above	failed	no

10.2. Email (event function)

Details concerning the email sent when "Email" is selected as an action in an event are as follows.

Parameters		Value
To		Email address specified for the event
From		eventinfo@<zone>.fujitsu.com(*1)
Return- Path		iot- system- info@ml.css.fujitsu.com
Subject		Subject specified for the event
Body	Content- Type	text/plain; charset=UTF- 8
	Body	Body text specified for the event
Attachment	Content- Type	text/plain; charset=UTF- 8; name="eventinfo.txth
	Body	Table 13 data is stored in JSON format

(*1) The domain name depends on the service provision system. For details information of "<zone>" etc., please follow the notification received after signing up for a contract.

10.3. Calling APIs (Resource(Transfer)_JSON)

API details that is called when using Resource(Transfer)_JSON is as follows.

- Request

Parameters	Value
Method	<Method specified for the Resource(Transfer)_JSON>
URI	<URI specified for the Resource(Transfer)_JSON>
Headers	Host: <hostname which is included in URI specified for the Resource(Transfer)_JSON>
	<Header name specified for the Resource(Transfer)_JSON>:<Header value specified for the Resource(Transfer)_JSON> (multiple)
	If an authentication ID and authentication password are included: Authorization: Basic <Authentication ID: Authentication password with Base64 encoding>
Body	Data registerd to Resource(Transfer)_JSON.

[note] This service calls APIs according to HTTP/1.1 speficications, but sometimes it may defer from that specifications due to user's settings or implementation of this service. Please contact us in case of any problem occurs.

- Response

Needs to match the HTTP message format specification. Only 'status-line' is required. Details process specifications related to 'status-code' are as follows.

status-code	Judgment of call accepting	Retry
2xx	successed	no
408, 503, 504, 509 no response	failed	yes
else above	failed	no

Chapter 11 Referencing Data stored by the System

This service includes data that is autonomously stored by the system on resources. This chapter describes the format and contents of such stored data.

11.1. Recommend Resource

Recommend values referenced by the gateway device are stored in JSON format by the Dynamic Resource Controller.

Recommended resource name	Set from the Service Portal

Parameters	Format	M/O(*1)	Description
recommend_value	string	M	The recommend value (“ON” or “OFF”)
recommend_parameter	string	M	The recommend parameter (Argument configured in distribution settings on the Service Portal) However, “ALM” is used when there is an extreme system load placed on this service)

(*1) M: Mandatory, O: Option

The following example shows JSON data stored to the recommend resource when the recommend value is changed to “ON”, and the edge computing condition parameter is “recommend is on”.

```
{
    "recommend_value" : "ON”
    "recommend_parameter" : "recommend is on"
}
```

11.2. Load Resource

API access load measurement results are stored in JSON format by the Dynamic Resource Controller.

Load resource name	Set from the Service Portal

Parameters	Format	M/O(*1)	Description
tps	string	M	The load data on the customer tenant (tps) [transaction/second]
bps	string	M	The load data on the customer tenant (bps) [bit/second]

(*1) M: Mandatory, O: Option

The following example shows JSON data stored to the load resource if the load data is 10 tps and 10,000 bps.

```
{
    "tps" : "10",
    "bps" : "10000"
}
```

11.3. Error Collection Resource

The error log storage function stores error information when requests are lost due to issues within the service and the transmission destination. Confirm processing results around the time of occurrence before proceeding with countermeasures.

Error collection resource name	_error

Parameters		Format	M/O (*1)	Description
collection_first_time		Timestamp(*2)	M	Start time for error collection period
collection_last_time		Timestamp(*2)	M	End time for error collection period
errors		— (array)	M	Error information
	trigger	string	M	Cause of failure.
	resource_path	string	M	The resource path where an error has occurred. The resource path with the oldest information within the error collection period is set.
	data_registration_time	Timestamp(*2)	O	Resource data registration time. Only set when the cause of failure is an event.
	event_id	string	O	Event ID Only set when the cause of failure is an event.
	error_factor_message	string	M	Cause of failure message. Reference the following.
	number_of_times	Numerical values	M	Number of times that the cause of failure and the cause of failure message was the same within an error collection period.
	request_id	array of string	O	The value of x-iotpf-request-id specified by an error request is set.
message		string	O	The following is set in case of failing to register the error information. ・ 'request_id' was not able to be registered because of too many numbers.

(*1) M: Mandatory, O: Option
(*2) Conforms to ISO8601 standards (use basic notation millisecond expressions) (20141225T103612.001Z, etc.). Millisecond-level precision used (when omitting milliseconds, the system will read this as 0 milliseconds).
*Seconds and milliseconds are separated with a ".". Timezones are specified in "±hhmm" format, with a "Z" added when omitted. This service uses UTC time when storing in a response.

The cause of failure message list contained in error_factor_message is as follows.

trigger	Cause of failure message	Description
Transfer Resource	It failed in the transmission of forwarding data. (IoT-PF internal error)	Forwarding failure: Failure due to system error.
Transfer Resource	It failed in the transmission of forwarding data. (Forwarding parameter error)	Forwarding failure: Failure due to insufficient forwarding settings information.
Transfer Resource	It failed in the transmission of forwarding data. (Connection error(HTTP))	Forwarding failure: Failure due to connection error (HTTP).

Transfer Resource	It failed in the transmission of forwarding data. (Connection error(TCP))	Forwarding failure: Failure due to connection error (TCP).
Event	Event connection failed.	Event sending failure: Failed to connect to event destination.
Event	Event transmission retry over. code=408	Event sending failure: Attempt to resend event with response code 408 failed due to being over the resend limit.
Event	Event transmission failed. code=404	Event sending failure: Attempt to send event with response code 404 failed.
Event	Event transmission retry over. code=503	Event sending failure: Attempt to resend event with response code 503 failed due to being over the resend limit.
Event	Event transmission failed. code=500	Event sending failure: Attempt to send event with response code 500 failed.
Event	Event Send Request was rejected due to inflow regulation.	Event sending failure: Attempt to send event by the overload.
Event	Fail to send SMTP mail.	Event sending failure: Attempt to send event(EMail) by the internal factors of this service.
Event	Retry out to send SMTP mail.	Event sending failure: Attempt to send event(Email) by the communication error with mail server.
MQTT Broker	Failed to execute publishing resource data, in the system internal processing.	MQTT data storage failure: Failed to store resource data after data loss during internal processing.
MQTT Parser	Failed to execute publishing resource data, in the system internal processing.	MQTT data storage failure: Failed to store resource data after data loss during internal processing.
MQTT Parser	Failed to execute publishing resource data, in invalid data format.	MQTT data storage failure: Failed to store resource data because of the form that the registration data (payload JSON/registration date/request identification ID etc.) did not apply to the specification.
MQTT Parser	Failed to execute publishing resource data, in message payload length error.	MQTT data storage failure: Failed to store resource data because data size(payload length) is too big.

Appendix 1 List of Response Error Messages

Here, it explains the content notified in the Body when an API request is generated and returns an error.. The error is notified in the form of the following.

Field name	Value
errors.message	Error text
errors.acceptable_top	The top count for the response Body size to be 16 MBs or less (Only when the error code is “number of response- data is larger than 1000” or “response size is larger than 16MB”)

The BODY text example is as follows.

```
{ “errors”: [
    { “message”: “response size is larger than 16MB”,
        “acceptable_top”: 100
     }
  ]
}
```

The error text list contained in the errors.message is as follows.

HTTP response	Error body text	Countermeasure
503 Service Unavailable	{"errors":[{"message":"[CREATE] iot- pf is temporarily unavailable.[11007]"}]}	An unexpected internal error has been detected. Please notify the help desk about the error message.
503 Service Unavailable	{"errors":[{"message":"[CREATE] iot-pf is temporally unavailable.[11007]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[CREATE] iot-pf is temporally unavailable.[11009]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[CREATE] iot-pf is temporally unavailable.[11010]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[CREATE] iot-pf internal error.[11011]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[CREATE] iot-pf internal error.[11013]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[CREATE] iot-pf internal error.[11016]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[CREATE] iot-pf internal error.[11017]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[CREATE] iot-pf is temporally unavailable.[11022]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[CREATE] No Connection.[11027]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.

503 Service Unavailable	{"errors":[{"message":"[SEARCH] iot-pf is temporally unavailable.[11028]"}]}	This error may occur due to system overloading. We recommend refining your search conditions and trying again.
503 Service Unavailable	{"errors":[{"message":"[SEARCH] iot-pf is temporally unavailable.[11030]"}]}	This error may occur due to system overloading. We recommend refining your search conditions and trying again.
503 Service Unavailable	{"errors":[{"message":"[SEARCH] iot-pf is temporally unavailable.[11031]"}]}	This error may occur due to system overloading. We recommend refining your search conditions and trying again.
500 Internal Server Error	{"errors":[{"message":"[SEARCH] iot-pf internal error.[11032]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[SEARCH] iot-pf internal error.[11034]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[SEARCH] iot-pf internal error.[11036]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[SEARCH] iot-pf internal error.[11037]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[SEARCH] iot-pf is temporally unavailable.[11039]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[SEARCH] No Connection.[11040]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[UPDATE] iot-pf is temporally unavailable.[11041]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[UPDATE] iot-pf is temporally unavailable.[11043]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[UPDATE] iot-pf is temporally unavailable.[11044]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[UPDATE] iot-pf internal error.[11045]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[UPDATE] iot-pf internal error.[11047]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[UPDATE] iot-pf internal error.[11049]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[UPDATE] iot-pf internal error.[11050]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[UPDATE] iot-pf is temporally unavailable.[11051]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[UPDATE] No Connection.[11052]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[REMOVE] iot-pf is temporally unavailable.[11053]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[REMOVE] iot-pf is temporally unavailable.[11055]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[REMOVE] iot-pf is temporally unavailable.[11056]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.

500 Internal Server Error	{"errors":[{"message":"[REMOVE] iot-pf internal error.[11057]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[REMOVE] iot-pf internal error.[11059]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[REMOVE] iot-pf internal error.[11061]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[REMOVE] iot-pf internal error.[11062]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[REMOVE] iot-pf is temporally unavailable.[11063]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[REMOVE] No Connection.[11064]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[CREATE] iot-pf is temporally unavailable.[12003]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
400 Bad Request	{"errors":[{"message":"[CREATE] main data is required."}]}	No BODY text. Please check that BODY text has been written correctly.
400 Bad Request	{"errors":[{"message":"[CREATE] main data is too large."}]}	The BODY text is too large. Please reduce the BODY text by partitioning the data.
503 Service Unavailable	{"errors":[{"message":"[CREATE] iot-pf is temporally unavailable.[12006]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
400 Bad Request	{"errors":[{"message":"[CREATE] url format error."}]}	Invalid URL specified. Please check the URL.
400 Bad Request	{"errors":[{"message":"[CREATE] query num invalid."}]}	Multiple instances of "?" are included in the URL. Please check the query.
400 Bad Request	{"errors":[{"message":"[CREATE] query too large."}]}	The query is too large. Please check the query.
400 Bad Request	{"errors":[{"message":"[CREATE] url unescape error."}]}	Invalid URL specified. Please check the URL.
400 Bad Request	{"errors":[{"message":"[CREATE] access code is wrong."}]}	Invalid access code. Please check that the correct access code has been entered.
400 Bad Request	{"errors":[{"message":"[CREATE] x-iotpf-request-id format error."}]}	Invalid value of x-iotpf-request-id.
400 Bad Request	{"errors":[{"message":"[SEARCH] url format error."}]}	Invalid URL specified. Please check the URL.
400 Bad Request	{"errors":[{"message":"[SEARCH] query num invalid."}]}	Multiple instances of "?" are included in the URL. Please check the query.
400 Bad Request	{"errors":[{"message":"[SEARCH] query too large."}]}	The query is too large. Please check the query.
400 Bad Request	{"errors":[{"message":"[SEARCH] url unescape error."}]}	Invalid URL specified. Please check the URL.
400 Bad Request	{"errors":[{"message":"[SEARCH] access code is wrong."}]}	Invalid access code. Please check that the correct access code has been entered.
503 Service Unavailable	{"errors":[{"message":"iot-pf is temporally unavailable.[12026]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
400 Bad Request	{"errors":[{"message":"[UPDATE] main data is required."}]}	No BODY text. Please check that BODY text has been written correctly.
400 Bad Request	{"errors":[{"message":"[UPDATE] main data is too large."}]}	The BODY text is too large. Please reduce the BODY text by partitioning the data.

503 Service Unavailable	{"errors":[{"message":"iot-pf is temporally unavailable.[12029]"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
400 Bad Request	{"errors":[{"message":"[UPDATE] url format error."}]}	Invalid URL specified. Please check the URL.
400 Bad Request	{"errors":[{"message":"[UPDATE] query num invalid."}]}	Multiple instances of "?" are included in the URL. Please check the query.
400 Bad Request	{"errors":[{"message":"[UPDATE] query too large."}]}	The query is too large. Please check the query.
400 Bad Request	{"errors":[{"message":"[UPDATE] url unescape error."}]}	Invalid URL specified. Please check the URL.
400 Bad Request	{"errors":[{"message":"[UPDATE] access code is wrong."}]}	Invalid access code. Please check that the correct access code has been entered.
400 Bad Request	{"errors":[{"message":"[UPDATE] x-iotpf-request-id format error."}]}	Invalid value of x-iotpf-request-id.
400 Bad Request	{"errors":[{"message":"[REMOVE] url format error."}]}	Invalid URL specified. Please check the URL.
400 Bad Request	{"errors":[{"message":"[REMOVE] query num invalid."}]}	Multiple instances of "?" are included in the URL. Please check the query.
400 Bad Request	{"errors":[{"message":"[REMOVE] query too large."}]}	The query is too large. Please check the query.
400 Bad Request	{"errors":[{"message":"[REMOVE] url unescape error."}]}	Invalid URL specified. Please check the URL.
400 Bad Request	{"errors":[{"message":"[REMOVE] query must not be exists. for present"}]}	Query specified despite most recent deletion (_present). Remove query and try again.
400 Bad Request	{"errors":[{"message":"[REMOVE] query is required. for past."}]}	Query not specified despite past data deletion (_past). Please specify a query.
400 Bad Request	{"errors":[{"message":"[REMOVE] access code is wrong."}]}	Invalid access code. Please check that the correct access code has been entered.
400 Bad Request	{"errors":[{"message":"[REMOVE] x-iotpf-request-id format error."}]}	Invalid value of x-iotpf-request-id.
429 Too Many Requests	{"errors":[{"message":"Number of request per second has exceeded maximum usage limit of service contract."}]}	The access frequency exceeds the system capacity. Please review the access frequency.
400 Bad Request	{"errors":[{"message":"fail to get decompressed data size."}]}	Invalid compression form.
400 Bad Request	{"errors":[{"message":"decompressed data is too large."}]}	Data length after decompression is too large.
423 Locked	{"errors":[{"message":" request access code has event. "}]}	Event associated with target access code found. Please remove event associations.
423 Locked	{"errors":[{"message":" request access code can't read event."}]}	The read permission cannot be deleted because event associated with target access code found.
423 Locked	{"errors":[{"message":" request access code has dispersion policy."}]}	Enabled distribution policy associated with target access code found. Please remove the distribution policy association settings.
423 Locked	{"errors":[{"message":" request access code has recommend resource."}]}	Enabled recommend resource associated with target access code found. Please remove the recommend resource association

		settings.
423 Locked	{"errors":[{"message":" request access code can't update dispersion policy."}]}	Enabled distribution policy associated with target access code found.
423 Locked	{"errors":[{"message":" request access code can't update recommend resource."}]}	The update permission cannot be deleted because enabled recommend resource associated with target access code is found.
423 Locked	{"errors":[{"message":" request access code and resource have event."}]}	Enabled event associated with target access code and resource found. Please remove event associations.
503 Service Unavailable	{"errors":[{"message":" iot-pf Service Unavailable."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":" iot-pf internal error."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
404 Not Found	{"errors":[{"message":" Reload configuration to fail to read config."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
400 Bad Request	{"errors":[{"message":" tenant id is required. "}]}	Tenant ID not specified.
400 Bad Request	{"errors":[{"message":" access code is requred. "}]}	Access code not specified.
400 Bad Request	{"errors":[{"message":" input parameter error. : resource_path is duplicated. (Resourcepath=[resource path])"}]}	Duplicate resource path found.
400 Bad Request	{"errors":[{"message":" parameter duplication. "}]}	Duplicate parameters found.
400 Bad Request	{"errors":[{"message":" request access code already exists. "}]}	The access code specified already exists.
400 Bad Request	{"errors":[{"message":" request resource path does not exist. : ResourcePath= [Resource path]"}]}	The resource path specified does not exist.
404 Not Found	{"errors":[{"message":" resource path not found. "}]}	The target resource ID has not been registered.
404 Not Found	{"errors":[{"message":" access code not found. "}]}	An access code matching search conditions was not found.
404 Not Found	{"errors":[{"message":" event not found. "}]}	An event matching search conditions was not found.
404 Not Found	{"errors":[{"message":" target resource not found. "}]}	Target resource not found.
404 Not Found	{"errors":[{"message":" tenant ID not found. "}]}	Related tenant ID not found.
400 Bad Request	{"errors":[{"message":" event id is required. "}]}	Event ID not specified.
400 Bad Request	{"errors":[{"message":" URL FORMAT ERROR"}]}	Invalid request URL.
400 Bad Request	{"errors":[{"message":" Incorrect access code search conditions. "}]}	An error was found in the access code search conditions.
400 Bad Request	{"errors":[{"message":" Incorrect event code search conditions. "}]}	An error was found in the event search conditions.
400 Bad Request	{"errors":[{"message":" Incorrect filter condition. "}]}	An error was found in the filter conditions.
400 Bad Request	{"errors":[{"message":" input parameter is error. : incorrect top condition "}]}	An error was found with the top condition.

400 Bad Request	{"errors":[{"message":" input parameter is error. : incorrect skip condition "}]}	An error was found with the skip condition.
400 Bad Request	{"errors":[{"message":" Request data format error. "}]}	Invalid value found in request data.
400 Bad Request	{"errors":[{"message":" Number of resource path is over for one request. (resourcePathSize=[Number of resource path])"}]}	The number of resource paths that can be added with a single request has been exceeded.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : permissions"}]}	Insufficient access code permission information.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : resource_path and operations in resource_operations"}]}	Insufficient resource permission information.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : resource_path of resource_operations"}]}	Insufficient resource permission resource path.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : operations of resource_operations"}]}	Insufficient resource access permission.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : conditions"}]}	Insufficient event conditions.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : targets"}]}	Insufficient event targets.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : resource_path of targets"}]}	Insufficient event target resource path.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : operations of targets"}]}	Insufficient target resource data operation.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : read_access_code of targets"}]}	Insufficient access code with a target resource path with read permissions.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : path_type of notification_condition"}]}	Insufficient notification condition path format.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : path of notification_condition"}]}	Insufficient resource path specifying notification condition body elements.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : comparing_operator of notification_condition"}]}	Insufficient notification condition comparative operators.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : value of notification_condition"}]}	Insufficient notification condition comparative target value.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : path_type of awake_condition"}]}	Insufficient inhibit condition path format.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : path of awake_condition"}]}	Insufficient resource path specifying inhibit condition body elements.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : comparing_operator of awake_condition"}]}	Insufficient inhibit condition comparative operators.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : value of awake_condition"}]}	Insufficient inhibit condition comparative target value.

400 Bad Request	{"errors":[{"message":" input parameter error is required. : notification smtp or http"}]}	Notification content not specified.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : send_to of smtp notification"}]}	Insufficient notification destination email address.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : body of smtp notification"}]}	Insufficient notification email body text.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : method of http notification"}]}	Notification HTTP Method required.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : uri of http notification"}]}	Insufficient notification URI.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : field_name and field_value of http notification"}]}	Insufficient notification header field.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : filed_name of http notification"}]}	Insufficient notification header field name.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : filed_value of http notification"}]}	Insufficient notification header field value.
400 Bad Request	{"errors":[{"message":" input parameter error is required. : notification"}]}	Insufficient notification.
403 Forbidden	{"errors":[{"message":" Authorization accesscode is required."}]}	Access code not specified.
401 Unauthorized	{"errors":[{"message":" Authorization error. (AccessCode=[Access code], NG_ResoucePath=[Resouce path])"}]}	Resources with no permission included in the access code specified.
401 Unauthorized	{"errors":[{"message":" Client authorization error. (AccessCode=[Access code])"}]}	Accessed from a client not permitted in the access code.
401 Unauthorized	{"errors":[{"message":" Authorization error. (AccessCode=[Access code])"}]}	Accessed by a protocol not permitted in the access code.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect access code operations"}]}	Incorrect combination of access code access permissions.
400 Bad Request	{"errors":[{"message":" request ipfilter does not exist. (tenanatId=[Tenant ID], startIpAddr=[IP address], endIpAddr=[IP address])"}]}	Specified access permission setting not found.
400 Bad Request	{"errors":[{"message":" hit data is larger than limit. (acceptable_top=1000)"}]}	The results exceed 1000.
403 Forbidden	{"errors":[{"message":"[12670] Authorization accesscode format error."}]}	Error found in the access code.
400 Bad Request	{"errors":[{"message":" Fail to url decode(utf-8)."}]}	Incorrect character code included.
400 Bad Request	{"errors":[{"message":" URL format error. : Url Path is required."}]}	URL path required.
400 Bad Request	{"errors":[{"message":" URL format error. : Incorrect protocol"}]}	Incorrect protocol specified.
400 Bad Request	{"errors":[{"message":" URL format error. : incorrect url path"}]}	Incorrect URL path.
400 Bad Request	{"errors":[{"message":" URL format error. : incorrect url path start"}]}	The URL path starts with an incorrect character.

400 Bad Request	{"errors":[{"message":" URL format error. : tenant id format error."}]}	Invalid tenant ID format.
400 Bad Request	{"errors":[{"message":" URL format error. : input _access_codes or _events."}]}	Invalid URL specified..
400 Bad Request	{"errors":[{"message":" URL format error. : unnecessary query is input"}]}	Unnecessary query specified.
400 Bad Request	{"errors":[{"message":" URL format error. : access code format error."}]}	Invalid access code format.
400 Bad Request	{"errors":[{"message":" URL format error. : input _counts."}]}	Invalid URL specified to get HIT count.
400 Bad Request	{"errors":[{"message":" URL format error. : event id format error."}]}	Invalid event ID format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect access code name's string length"}]}	The access code length is incorrect.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect access code discription's string length"}]}	Incorrect access code explanation length.
400 Bad Request	{"errors":[{"message":" input parameter error. : number of ipfilter is larger than 5."}]}	Over 5 access permissions specified.
400 Bad Request	{"errors":[{"message":" input parameter error. : start ipaddress format error"}]}	Invalid access permission IP address format.
400 Bad Request	{"errors":[{"message":" input parameter error. : end ipaddress format error"}]}	Invalid access permission IP address format.
400 Bad Request	{"errors":[{"message":" input parameter error. : ipfilter format error."}]}	Invalid access permission specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect ipfilter range"}]}	Invalid access permission specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : operations format error. (NG Operation kind=[Access permission])"}]}	Invalid access permission specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : operation is duplicated."}]}	Same access permission specified.
400 Bad Request	{"errors":[{"message":" input parameter error. : resource path format error."}]}	Invalid resource path specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect event name's string length"}]}	Incorrect event name string length.
400 Bad Request	{"errors":[{"message":" input parameter error. :event description's string length"}]}	Incorrect event explanation string length.
400 Bad Request	{"errors":[{"message":" input parameter error. : event target operations format eror."}]}	Invalid resource operation specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : inccorect notification_condition's start_time or end_time (start_time=[Strat time], end_time=[End time])"}]}	Invalid event judgment specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : cannot be set to body conditions to binary resource."}]}	An event condition for resource_binary specified.

400 Bad Request	{"errors":[{"message":" input parameter error. : access code format error."}]}	Error found in access code specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect path_type of notification condition"}]}	Error found in path_type notification condition specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect path_type's string length of notification condition"}]}	Incorrect notification condition path_type string length.
400 Bad Request	{"errors":[{"message":" input parameter error. : comparing_operator of notification condition (comparing_operator=[Comparing condition])"}]}	Incorrect notification condition comparing condition.
400 Bad Request	{"errors":[{"message":" input parameter error. : comparing operator format error (comparing_operator=[Comparing operator])"}]}	Incorrect notification condition comparing operator.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect path_type of awake condition"}]}	Error found in inhibit condition specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect path's string length of awake condition"}]}	Error found in inhibit condition specifying string length.
400 Bad Request	{"errors":[{"message":" input parameter error. : comparing_operator of awake condition (comparing_operator=[Comparing condition])"}]}	Error found in inhibit condition comparing condition specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect comparing operator of awake condition (comparing_operator=Comparing operator)"}]}	Error found in inhibit condition comparing condition comparing operator.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorret send_to's length of smtp notification"}]}	Error found in notification contact email address format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorret body's length of smtp notification"}]}	Error found in notification email text format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorret subject's length of smtp notification"}]}	Error found in notification email subject format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect method of http notification"}]}	Error found in notification HTTP method specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect uri's length of http notification"}]}	Incorrect notification URL specifying format string length.
400 Bad Request	{"errors":[{"message":" input parameter error. : number of header fields is larger than 10"}]}	Over 11 notification header fields specified.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect field_name's length of http	Incorrect notification header field specifying format string length.

	notification"}]}	
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect field_value's length of http notification"}]}	Incorrect notification header field specifying format string length.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect basic_auth_id's length of http notification"}]}	Incorrect notification HTTP basic authentication ID string length.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect basic_auth_pass's length of http notification"}]}	Incorrect notification HTTP basic authentication password string length.
400 Bad Request	{"errors":[{"message":" input parameter error. : certification info format error."}]}	Error found in certificate specifying format.
400 Bad Reuqest	{"errors":[{"message":" input parameter error. :protocols format error."}]}	Error found in protocol specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : certification file is too large. Size=[Size]"}]}	The certificate size exceeds the system capacity.
400 Bad Request	{"errors":[{"message":" input parameter error. : certification format error.([Certificate format string] is required)"}]}	An error found in the certificate format.
400 Bad Request	{"errors":[{"message":" input parameter error. : protocols is duplicated."}]}	Duplicate certificate specified.
500 Internal Server Error	{"errors":[{"message":" certification file parse error.([error number])"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[12401] transfer resource is temporarily unavailable"}]}	The transfer resource cannot temporarily unavailable. Please wait and try again.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect body's length of http notification"}]}	The number of items is incorrect.
403 Forbidden	{"errors":[{"message":" Ip filter NG. (tenantId=[Tenant ID], accessCode=[Access code], srcIp=[Request source IP address])"}]}	Access denied due to access permission setting.
400 Bad Request	{"errors":[{"message":" request http header error. (Header:[Header content])"}]}	Error found in HTTP header.
401 Unauthorized	{"errors":[{"message":" Authorization error. (AccessCode=[Access code])"}]}	Error found in access code.
400 Bad Request	{"errors":[{"message":" input parameter error. : uri of http notification format error."}]}	Incorrect notification HTTP Method specifying format.
400 Bad Request	{"errors":[{"message":" input parameter error. : field_name length of http notification format error."}]}	Incorrect notification HTTP Method specifying format length.
400 Bad Request	{"errors":[{"message":" input parameter error. : incorrect body_conditions of awake condition"}]}	Incorrect inhibit condition body_conditions specifying format.
400 Bad Request	{"errors":[{"message":" URL format error. : access code is necessary."}]}	Access code not specified in the URL.
400 Bad Request	{"errors":[{"message":"[12990] Illigal put data."}]}	An error found in specifying parameter.
500 Internal Server	{"errors":[{"message":"[11901] Fail to	An unexpected internal error has been detected.

Error	loadCollection."}]}	Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[12992] Abnormality occurred by PUT processing."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11903] Fail to create CommonData."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11904] Fail to get socket(PUT)."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11905] Fail to send PreProcess(PUT)."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
400 Bad Request	{"errors":[{"message":"[12980] Illigal delete data."}]}	An error found in specifying parameter.
500 Internal Server Error	{"errors":[{"message":"[11911] Fail to loadCollection."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[12912] Abnormality occurred by DELETE processing."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11913] Fail to create CommonData."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11914] Fail to get socket(DELETE)."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11915] Fail to send PreProcess(DELETE)."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
400 Bad Request	{"errors":[{"message":"[12920] Illigal get data."}]}	An error was found with the skip condition.
500 Internal Server Error	{"errors":[{"message":"[11921] Fail to loadCollection."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[12922] Abnormality occurred by GET processing."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11923] Fail to create CommonData."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11924] Fail to get socket(GET)."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11925] Fail to send PreProcess(GET)."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11935] Fail to send PreProcess."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
408 Request Timeout	{"errors":[{"message":"[11936] REST-Connection Error."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[11937] Service Unavailable.."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
500 Internal Server Error	{"errors":[{"message":"[11938]Internal Server Error."}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
503 Service Unavailable	{"errors":[{"message":"[12900] Before start service(for initialization)."}]}	System now in initializing. Please wait and try again.
429 Too Many Requests	{"errors":[{"message":"[12906]Number of request per second has exceeded maximum usage limit of service contract."}]}	Number of request per second has exceeded maximum usage limit of service contract. Please review the access frequency.
503 Service Unavailable	{"errors":[{"message":"[11937] Service Unavailable"}]}	An unexpected internal error has been detected. Please notify the support desk of the error message.
400 Bad Request	{"errors":[{"message":"number of response-data is larger than	This reference request exceeds 1,001 acquired data items (no top option).

	1000.","acceptable_top": XXXX}]}	Respecify the top count based on the acceptable_top value.
400 Bad Request	{"errors":[{"message":"response size is larger than 16MB","acceptable_top": [Top count available]}]}	This reference request exceeds 16 MB in acquired data volume. Respecify the top count based on the acceptable_top value.
503 Service Unavailable	{"errors":[{"message":"number of concurrent request for heavy queries exceeded the maximum usage limit."}]}	Too much heavy queries are requested at the same time. heavy queries are too much. Please decrease a concurrent number of requests.