FUJITSU IoT Platform provider

For REST API

Version 1.0.0

User's guide

August 31, 2018

Remarks:

・This document uses the machine translation.

【 revision history 】

Version	Date	Content
1.0.0	2018-4-16	First edition.
	2018-6-5	Mount registration (JSON,CSV,TXT,BIN), the latest data reference, and registration data HIT number acquisition command as Variable.
	2018-7-11	Mount following API command as Execute. Resource control (registration, update, and deletion) Access code control (registration, reference, update, and deletion) Event control (registration, reference, update, and deletion)
	2018-7-20	Add the response status code at REST.
	2018-7-30	Add a simple explanation to the outline about API. Add the example to the filter condition. Add the filter condition that can be used when the access code reference and the event code are referred.
	2018-8-2	Sample program addition
	2018-8-31	Sample program addition (Variable, general purpose REST, data control, resource control, access code control, and event control)

Contents

1. Introduction

The IOTPLATFORM provider is a provider that offers the connection and API call to use various services that Fujitsu IOTPLATFORM Platform provides.

Various services of the one-making of each manufacturing company and the service provider are developed by using this provider on the IOTPLATFORM core, and various knowledge concerning information etc. on plant facilities such as the one-making knowhow, the IoT equipment, PLC, and robots and equipment comes to be able to be shared, and to be used.

In this document, it explains the outline of the IOTPLATFORM provider and the mounted CAO interface (function specification).

2. Outline of provider

2.1. Installation

The IOTPLATFORM provider module is composed of following DLL. The installation work is unnecessary when installing it with the installer of ORiN2 SDK. Execute it as shown in Table 2-1 when you install it by hand power.

Table2-1IOTPLATFORM provider

File name	CaoProv.FUJITSU.IOTPLATFORM.dll
ProgID	CaoProv.FUJITSU.IOTPLATFORM
Registry registration	RegistAsm.bat CaoProv.FUJITSU.IOTPLATFORM.dll
Blotting out of registry registration	UnregistAsm.bat CaoProv.FUJITSU.IOTPLATFORM.dll

※ There are RegistAsm.bat and UnregistAsm.bat in case of { ORiN2 installation folder } ¥DotNet¥Bat.

2.2. Outline

The IOTPLATFORM provider offers the function to wrap API (general purpose REST, data control, resource control, access code control, and event control, etc.) for various service use that Fujitsu IOTPLATFORM Platform offers.

To IOTPLATFORM Platform when you easily explain the content of API

・REST is executable by general purpose REST.

・The registration destination of data can be made by the resource control.

・It can be an operation of the registration of data etc. by the data control for the resource.

・The right of access is given to the resource and it is possible to limit it by the access code control.

・In the event control, contingent on the value of the data registered in the resource

The HTTP notification or mail can notify from IOTPLATFORM Platform.

A detailed usage has been described as follows about information related to API and refer.

API reference

https://iot-docs.jp-east-1.paas.cloud.global.fujitsu.com/ja/manual/v5/apireference.pdf

User guide

https://iot-docs.jp-east-1.paas.cloud.global.fujitsu.com/ja/manual/v5/userguide.pdf

2.3. Method property

2.3.1. CaoWorkspace::AddController method

Format AddController (<bstrCtrlName:BSTR>, <bstrProvName:BSTR>, <bstrPcName:BSTR>, <bstrOption:BSTR>)

< bstrCtrlName >	:	In controller name
< bstrProvName >	:	In provider name. "CaoProv.FUJITSU.IOTPLATFORM fixed value ="
<bstrPcName>	:	Execution machine name of in provider (unused)
<bstrOption>	:	"< option 1> and < option 2> in optional character string =" ・Base URL ・Tenant ID ・Access code ・API version ・Time-out Specify it by switching off the comma district in the form of a set set item = content. Setting example: "BaseURL=http://<zone>.fujitsu.com, APIVersion=v1,…"

Details of optional character string (bstrOption) are shown as follows.

Table2-2Optional character string of CaoWorkspace:: AddController method

Set item	Set content	Indispensability	Remarks
BaseURL	http://<zone>.fujitsu.com	○	Follow the content of the notification after IoT Platform is contracted about the value that enters < zone >.
TenantID	<Tenant>	○	Resource ownership tenant's identifier Follow the content of the notification after IoT Platform is

			contracted.
AccessCode	<Access code>	○	The access code is a value set by the service portal of IOTPLATFORM.
APIVersion	<Version>	-	Specify "V1" as of March, 2018. When omitting it, "V1" is set.
Timeout	<Timeout>	-	Set the timeout period when REST is communicated in each millisecond. When omitting it, 30000 is set.

2.3.2. CaoController::AddVariable method

Add the CaoVariable object to execute a part of the Execute command of CaoController with get/put of the CaoVariable object.

Format AddVariable (<bstrName:BSTR>, [<bstrOption:BSTR>])

< bstrName > : In variable identifier

The following system variable identifier or user variable identifiers can be specified.

System variable : Specify the following reservation character strings that start by '@'.
@RegistJSON
@RegistCSV,
@RegistTXT
@RegistBIN
@ReferenceLatestData
@DataCount

User variable : Specify an arbitrary character string that doesn't start by '@'. In this case, it is indispensable to specify Type of an optional character string.

<bstrOption> : Specify the specified option of each done Execute command for in by switching off the comma district.

Specify following optional Type when you specify user variables. This Execute command that corresponds by optional Type is fixed.

When the system variable is specified, the Execute command that

corresponds according to the system variable identifier is fixed.

Type : Specify the system variable identifier for the user variable to decide behavior.
@RegistJSON
@RegistCSV,
@RegistTXT
@RegistBIN
@ReferenceLatestData
@DataCount
This option is ineffectual for the system variable.
Example)
Type=@RegistJSON

An option necessary in each corresponding Execute command is described to the following clause.

2.3.2.1. @RegistJSON

Use a system variable identifier or optional Type of user variables "@RegistJSON" specifying it. In doing JSON character string (BSTR) put,"RegistJSON" command The JSON data can be registered to the resource in similar.

Specify the following options besides optional Type when the user variable is specified to add this variable to CaoController.

Table2-3Optional character string of @RegistJSON

Set item	Indispensability	Remarks
ResourcePath	○	Specify the resource passing.
Date	-	Specify (*1) at the registration date given to the registration data. Adopt the request reception date when you omit it.
Retain	-	Specify whether to maintain this registration data on the MQTT broker side. ・True: Maintain it. ・False: Do not maintain it. When omitting it, it is assumed the one that false was

		specified. - Disregard it even if RETAIN is specified when you specify Bulk Insert.
BulkInsert	-	Specify whether to execute Bulk Insert (Transmit two or more requests to one degree). ・Do not do none:Bulk Insert. ・Single_resource_path: Execute Bulk Insert for a single resource. When omitting it, it is assumed the one that none was specified.

(*1)It follows ISO8601 (The millisecond expression as a basic mark is used)(20141225T103612.001Z etc.). Accuracy is a millisecond (When the millisecond is omitted, it is considered 0 milliseconds). The following "Registration date" are the same all specifications. * When "." and the zone specification of time are omitted by the "±hhmm" form, the delimitation of the second and the millisecond affixes "Z". Use UTC when this service stores it in the response. The following date specification follows this rule.

2.3.2.2. @RegistCSV

Use a system variable identifier or optional Type of user variables "@RegistCSV" specifying it. In doing CSV character string (BSTR) put, The CSV data can be registered to the resource in similar to "RegistCSV" command.

Specify the following options besides optional Type when the user variable is specified to add this variable to CaoController.

Table2-4Optional character string of @RegistCSV

Set item	Indispensability	Remarks
ResourcePath	○	Specify the resource passing.
Date	-	Specify the registration date given to the registration data. Adopt the request reception date when you omit it.
Retain	-	Specify whether to maintain this registration data on the MQTT broker side. ・True: Maintain it. ・False: Do not maintain it. When omitting it, it is assumed the one that false was specified.
Skip	-	Specify the number of lines deleted from the head of the

		Body data. Do not delete the line when you omit it.
NumConv	-	Specify whether to convert the numerical value in the Body data into the character string. ・True: Convert the numerical value. ・False: Do not convert the numerical value. When omitting it, it is assumed the one that false was specified.

2.3.2.3. @RegistTXT

Use a system variable identifier or optional Type of user variables "@RegistTXT" specifying it. In doing character string (BSTR) put, The TXT data can be registered to the resource in similar to "RegistTXT" command.

Specify the following options besides optional Type when the user variable is specified to add this variable to CaoController.

Table2-5Optional character string of @RegistTXT

Set item	Indispensability	Remarks
ResourcePath	○	Specify the resource passing.
Date	-	Specify the registration date given to the registration data. Adopt the request reception date when you omit it.
Retain	-	Specify whether to maintain this registration data on the MQTT broker side. ・True: Maintain it. ・False: Do not maintain it. When omitting it, it is assumed the one that false was specified.

2.3.2.4. @RegistBIN

Use a system variable identifier or optional Type when the user variable is specified "@RegistBIN" specifying it. In doing binary (ARRAY|UI1) in put CaoController::Execute("RegistBIN") The binary data can be newly registered to the resource in similar to "RegistBIN" commnad.

Specify the following options besides optional Type when the user variable is specified to add this variable to CaoController.

Table2-6Optional character string of @RegistBIN

Set item	Indispensability	Remarks
ResourcePath	○	Specify the resource passing.
MimeType	○	Specify MIME-TYPE specified for Content-Type.
Date	-	Specify the registration date given to the registration data. Adopt the request reception date when you omit it.
Retain	-	Specify whether to maintain this registration data on the MQTT broker side. ・True: Maintain it. ・False: Do not maintain it. When omitting it, it is assumed the one that false was specified.
Compression	-	Specify the compression type when it transmits by compressing the Body data as follows. ・gz Consider the Body data to be no compression when you omit it.

2.3.2.5. @ReferenceLatestData

Use a system variable identifier or optional Type when the user variable is specified "@ReferenceLatestData" specifying it. In doing get Similarly, the latest data of resource _ JSON can be acquired as character string (BSTR).

Specify the following options besides optional Type when the user variable is specified to add this variable to CaoController.

Table2-7Optional character string of @ReferenceLatestData

Set item	Indispensability	Remarks
ResourcePath	○	Specify the resource passing.
Extension	-	The following specification. ・json Treat assuming that json is specified when you omit it.
Select	-	Specify the $select parameter of QUERY. Return only the data of the field specified by < selection key >. ・< selection key > corresponds to the element name and the attribute name in name and XML in JSON, and can specify arbitrary key in the registration data. The hierarchy of the field is expressed by ".". ・Two or more < selection key > can be specified by switching off the "," district. ・_ date/_resource_path/_data that is the management data of this service cannot be used for < selection key >. Example) Ensor.id, sensor.name, and sensor.data.temp

2.3.2.6. @DataCount

Use a system variable identifier or optional Type when the user variable is specified "@DataCount" specifying it. In doing get "DataCount" _CaoController::Execute("DataCount")Similarly, number (I4) of HIT data of resource _ JSON can be acquired.

Specify the following options besides optional Type when the user variable is specified to add this variable to CaoController.

Table2-8Optional character string of @DataCount

Set item	Indispensability	Remarks
ResourcePath	○	Specify the resource passing (/$all can be used). Details of this option4.1.Refer to [wo].
Filter	-	Specify the $filter parameter of QUERY. Limit the result to return seeing though it agrees to < filter condition >. Two or more < filter condition > can be assumed, "Property name operator condition value", and be defined with and or. Describe the operator and the property name that can be used later. Details of this option4.2.Refer to [wo].

2.4. Variable list

The following system variables have been reserved in the IOTPLATFORM provider. Moreover, an arbitrary name can be used for user variables.

Variable identifier	Data type	Explanation	Attribute	
			get	put
@RegistJSON	VT_BSTR	Register the data of the JSON character string to resource _ JSON (accumulation).	-	-
@RegistCSV	VT_BSTR	Register the data of the CSV character string to resource _ JSON (accumulation).	-	-
@RegistTXT	VT_BSTR	Register the data of the TXT character string to resource _ JSON (accumulation).	-	-
@RegistBIN	VT_ARRAY| VT_UI1	Register the binary data to resource _ JSON (accumulation).	-	-
@ReferenceLatestData	VT_BSTR	Refer to the latest data of resource _ JSON.	-	-
@DataCount	VT_I4	Acquire the number of HIT data of resource _ JSON.	-	-

3. Command reference

3.1. Controller class

Table3-1CaoController::Execute command list

Type	Command	Function	Correspondence variable	Page
General purpose REST	REST	Give the URL passing and [kueri] for which the user is specified and execute PUT, GET, DELETE, and POST.	-	P20
Data registration/forwarding	RegistJSON	Register the data of the JSON character string to resource _ JSON (accumulation).	@RegistJSON	P22
	RegistCSV	Register the data of the CSV character string to resource _ JSON (accumulation).	@RegistCSV	P23
	RegistTXT	Register the data of the TXT character string to resource _ JSON (accumulation).	@RegistTXT	P24
	RegistBIN	Register the binary data to resource _ JSON (accumulation).	@RegistBIN	P25
Reference to resource data	ReferenceLatestData	Refer to the latest data of resource _ JSON.	@ReferenceLatestData	P26
	ReferencePastData	Refer to data (specified date) in the past of resource _ JSON.	-	P27
Retrieval of resource data	RetrieveData	Retrieve the data of resource _ JSON.	-	P28
	DataCount	Acquire the number of HIT data of resource _ JSON.	@DataCount	P29
Update of resource data	UpdateJSON	Update the JSON character string of resource _ JSON.	-	P30
	UpdateCSV	Update the CSV character string of resource _ JSON.	-	P31
	UpdateTXT	Update the TXT character string of resource _ JSON.	-	P32
	UpdateBIN	Update the binary data of resource _ JSON.	-	P33
Deletion of resource data	DeleteData	Delete the data of resource _ JSON.	-	P34

Type	Command	Function	Correspondence variable	Page
Registration of resource	RegistMetaData	Register the resource.	-	P34
Refer to the resource meta data.	ReferenceMetaData	Refer to the meta data of the resource.	-	P35
Update of meta data of resource	UpdateMetaData	Update the meta data of the resource.	-	P36
Resource deletion	DeleteMetaData	Delete the resource.	-	P36
Registration of access code	RegistAccessCodeData	Register the access code.	-	P37
Reference to access code	ReferenceAccessCodeData	Refer to the access code.	-	P38
	DataCountAccessCodeData	Acquire the number of HIT data of access codes.	-	P39
Update of access code	UpdateAccessCode	Update the access code.	-	P39
Deletion of access code	DeleteAccessCodeData	Delete the access code.	-	P40
Registration of event	RegistEventData	Register the event.	-	P40
Reference to event information	ReferenceEventData	Refer to event information.	-	P41
Update of event information	UpdateEventData	Update event information.	-	P42
Deletion of event	DeleteEventData	Delete the event.	-	P42

3.1.1. General purpose REST

3.1.1.1. CaoController::Execute("REST") command

Execute PUT, GET, DELETE, and POST generally for Fujitsu IOTPLATFORM Platform. Content of the response is interpreted as a character string of UTF-8.

Format REST (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< REST method > (BSTR) Requisite element Specify either the following. ・PUT ・GET ・DELETE ・POST
The second element	=	< passing added to URL that is basic > (BSTR) Requisite element Make URL at the request destination from passing specified by the parameter and this element specified at AddController. Made URL is as follows. < Base URI>/<API version >/< passing added to URL that becomes basic tenant ID>/< >
The third element	=	< [kueri] character string added to URL passing > (BSTR) Element that can be omitted (*1) Specify it without putting "?" on the head. Example) and "KeyA=ValueA&KeyB=ValueB"
The fourth element	=	<Content-Type>(BSTR) Element that can be omitted (*1) When omitting it, Content-Type is not specified.
The fifth element	=	< transmitted Body data > (ARRAY|UI1) Element that can be omitted (*1) Set Content transmitted by the binary.

Return value : [out] (ARRAY|VARIANT)

The first element	=	StatusCode(int)
The second element	=	Header(BSTR)
The third element	=	Body character string of response (BSTR)

(*1)It is considered that the element was omitted by specifying null or the dead letter character for a value. Moreover, when it omits since a certain element all, the array only of the specified element (Length is an array of three when specifying it from the head only by three elements while there are all of the five elements) can be passed to the argument. Only the element of not the array type but contents can be passed as an argument when becoming an array of one element as a result of omitting the element. When the array is taken in the argument of the Execute command from now on all, this is applied.

3.1.2. Data registration/forwarding to resource _ JSON

3.1.2.1. CaoController::Execute("RegistJSON") command

Registration/transmit the JSON data to the resource newly.

When an illegal argument is specified, E_INVALIDARG is generated.

Format RegistJSON (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< transmitted JSON character string > (BSTR) Requisite element
The third element	=	< registration date given to registration data > (BSTR) Element that can be omitted Adopt the request reception date when you omit it.
The fourth element	=	<RETAIN>(BOOL) Element that can be omitted Specify whether to maintain this registration data on the MQTT broker side. ・True: Maintain it. ・False: Do not maintain it. When omitting it, it is assumed the one that false was specified. - Disregard it even if RETAIN is specified when you specify Bulk Insert.
The fifth element	=	< Bulk Insert flag > (BSTR) Element that can be omitted Specify whether to execute Bulk Insert (Transmit two or more requests to one degree). ・Do not do none:Bulk Insert. ・Single_resource_path: Execute Bulk Insert for a single resource. When omitting it, it is assumed the one that none was specified.

3.1.2.2. CaoController::Execute("RegistCSV") command

Registration/transmit the CSV data to the resource newly.

When an illegal argument is specified, E_INVALIDARG is generated.

Format RegistCSV (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< transmitted CSV character string > (BSTR) Requisite element
The third element	=	< registration date given to registration data > (BSTR) Element that can be omitted Adopt the request reception date when you omit it.
The fourth element	=	<RETAIN>(BOOL) Element that can be omitted Specify whether to maintain this registration data on the MQTT broker side. ・True: Maintain it. ・False: Do not maintain it. When omitting it, it is assumed the one that false was specified.
The fifth element	=	< Body data deletion specification line > (I4) Element that can be omitted Specify the number of lines deleted from the head of the Body data. Do not delete the line when you omit it.
The sixth element	=	< numerical transformation > (BOOL) Element that can be omitted Specify whether to convert the numerical value in the Body data into the character string. ・True: Convert the numerical value. ・False: Do not convert the numerical value. When omitting it, it is assumed the one that true was specified.

3.1.2.3. CaoController::Execute("RegistTXT") command

Registration/transmit the TXT data to the resource newly.

When an illegal argument is specified, E_INVALIDARG is generated.

Format RegistTXT (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< transmitted TXT character string > (BSTR) Requisite element
The third element	=	< registration date given to registration data > (BSTR) Element that can be omitted Adopt the request reception date when you omit it.
The fourth element	=	<RETAIN>(BOOL) Element that can be omitted Specify whether to maintain this registration data on the MQTT broker side. ・True: Maintain it. ・False: Do not maintain it. When omitting it, it is assumed the one that false was specified.

3.1.2.4. CaoController::Execute("RegistBIN") command

Registration/transmit the binary data to the resource newly.

When an illegal argument is specified, E_INVALIDARG is generated.

Format RegistBIN (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	It is specified MIME-TYPE>(BSTR) in < Content-Type. Requisite element
The third element	=	< transmitted binary > (ARRAY|UI1) Requisite element
The fourth element	=	< registration date given to registration data > (BSTR) Element that can be omitted Adopt the request reception date when you omit it.
The fifth element	=	<RETAIN>(BOOL) Element that can be omitted Specify whether to maintain this registration data on the MQTT broker side. ・True: Maintain it. ・False: Do not maintain it. When omitting it, it is assumed the one that false was specified.
The sixth element	=	< compression type > (BSTR) Element that can be omitted Specify the compression type when it transmits by compressing the Body data as follows. ・gz Consider the Body data to be no compression when you omit it.

3.1.3. Reference to resource data

3.1.3.1. CaoController::Execute("ReferenceLatestData") command

Refer to the latest data registered in the resource.

When an illegal argument is specified, E_INVALIDARG is generated.

Format ReferenceLatestData (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< extension > (BSTR) Element that can be omitted The following specification. ・json Treat assuming that json is specified when you omit it.
The third element	=	$select parameter < > (BSTR) of QUERY Element that can be omitted Return only the data of the field specified by < selection key >. ・< selection key > corresponds to the element name and the attribute name in name and XML in JSON, and can specify arbitrary key in the registration data. The hierarchy of the field is expressed by ".". ・Two or more < selection key > can be specified by switching off the "," district. ・_ date/_resource_path/_data that is the management data of this service cannot be used for < selection key >.

Return value : [out] (BSTR)
Referred data.

3.1.3.2. CaoController::Execute("ReferencePastData") command

Refer to the data registered in the resource as the registration date was corresponding to a specified date.

When an illegal argument is specified, E_INVALIDARG is generated.

(*1) When two or more data of < registration date > exists, all data is returned.

Format ReferencePastData (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< registration date > (BSTR) Requisite element At the registration date of data for the reference
The third element	=	< extension > (BSTR) Element that can be omitted The following specification. ・json Treat assuming that json is specified when you omit it.
The fourth element	=	Select condition < > (BSTR) Element that can be omitted Return only the data of the field specified by < selection key >. ・< selection key > corresponds to the element name and the attribute name in name and XML in JSON, and can specify arbitrary key in the registration data. The hierarchy of the field is expressed by ".". ・Two or more < selection key > can be specified by switching off the "," district. ・_ date/_resource_path/_data that is the management data of this service cannot be used for < selection key >.

Return value : [out] (BSTR)
Referred data.

3.1.4. Retrieval of resource data

3.1.4.1. CaoController::Execute("RetrieveData") command

Retrieve the data registered in a corresponding resource to a specified condition.

When an illegal argument is specified, E_INVALIDARG is generated.

(*1) Sort the order of the data obtained as a retrieval result by < resource passing > and < registration date >.

Format RetrieveData (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing (/$all can be used) > (BSTR) Requisite element Details4.1.Refer to [wo].
The second element	=	< extension > (BSTR) Element that can be omitted The following specification. ・json Treat assuming that json is specified when you omit it.
The third element	=	< filter condition > (BSTR) Element that can be omitted Limit the result to return seeing though it agrees to < filter condition >. Details4.2.Refer to [wo].
The fourth element	=	Top condition < > (I4) Element that can be omitted Limit the data obtained as a retrieval result to n. The maximum acquisition number is 1000.
The fifth element	=	Skip condition < > (I4) Element that can be omitted Skip does n acquired data as a retrieval result. The order of the data obtained as a retrieval result is > thesecond sorting key:< the first sorting key:< resource passing Noboru. Sort it by [**] date >.
The sixth	=	Select condition < > (BSTR)

element Element that can be omitted
Return only the data of the field specified by < selection key >.
・< selection key > corresponds to the element name and the attribute name in name and XML in JSON, and can specify arbitrary key in the registration data. The hierarchy of the field is expressed by ".".
・Two or more < selection key > can be specified by switching off the "," district.
・_ date/_resource_path/_data that is the management data of this service cannot be used for < selection key >.

Return value : [out] (BSTR)
Corresponding resource data.

3.1.4.2. CaoController::Execute("DataCount") command

Acquire the data number registered in a corresponding resource to a specified condition.
When an illegal argument is specified, E_INVALIDARG is generated.
(*1) Sort the order of the data obtained as a retrieval result by < resource passing > and < registration date >.

Format DataCount (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element = < resource passing (/$all can be used) > (BSTR)
Requisite element
Details4.1.Refer to [wo].

The second element = < filter condition > (BSTR)
Element that can be omitted
Limit the result to return seeing though it agrees to < filter condition >.
Details4.2.Refer to [wo].

Return value : [out] (I4)
Corresponding number of resource data.

3.1.5. Update of resource data

3.1.5.1. CaoController::Execute("UpdateJSON") command

Update the data registered in a corresponding resource to a specified condition in the JSON character-string data.

When an illegal argument is specified, E_INVALIDARG is generated.

(*1) When two or more data of the same registration date exists, only one is updated. (To which data irregular.)

Format UpdateJSON (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< registration date of data for update > (BSTR) Requisite element
The third element	=	< transmitted JSON character string > (BSTR) Requisite element
The fourth element	=	< registration date after it updates it > (BOOL) Element that can be omitted ・Do not update the registration date when you omit it. ・Do not check, and overwrite whether for the data of a specified registration date to already already exist.

3.1.5.2. CaoController::Execute("UpdateCSV") command

Update the data registered in a corresponding resource to a specified condition in the CSV character-string data.

When an illegal argument is specified, E_INVALIDARG is generated.

(*1) When two or more data of the same registration date exists, only one is updated. (To which data irregular.)

Format UpdateCSV (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< registration date of data for update > (BSTR) Requisite element
The third element	=	< transmitted CSV character string > (BSTR) Requisite element
The fourth element	=	< registration date after it updates it > (BOOL) Element that can be omitted ・Do not update the registration date when you omit it. ・Do not check, and overwrite whether for the data of a specified registration date to already already exist.
The fifth element	=	< Body data deletion specification line > (I4) Element that can be omitted Specify the number of lines deleted from the head of the Body data. Do not delete the line when you omit it.
The sixth element	=	< numerical transformation > (BOOL) Element that can be omitted Specify whether to convert the numerical value in the Body data into the character string. ・True: Convert the numerical value. ・False: Do not convert the numerical value. When omitting it, it is assumed the one that false was specified.

3.1.5.3. CaoController::Execute("UpdateTXT") command

Update the data registered in a corresponding resource to a specified condition in the TXT character-string data.

When an illegal argument is specified, E_INVALIDARG is generated.

(*1) When two or more data of the same registration date exists, only one is updated. (To which data irregular.)

Format UpdateTXT (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< registration date of data for update > (BSTR) Requisite element
The third element	=	< transmitted TXT character string > (BSTR) Requisite element
The fourth element	=	< registration date after it updates it > (BOOL) Element that can be omitted ・Do not update the registration date when you omit it. ・Do not check, and overwrite whether for the data of a specified registration date to already already exist.

3.1.5.4. CaoController::Execute("UpdateBIN") command

Update the data registered in a corresponding resource to a specified condition in the binary data.

When an illegal argument is specified, E_INVALIDARG is generated.

(*1) When two or more data of the same registration date exists, only one is updated. (To which data irregular.)

Format UpdateBIN (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< registration date of data for update > (BSTR) Requisite element
The third element		It is specified MIME-TYPE>(BSTR) in < Content-Type. Requisite element
The fourth element	=	< transmitted binary > (ARRAY|UI1) Requisite element
The fifth element	=	< registration date after it updates it > (BOOL) Element that can be omitted ・Do not update the registration date when you omit it. ・Do not check, and overwrite whether for the data of a specified registration date to already already exist.
The sixth element	=	< compression type > (BSTR) Element that can be omitted Specify the compression type when it transmits by compressing the Body data as follows. Consider the Body data to be no compression when you omit it. ・gz

3.1.6. Deletion of resource data

3.1.6.1. CaoController::Execute("DeleteData") command

Delete the data registered in a corresponding resource to a specified condition.

When an illegal argument is specified, E_INVALIDARG is generated.

Format DeleteData (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< filter condition > (BSTR) Requisite element Limit the result to return seeing though it agrees to < filter condition >. Details4.2.Refer to [wo].

3.1.7. Registration of resource

3.1.7.1. CaoController::Execute("RegistMetadata") command

Register a new resource.

When an illegal argument is specified, E_INVALIDARG is generated.

Format RegistMetadata (<Data>)

<Data> : [in] (ARRAY | BSTR)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< transmitted JSON character string > (BSTR) Element that can be omitted Details4.3.1Refer to [wo]. Register only the resource when you omit it.

3.1.8. Refer to the meta data of the resource.

3.1.8.1. CaoController::Execute("ReferenceMetadata") command

Refer to all the meta data registered in a corresponding resource to a specified condition.

When an illegal argument is specified, E_INVALIDARG is generated.

Format ReferenceMetadata (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< resource passing (/$all) > (BSTR) Requisite element Details4.1.Refer to [wo].
The second element	=	Top condition < > (I4) Element that can be omitted Limit the data obtained as a retrieval result to n. The maximum acquisition number is 1000.
The third element	=	Skip condition < > (I4) Element that can be omitted Skip does n acquired data as a retrieval result. The order of the data obtained as a retrieval result is > thesecond sorting key:< the first sorting key:< resource passing Noboru. Sort it by [**] date >.

Return value : [out] (BSTR)

Referred resource data.

3.1.9. Update of meta data of resource

3.1.9.1. CaoController::Execute("UpdateMetadata") command

Overwrite all the meta data of the object resource for the resource in the update data in the JSON form.

When an illegal argument is specified, E_INVALIDARG is generated.

Format UpdateMetadata (<Data>)

<Data> : [in] (ARRAY|BSTR)

The first element	=	< resource passing > (BSTR) Requisite element
The second element	=	< transmitted JSON character string > (BSTR) Requisite element Details4.3.1Refer to [wo].

3.1.10. Deletion of resource

3.1.10.1. **CaoController::Execute("DeleteMetadata") command**

Delete the specified resource.

When an illegal argument is specified, E_INVALIDARG is generated.

Format DeleteMetadata (<Data>)

<Data> : [in](BSTR)
< resource passing > Requisite element

3.1.11. Registration of access code

3.1.11.1. CaoController::Execute("RegistAccessCodeData") command

Register the access code to the resource newly.

When an illegal argument is specified, E_INVALIDARG is generated.

Format RegistAccessCodeData (<Data>)

<Data> : [in] (ARRAY |BSTR)

The first element	=	< access code > (BSTR) Requisite element Registered access code name
The second element	=	< transmitted JSON character string > (BSTR) Requisite element Details4.3.2Refer to [wo].

3.1.12. Reference to access code

3.1.12.1. CaoController::Execute("ReferenceAccessCodeData") command

Refer to all the access codes registered in a corresponding resource to a specified condition.

When an illegal argument is specified, E_INVALIDARG is generated.

Format ReferenceAccessCodeData (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< access code > (BSTR) Requisite element
The second element	=	< filter condition > (BSTR) Element that can be omitted Limit the result to return seeing though it agrees to < filter condition >. Details4.2.Refer to [wo].
The third element	=	Top condition < > (I4) Element that can be omitted Limit the data obtained as a retrieval result to n. The maximum acquisition number is 1000.
The fourth element	=	Skip condition < > (I4) Element that can be omitted Skip does n acquired data as a retrieval result. The order of the data obtained as a retrieval result is > thesecond sorting key:< the first sorting key:< resource passing Noboru. Sort it by [**] date >.

Return value : [out] (BSTR)
Corresponding access code information.

3.1.12.2. CaoController::Execute("DataCountAccessCodeData") command

Acquire the number of access codes registered in a corresponding resource to a specified condition.

When an illegal argument is specified, E_INVALIDARG is generated.

Format DataCountAccessCodeData (<Data>)

<Data> : [in] (ARRAY|BSTR)

The first element	=	< access code > (BSTR) Requisite element
The second element	=	< filter condition > (BSTR) Element that can be omitted Limit the result to return seeing though it agrees to < filter condition >. Details4.2.Refer to [wo].

Return value : [out] (I4)

Corresponding number of access codes.

3.1.13. Update of access code

3.1.13.1. CaoController::Execute("UpdateAccessCodeData") command

Update the access code.

When an illegal argument is specified, E_INVALIDARG is generated.

Format UpdateAccessCodeData (<Data>)

<Data> : [in] (ARRAY| BSTR)

The first element	=	< access code > (BSTR) Requisite element.
The second element	=	< transmitted JSON character string > (BSTR) Requisite element Details4.3.2Refer to [wo].

3.1.14. Deletion of access code

3.1.14.1. CaoController::Execute("DeleteAccessCodeData") command

Delete the specified access code.

When an illegal argument is specified, E_INVALIDARG is generated.

Format DeleteAccessCodeData (Data)

<Data> : [in] (BSTR)
< access code > (BSTR)
Requisite element

3.1.15. Registration of event

3.1.15.1. **CaoController::Execute("RegistEventData") command**

New event information can be registered to the specified resource data.

When an illegal argument is specified, E_INVALIDARG is generated.

Format RegistEventData (<Data>)

<Data> : [in] (BSTR)
< transmitted JSON character string >
Requisite element
Details4.3.3 を参照してください

Return value : [out] (BSTR)
Event ID of registered event information.

3.1.16. Reference to event information

3.1.16.1. CaoController::Execute("ReferenceEventData") command

Refer to event information.

When an illegal argument is specified, E_INVALIDARG is generated.

Format ReferenceEventData (<Data>)

<Data> : [in] (ARRAY|VARIANT)

The first element	=	< event ID>(BSTR) Element that can be omitted - Specify null when you want to refer to event information on the condition of the 2-4th element.
The second element	=	< filter condition > (BSTR) Element that can be omitted Limit the result to return seeing though it agrees to < filter condition >. Details4.2.Refer to [wo].
The third element	=	Top condition < > (I4) Element that can be omitted Limit the data obtained as a retrieval result to n. The maximum acquisition number is 1000.
The fourth element	=	Skip condition < > (I4) Element that can be omitted Skip does n acquired data as a retrieval result. The order of the data obtained as a retrieval result is > thesecond sorting key:< the first sorting key:< resource passing Noboru. Sort it by [**] date >.

Return value : [out] (BSTR)
Corresponding event information.

3.1.17. Update of event information

3.1.17.1. CaoController::Execute("UpdateEventData") command

Update event information on specified event ID.

When an illegal argument is specified, E_INVALIDARG is generated.

Format UpdateEventData(<Data>)

<Data> : [in] (ARRAY| BSTR)

The first element	=	< event ID>(BSTR) Requisite element. Event ID to be updated
The second element	=	< transmitted JSON character string > Requisite element Details4.3.3 を参照してください

3.1.18. Deletion of event

3.1.18.1. CaoController::Execute("DeleteEventData") command

Delete event information on specified event ID.

When an illegal argument is specified, E_INVALIDARG is generated.

Format DeleteEventData (Data)

<Data> : [in] (BSTR)

< event ID >

Requisite element

4. Supplementation

4.1. Description method of < resource passing (/$all can be used)>.

- Specify the resource passing in the full path.
  Return the resource data of the specified resource passing.

- Specify on the way of the resource passing, and add "/$all" to the furnace afterwards.
  Return the resource data of the specified resource passing of all the passing subordinates.
  Example)
  Target two resources of "A/B" and "A/B/C" when you specify "A/$all" with three resources of "AX" "A/B" "A/B/C" exist.

4.2. About the filter condition

4.2.1. Operator of filter condition

Operator	Explanation	Example
eq	Equal sign	Owner eq 'Tom'
ne	Sign of inequality	Owner ne null
gt	It is larger.	Floor1.Value gt 1000
ge	This concludes my report.	Floor1.Value ge 1000
lt	It is smaller.	Floor1.Value lt 1000
le	As follows	Floor1.Value le 1000
and	Logical product	Floor1.Value ge 1000 and Owner eq 'Tom'
or	Logical add	Id eq 2 or Id eq 1

1. Null shows that the value doesn't exist.
2. When name of data is a layered structure, it expresses it by < name>.<name >.
3. (A eq 1 and B eq 1)The expression like or(A eq 2 and B eq 2) is also possible.
   - However, () cannot be defined while it is () in this case. For instance, it makes an error of ((A eq 1 and B eq 1)or(A eq 2 and B eq 2)) and(C eq 1).
4. It doesn't correspond to the match by the regular expression.
5. Enclose the character string with a single quart. The one not enclosed with a single quart is considered to be a numerical value.

6. When the content of the JSON array is specified, it expresses it by < name>. < array index >. Array index is a figure.

   Example:

Data for retrieval	{
	"Owners":[
	"Taro",
	"Jiro"
	]
	}

Example of specifying filter condition	Owners.0 eq 'Taro'

- When there is an object structure in the array and name is composed only of the figure, specification and array index name might not be able to be identified. Specify both array index and name referring to the example of the following.

  Example of data for retrieval:

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

  Retrieval example:

Filter specification condition	Suited data
data.0 eq 'Taro'	①, ②, ③
data.1.0 eq 'Taro'	②

4.2.2. Property name that can be used for filter condition

Property name	Explanation	Remarks
_date	At the registration date	At the registration date of data for the retrieval - _About the registration date when the condition is specified with date Do not enclose it by using a single quart.
Arbitrary name	Arbitrary name included in registration data	Non-reservation character of URI ("Alphanumeric character" Excluding "-", ".", "_", and "~".)[Ha] Encode percent.

(*1) It follows ISO8601 (The millisecond expression as a basic mark is used)(20141225T103612.001Z etc.). Accuracy is a millisecond (When the millisecond is omitted, it is considered 0 milliseconds). Affix

"Z" to the delimitation of the second and the millisecond when you omit "." and the zone specification of time by the "±hhmm" form. Use UTC when this service stores it in the response.

Example)
To the resource passing that has the following JSON structure
When you call the part that enclosed it with a red frame the 1~3th element on.


```
[
{
"_data": {
"Key3": "Value3"
},
"_date": "20180730T044740.351Z",
"_resource_path": "Test/ABC"
},
{
"_data": {
"Key": "Value"
},
"_date": "20180730T044607.267Z",
"_resource_path": "Test/ABC"
},
{
"_data": {
"Key": "Value"
},
"_date": "20180730T044549.801Z",
"_resource_path": "Test/ABC"
}
]
```


・Specify the following for the filter condition when you acquire only information on the first element.
"_date eq _date eq 20180730T044740.351Z" Or,「Key3 eq 'Value3'」

4.2.3. About the filter condition that can be used by referring to the access code reference and event information

Filter conditional operator

Operator	Explanation	Example
eq	Equal sign	Owner eq 'Tom'

Property name of filter condition

Property name	Explanation	Remarks
_resource_path	Resource passing	Percent encode including "/" is unnecessary.

Support function of filter condition

Function	Explanation	Example
bool startwith(string p0,string p1)	Agreement forward	/abc* ?$filter=startswith(_resource_path,'hoge') eq true

1. The false specification is uncorrespondence.
2. When specified _ resource_path eq hoge right under the $filter condition, it becomes complete specification.
   When specified by using startswith(), it becomes agreement forward specification.

(As of July, 2018.)

4.3. JSON format at REST

This format is made based on following URL API references.

https://iot-docs.jp-east-1.paas.cloud.global.fujitsu.com/ja/manual/v5/apireference.pdf

4.3.1. When resource is controlled

Parameters	Form	M/O	Explanation	Maximum length
resource	-	M	Start tag	-
retention_period	string	O(*2)	Make the preservation period a day when preservation period (Sunday) of the resource data and it is not set. - The range of the value is 1-9999.	9999
fwd_info		O(*3)	Forwarding site information	-
http	-	M	Forwarding information on HTTP	-
method	-	M	Either "GET", "POST", "PUT", "DELETE", "HEAD", "OPTIONS" or "TRACE"	Seven characters
uri	string	M	URI "Http://-" or "Https://-"	256 characters
basic_auth_id	string	O	ID for Basic attestation	20 characters
basic_auth_pass	string	O	Password for Basic attestation	20 characters
header_fields	-(array)	O	HTTP header (The number of array elements is 10 or less).	-
field_name	string	M	Header field name ":Do not contain "	20 characters
field_value	string	M	Value stored in the above-mentioned header field	512 characters

(*1) M: Indispensability and O: Option. M/O of each child element shows no necessary when the parents element is set.

(*2) It is effective only for resource _ JSON and resource _ Binary.

(*3) When the object resource is only a forwarding resource of the JSON form, it is possible to set it.

The input example

https://iot-docs.jp-east-1.paas.cloud.global.fujitsu.com/ja/manual/v5/userguide.pdf

Chapter 11 Refer to the example of registering 11.1 resources.

4.3.2. When access code is controlled

Parameters	Form	M/O	Explanation	Maximum length
access_code	-	M	Start tag	-
permissions	-	M	Authority information	-
ip_filter	String (array)	O	Specify Internet Protocol address range information on control of access (access code) setting defined by the service portal. Return error response (400 Bad Request) when you set undefined Internet Protocol address range information. - Two Internet Protocol addresses can be specified in the form of "Beginning Internet Protocol address" and "End Internet Protocol address". - Make beginning Internet Protocol address and end Internet Protocol address the same address when you set one Internet Protocol address. - The setting that becomes "Beginning Internet Protocol address > end Internet Protocol address (When comparing it considering Internet Protocol address to be 32bit)" should be able to be done.	35 characters
resource_operations	-(array)	M	It is authority information, that is, the string in the resource.	
resource_path	string	M	Resource passing	128 Character
operations	String (array)	M	Either "Hierarchy_get", "Hierarchy_put", "Create", "Read", "Update", "Delete" or "List". Plurals can be specified by the array.	72 characters
certification_info	-	O	Client information	
certification	string	M	Include client certificate, PEM form "-----BEGIN CERTIFICATE-----", and "-----END CERTIFICATE-----". Moreover, register the line feed code as "¥n".	10000 characters
certificate_usage	string	M	Usage of client certificate. Specify the following. - auth: Use it by the client authorization.	4 Character

	protocols	String (array)	O	Either protocol specification information "Http", "Https", "Mqtt" or "Mqtts". Plurals can be specified by the array.	29 characters

(*1) M: Indispensability and O: Option. M/O of each child element shows no necessary when the parents element is set.

The input example

https://iot-docs.jp-east-1.paas.cloud.global.fujitsu.com/ja/manual/v5/userguide.pdf

Chapter 12 Refer to the example of registering 12.1 access code.

4.3.3. When event is controlled

Parameters	Form	M/O	Explanation	Maximum length
event	-	M	Start tag	-
conditions	-	M	Event condition	-
targets	-	M	Object	-
resource_path	string	M	At the operation to the resource passing binary data of the object, Object..resource..pass..setting.	128 characters
operations	String (array)	M	Specify "Create" and "Update" for a resource that operates of the resource data of the object (Differ from the right of access of the access code) and is usual. - Specify "Create" for resource _ Binary.	33 characters
read_access_code	string	M	Access code with read right or hierarchy_get right of object resource passing	48 characters
notification_condition	-	O	Notification condition	-
start_time	Date or time	O(*2)	Beginning date or time (*3)	20 characters
end_time	Date or time	O(*4)	End date or time (*3)	20 characters
body_conditions	-	O	It is not possible to set it to data body condition Binary resource.	-
path_type	string	M	Only the form "JSONPath" of path can be specified.	Eight characters
path	string	M	Passing to specify body element	1902 characters
comparing_operator	string	M	Either relative operator "Eq" (=), "Ne" (≠), "Gt" (>), "Ge" (≧), "Lt" (<), "Le" (≦) or "Substring_of" (partial agreement). (*5)	12 characters

				value		String or numerical value		Value of object of comparison.	— (*7)
			awake_condition			-	O	Event control release condition. It is not possible to set it to the Binary resource. - It enters the state of the event control after notification_condition is executed notifying the filling event if this setting is done. Set the condition to release the event control here.	-
				body_conditions		-	M	It is set [fu] for data body condition Binary resource.	
					path_type	string	M	Only the form "JSONPath" of path can be specified.	Eight characters
					path	string	M	Passing to specify body element	1902 characters
					comparing_operator	string	M	Either relative operator "Eq" (=), "Ne" (≠), "Gt" (>), "Ge" (≧), "Lt" (<), "Le" (≦) or "Substring_of" (partial agreement). (*5)	12 characters
					value	String or numerical value		Value of object of comparison.	— (*7)
	notification					—(*6)	M	Content of notification	-
		http				-	O	HTTP notification setting	-
			method			string	M	Either "GET", "POST", "PUT", "DELETE", "HEAD", "OPTIONS" or "TRACE".	Seven characters
			uri			string	M	URI "Http://-" or "Https://-"	256 characters
			basic_auth_id			string	O	ID for Basic attestation	20 characters
			basic_auth_pass			string	O	Password for Basic attestation	20 charact

							ers
			header_fields	-(array)	O	HTTP header (The number of array elements is 10 or less).	-
				field_name	string	M	Header field name ":Do not contain "	20 characters
				field_value	string	M	Value stored in the above-mentioned header field	512 characters
			body	string	O	Main body of resource data, event ID event generation date, and object resource passing that became trigger of event when value is omitted storing it in body	1024 characters
		smtp		-	O	SMTP notification setting	-
			send_to	string	M	E-mail Address at notification destination	256 characters
			subject	string	O	Subject	256 characters
			body	string	M	Text	140 characters

(*1) M: Indispensability and O: Option. M/O of each child element shows no necessary when the parents element is set.

(*2) Only one of start_time and end_time cannot be set. Moreover, a set both value is a date or time.
It carves and it is necessary to arrange it to either [no].

(*3) Follow ISO8601 at the specification at the date (20141225T103612Z etc.). Specify accuracy until the second.
It is possible. It becomes it if information on the date was cut down from ISO8601 at the specification at time (T103612Z etc.). Accuracy.
It is up to a second in similar. peel offAffix "Z" to the zone specification of time when you omit it by the "±hhmm" form.

(*4) Only one of start_time and end_time cannot be specified. Moreover, a set both value is a date or time.
It carves and it is necessary to arrange it to either [no].

(*5) When value is a numerical value, eq, ne, gt, ge, lt, and le can be specified. Eq and ne when value is character string

Substring_of can be specified. In the string comparison, do the case sensitivity.

(*6) A setting that http or is either of smtp is necessary under the control of notification.

(*7) - in case of the character string and 1-128 character integer2,147,483,648 They are 10 digits, and 5 ..integer part.. ..fraction part.. digits for 2,147,483,647- real numbers.

The input example

https://iot-docs.jp-east-1.paas.cloud.global.fujitsu.com/ja/manual/v5/userguide.pdf

Chapter 13 Refer to the example of registering 13.1 events.

4.4. Response status code at REST

It introduces a part of the status code returned when the command is executed.

Status-Code	Reason-Phrase	Explanation
200	OK	Success and success in resource data creation
201	Created	Success in resource, access code, and event making
204	No Content	Deletion success of resource in either the following when it succeeds deletion, meta data access code event is referred to, and various information doesn't exist (It is likely to be going to change in the future) and the meta data access code event is deleted when no existence of the corresponding resource data and the resource are deleted when it corresponds and the resource data is referred though it exists.
206	Partial content	Success in partial acquisition
400	Bad Request	There is an illegal value in the request data.
401	Unauthorized	There is no right of access to the resource.
403	Forbidden	There is no right of access.
404	Not Found	The resource doesn't exist.
405	Method Not Allowed	The method type of the correspondence has not been permitted.
408	Request Time-out	It is a request time-out.
409	Conflict	It is rival with other resources.
411	Length Required	The server access was refused (There is no Content-Lengh specification).
412	Precondition Failed	The server access was refused (The request condition is illegal).
413	Payload Too Large	The server access was refused (The size of the request body transcends the tolerance of the server).
414	URI Too Long	The server access was refused (URI is long).
415	Unsupported Media Type	The server access was refused (unsupport Content-Type).
416	Requested Range Not Satisfiable	The server access was refused (The value of the Range demand is illegal).
421	Misdirected Request	The response was transmitted to a server not generable.
423	Locked	The resource is locked.

429	Too Many Request	The traffic upper bound in the contract is exceeded.
495	SSL Certificate Error	An invalid client certificate was received.
496	SSL Certificate Required	The client certificate was not sent from the client.
497	HTTP Request Sent toHTTPS Port	The HTTP request was received in the HTTPS request port.
500	Internal Server Error	It is a failure by the problem on the server side.
501	Not Implemented	The unsupport request method was transmitted with the server.
502	Bad Gateway	The gateway server doesn't start.
503	Service Unavailable	It is not possible to access it temporarily.
504	Gateway Time-out	The gateway server was not able to return the response in time.

Refer to following URL for more detailed information.

https://iot-docs.jp-east-1.paas.cloud.global.fujitsu.com/ja/manual/v5/apireference.pdf

5. Sample program

The sample ..C as follows.. is shown.

5.1. Setting and acquisition of value to Variable

Variables list The usage example of Variable to which [nite] is introduced is shown.

List 5-1

```
Abbreviation.
using ORiN2.ManagedCAO;

namespace IOTPLATFORM_Sample
{
 public class Sample
 {
          private readonly string BASE_URL = @"http://<zone>.fujitsu.com";
          private readonly string TENANT_ID = "TenantID";
          private readonly string ACCESS_CODE = "AccessCode";
          private readonly string RESOURCE_OPT = "ResourcePath=Test";
          private readonly string FILE_PATH = "jpegFilePath";


   private CCaoEngine m_CaoEngine = null;
   private CCaoWorkspaces m_CaoWorkspaces = null;
   private CCaoWorkspace m_CaoWorkspace = null;
   private CCaoController m_CaoController = null;

          private void executeAllVariable()
          {
               // CAO engine generation
 m_CaoEngine = new CCaoEngine();
 m_CaoWorkspaces = m_CaoEngine.Workspaces;
 m_CaoWorkspace = m_CaoWorkspaces[0];

// Controller argument
 var opstionStr = string.Format("BaseURL={0},TenantID={1},AccessCode={2}", BASE_URL, TENANT_ID,
ACCESS_CODE);

// Connect it with the controller.
 m_CaoController = m_CaoWorkspace.AddController("Test", "CaoProv.FUJITSU.IOTPLATFORM", null,
opstionStr);

// Optional character string
```

```
 var registBinOpt = string.Format("{0}, MimeType=image/jpeg, Retain=true", RESOURCE_OPT);
 var dataCountOpt = "ResourcePath=Test/$all";

 // @RegistJSON
 CCaoVariable systemVariable = m_CaoController.AddVariable("@RegistJSON", RESOURCE_OPT);
 CCaoVariable userVariable = m_CaoController.AddVariable("RegistJSON",
string.Format("Type=@RegistJSON,{0}", RESOURCE_OPT));

 string jsonStr = "{¥"test¥":1}";
 systemVariable.Value = jsonStr;
 userVariable.Value = jsonStr;

 // @RegistCSV
 systemVariable = m_CaoController.AddVariable("@RegistCSV", RESOURCE_OPT);
 userVariable = m_CaoController.AddVariable("RegistCSV", string.Format("Type=@RegistCSV, {0}",
RESOURCE_OPT));
 string csvStr = @"name,age,tel
Tanaka and 30,012-345-6789
"of ..Suzuki.. 40,098-765-4321;

 systemVariable.Value = csvStr;
 userVariable.Value = csvStr;

 // @RegistTXT
 systemVariable = m_CaoController.AddVariable("@RegistTXT", RESOURCE_OPT);
 userVariable = m_CaoController.AddVariable("RegistTXT", string.Format("Type=@RegistTXT,{0}",
RESOURCE_OPT));

 string txtStr = "test";
 systemVariable.Value = txtStr;
 userVariable.Value = csvStr;

 systemVariable = m_CaoController.AddVariable("@RegistBIN", registBinOpt);
 userVariable = m_CaoController.AddVariable("RegistBIN", string.Format("Type=@RegistBIN,{0}",
registBinOpt));

 if (!string.IsNullOrEmpty(FILE_PATH))
```

```
        {
        byte[] binaryArray = File.ReadAllBytes(FILE_PATH);
        systemVariable.Value = binaryArray;
        userVariable.Value = binaryArray;
        }

        // @ReferenceLatestData
        systemVariable = m_CaoController.AddVariable("@ReferenceLatestData", RESOURCE_OPT);
        userVariable = m_CaoController.AddVariable("ReferenceLatestData",
      string.Format("Type=@ReferenceLatestData,{0}", RESOURCE_OPT));

        var referenceLatestDataStr = systemVariable.Value.ToString();
        referenceLatestDataStr = userVariable.Value.ToString();

        // @DataCount
        systemVariable = m_CaoController.AddVariable("@DataCount", dataCountOpt);
        userVariable = m_CaoController.AddVariable("DataCount", string.Format("Type=@DataCount,{0}",
      dataCountOpt));

        var datacount = (int)systemVariable.Value;
        datacount = (int)userVariable.Value;
                  }
         }
        }
```

5.2. General purpose REST

汎用 REST[Tsuka] example is shown drinking.

List 5-2

```
Abbreviation.
using System.Collections.Generic;
using ORiN2.ManagedCAO;

namespace IOTPLATFORM_Sample
{
 public class Sample
  {

private readonly string BASE_URL = @"http://<zone>.fujitsu.com";
private readonly string TENANT_ID = "TenantID";
private readonly string ACCESS_CODE = "AccessCode";
private readonly string RESOUCE_PATH = "Test";


private CCaoEngine m_CaoEngine = null;
private CCaoWorkspaces m_CaoWorkspaces = null;
private CCaoWorkspace m_CaoWorkspace = null;
private CCaoController m_CaoController = null;

        private void executSample()
{
          // CAO engine generation
m_CaoEngine = new CCaoEngine();
m_CaoWorkspaces = m_CaoEngine.Workspaces;
m_CaoWorkspace = m_CaoWorkspaces[0];


// Controller argument
 var opstionStr = string.Format("BaseURL={0},TenantID={1},AccessCode={2}", BASE_URL, TENANT_ID,
ACCESS_CODE);


 if (m_CaoController == null)
 {
// Connect it with the controller.
 m_CaoController = m_CaoWorkspace.AddController("Test", "CaoProv.FUJITSU.IOTPLATFORM", null,
opstionStr);
 }
 var paramList = new List<object>();
```

```
        // General purpose REST
         paramList.Add("PUT");
         paramList.Add(RESOUCE_PATH);
         var restResult = m_CaoController.Execute("REST", paramList.ToArray());
}
           }
         }
```

5.3. Data control

リソース JSON へのデータ登録/転送,リソースデータの参照,リソースデータの検索,リソースデータの更新,リソースデータの削除[Tsuka] example is shown drinking.

List 5-3

```
Abbreviation.
using System.IO;
using System.Collections.Generic;
using ORiN2.ManagedCAO;

namespace IOTPLATFORM_Sample
{
 public class Sample
  {

private readonly string BASE_URL = @"http://<zone>.fujitsu.com";
private readonly string TENANT_ID = "TenantID";
private readonly string ACCESS_CODE = "AccessCode";


private readonly string RESOURCE_OPT = "ResourcePath=Test";
private readonly string FILE_PATH = "jpegFilePath";
private readonly string RESOUCE_PATH = "Test";
private readonly string REGIST_DATE = "20180831T023219.222Z";

private CCaoEngine m_CaoEngine = null;
private CCaoWorkspaces m_CaoWorkspaces = null;
private CCaoWorkspace m_CaoWorkspace = null;
private CCaoController m_CaoController = null;

        private void executSample()
{
          // CAO engine generation
m_CaoEngine = new CCaoEngine();
m_CaoWorkspaces = m_CaoEngine.Workspaces;
m_CaoWorkspace = m_CaoWorkspaces[0];


// Controller argument
 var opstionStr = string.Format("BaseURL={0},TenantID={1},AccessCode={2}", BASE_URL, TENANT_ID,
ACCESS_CODE);


 if (m_CaoController == null)
 {
// Connect it with the controller.
```

```
 m_CaoController = m_CaoWorkspace.AddController("Test", "CaoProv.FUJITSU.IOTPLATFORM", null,
opstionStr);
 }
// Optional character string
 var registBinOpt = string.Format("{0}, MimeType=image/jpeg, Retain=true", RESOURCE_OPT);
 var paramList = new List<object>();
 var paramStrList = new List<string>();

// Data control
// Data registration/forwarding to resource JSON
 // RegistJSON
 paramList.Clear();
 paramList.Add(RESOUCE_PATH);
 paramList.Add("{¥"test¥":1}");
 m_CaoController.Execute("RegistJSON", paramList.ToArray());

 // RegistCSV
 paramList.Clear();
 string csvStr = @"name,age,tel
Tanaka and 30,012-345-6789
"of ..Suzuki.. 40,098-765-4321;
 paramList.Add(RESOUCE_PATH);
 paramList.Add(csvStr);
 m_CaoController.Execute("RegistCSV", paramList.ToArray());

 // RegistTXT
 paramList.Clear();
 paramList.Add(RESOUCE_PATH);
 paramList.Add("testTxt");
 m_CaoController.Execute("RegistTXT", paramList.ToArray());

 // RegistBIN
 paramList.Clear();
 paramList.Add(RESOUCE_PATH);
 paramList.Add("MimeType=image/jpeg");
 if (!string.IsNullOrEmpty(FILE_PATH))
 {
```

```
        paramList.Add(File.ReadAllBytes(FILE_PATH));
        m_CaoController.Execute("RegistBIN", paramList.ToArray());
        }

      // Reference to resource data
        // ReferenceLatestData
        paramList.Clear();
        paramList.Add(RESOUCE_PATH);
        var referenceLatestData = m_CaoController.Execute("ReferenceLatestData",
      paramList.ToArray());

        // ReferencePastData
        paramList.Clear();
        paramList.Add(RESOUCE_PATH);
        paramList.Add(REGIST_DATE);
        var referencePastData = m_CaoController.Execute("ReferencePastData", paramList.ToArray());

      // Retrieval of resource data
        // RetrieveData
        paramList.Clear();
        paramList.Add(string.Format("{0}/$all", RESOUCE_PATH));
        paramList.Add("json");
        paramList.Add(null);
        paramList.Add(100);
        var retrieveData = m_CaoController.Execute("RetrieveData", paramList.ToArray());

        // DataCount
        paramList.Clear();
        paramList.Add(string.Format("{0}/$all", RESOUCE_PATH));
        paramList.Add("_date gt 20180830T082135.645Z");
        var dataCount = m_CaoController.Execute("DataCount", paramList.ToArray());

      // Update of resource data
        // UpdateJSON
        paramList.Clear();
        paramList.Add(RESOUCE_PATH);
        paramList.Add(REGIST_DATE);
```

```
        paramList.Add("{¥"test¥":2}");

        m_CaoController.Execute("UpdateJSON", paramList.ToArray());

        // UpdateCSV
        paramList.Clear();
        csvStr = @"name,age,tel
Tanaka and 31,012-345-6789
"of ..Suzuki.. 42,098-765-4321;
        paramList.Add(RESOUCE_PATH);
        paramList.Add(REGIST_DATE);
        paramList.Add(csvStr);
        m_CaoController.Execute("UpdateCSV", paramList.ToArray());

        // UpdateTXT
        paramList.Clear();
        paramList.Add(RESOUCE_PATH);
        paramList.Add(REGIST_DATE);
        paramList.Add("testTxt2");
        m_CaoController.Execute("UpdateTXT", paramList.ToArray());

        // UpdateBIN
        paramList.Clear();
        paramList.Add(RESOUCE_PATH);
        paramList.Add(REGIST_DATE);
        paramList.Add("MimeType=image/jpeg");
        if (!string.IsNullOrEmpty(FILE_PATH))
        {
        paramList.Add(File.ReadAllBytes(FILE_PATH));
        m_CaoController.Execute("UpdateBIN", paramList.ToArray());
        }

    // Deletion of resource data
        // DeleteData
        paramList.Clear();
        paramList.Add(RESOUCE_PATH);
        paramList.Add(string.Format("_date gt {0}", REGIST_DATE));
```

```
        m_CaoController.Execute("DeleteData", paramList.ToArray());
}
          }
        }
```

5.4. Resource control

Resource registration, resource metadata reference, resource metadata update, resource deletion

List 5-4

```
Abbreviation.
using System;
using System.Text;
using System.IO;
using System.Collections.Generic;
using ORiN2.ManagedCAO;

namespace IOTPLATFORM_Sample
{
 public class Sample
  {

private readonly string BASE_URL = @"http://<zone>.fujitsu.com";
private readonly string TENANT_ID = "TenantID";
private readonly string ACCESS_CODE = "AccessCode";
private readonly string EVENT_ACCESS_CODE = "TemperatureCDL";
private readonly string RESOURCE_OPT = "ResourcePath=Test";
private readonly string FILE_PATH = "jpegFilePath";
private readonly string RESOUCE_PATH = "Test";
private readonly string REGIST_DATE = "20180831T023219.222Z";

private CCaoEngine m_CaoEngine = null;
private CCaoWorkspaces m_CaoWorkspaces = null;
private CCaoWorkspace m_CaoWorkspace = null;
private CCaoController m_CaoController = null;

        private void executSample ()
{
          // CAO engine generation
m_CaoEngine = new CCaoEngine();
m_CaoWorkspaces = m_CaoEngine.Workspaces;
m_CaoWorkspace = m_CaoWorkspaces[0];

// Controller argument
 var opstionStr = string.Format("BaseURL={0},TenantID={1},AccessCode={2}", BASE_URL, TENANT_ID,
ACCESS_CODE);

 if (m_CaoController == null)
 {
// Connect it with the controller.
```

```
        m_CaoController = m_CaoWorkspace.AddController("Test", "CaoProv.FUJITSU.IOTPLATFORM", null,
      opstionStr);
        }
        var paramList = new List<object>();
        var paramStrList = new List<string>();

      // Resource control
      // Registration of resource
        // RegistMetadata
        paramStrList.Add(string.Format("{0}/child", RESOUCE_PATH));
        paramList.Add("{¥"resource¥": {¥"retention_period¥" : 2}}");
        m_CaoController.Execute("RegistMetadata", paramStrList.ToArray());

      // Refer to the meta data of the resource.
        // ReferenceMetadata
        paramList.Clear();
        paramList.Add(string.Format("{0}/$all", RESOUCE_PATH));
        paramList.Add(100);
        var referenceMetadata = m_CaoController.Execute("ReferenceMetadata",
      paramStrList.ToArray());

      // Update of meta data of resource
        // UpdateMetadata
        paramStrList.Clear();
        paramStrList.Add(string.Format("{0}/child", RESOUCE_PATH));
        paramStrList.Add("{¥"resource¥": {¥"retention_period¥" : 3}}");
        //{"resource": {"retention_period" : 3}}
        m_CaoController.Execute("UpdateMetadata", paramStrList.ToArray());

      // Deletion of resource
        // DeleteMetaData
        m_CaoController.Execute("DeleteMetaData", string.Format("{0}/child", RESOUCE_PATH));
}
        }
        }
```

5.5. Access code control

Register access code, reference access code, update access code, delete access code.

List 5-5

```
Abbreviation.
using System.Collections.Generic;
using ORiN2.ManagedCAO;

namespace IOTPLATFORM_Sample
{
 public class Sample
  {

private readonly string BASE_URL = @"http://<zone>.fujitsu.com";
private readonly string TENANT_ID = "TenantID";
private readonly string ACCESS_CODE = "AccessCode";

private CCaoEngine m_CaoEngine = null;
private CCaoWorkspaces m_CaoWorkspaces = null;
private CCaoWorkspace m_CaoWorkspace = null;
private CCaoController m_CaoController = null;

        private void executSample ()
{
          // CAO engine generation
m_CaoEngine = new CCaoEngine();
m_CaoWorkspaces = m_CaoEngine.Workspaces;
m_CaoWorkspace = m_CaoWorkspaces[0];

// Controller argument
 var opstionStr = string.Format("BaseURL={0},TenantID={1},AccessCode={2}", BASE_URL, TENANT_ID,
ACCESS_CODE);

 if (m_CaoController == null)
 {
// Connect it with the controller.
 m_CaoController = m_CaoWorkspace.AddController("Test", "CaoProv.FUJITSU.IOTPLATFORM", null,
opstionStr);
 }

 var paramList = new List<object>();
 var paramStrList = new List<string>();
```

```
// Access code control
// Resource passing making for access code giving
 var accessCodeRegistResoucePath = "Test/child";
 paramStrList.Add(accessCodeRegistResoucePath);
 paramStrList.Add("{¥"resource¥": {¥"retention_period¥" : 2}}");
 m_CaoController.Execute("RegistMetadata", paramStrList.ToArray());

// Registration access code
 var registAccessCode = "AccessCodeTest";

// Registration of access code
 // RegistAccessCodeData
 paramStrList.Clear();
 paramStrList.Add(registAccessCode);
 var registAccessCodeJson = "{¥"access_code¥":{ ¥"permissions¥": {¥"resource_operations¥":
[{¥"operations¥": [¥"update¥",¥"read¥"],¥"resource_path¥": ¥"Test/child¥"}]}}}";
 paramStrList.Add(registAccessCodeJson);
 m_CaoController.Execute("RegistAccessCodeData", paramStrList.ToArray());

// Reference to access code
 // ReferenceAccessCodeData
 paramList.Clear();
 paramList.Add(registAccessCode);
 var referenceAccessCodeData = m_CaoController.Execute("ReferenceAccessCodeData",
paramList.ToArray());

 // DataCountAccessCode
 paramStrList.Clear();
 paramStrList.Add(registAccessCode);
 var dataCountAccessCode = m_CaoController.Execute("DataCountAccessCodeData",
paramStrList.ToArray());

// Update of access code
 // UpdateAccessCodeData
 paramStrList.Clear();
 paramStrList.Add(registAccessCode);
 paramStrList.Add(registAccessCodeJson);
```

```
        m_CaoController.Execute("UpdateAccessCodeData", paramStrList.ToArray());


       // Deletion of access code
        // DeleteAccessCode
        m_CaoController.Execute("DeleteAccessCodeData", registAccessCode);


       // Resource deletion for access code giving
        m_CaoController.Execute("DeleteMetaData", accessCodeRegistResoucePath);
}
          }
        }
```

5.6. Event control

Registering events, browsing event information, updating event information, deleting events

List 5-6

```
        Abbreviation.
        using System;
        using System.Collections.Generic;
        using ORiN2.ManagedCAO;

        namespace IOTPLATFORM_Sample
        {
         public class Sample
          {
        private readonly string BASE_URL = @"http://<zone>.fujitsu.com";
        private readonly string TENANT_ID = "TenantID";
        private readonly string EVENT_ACCESS_CODE = "TemperatureCDL";


        private CCaoEngine m_CaoEngine = null;
        private CCaoWorkspaces m_CaoWorkspaces = null;
        private CCaoWorkspace m_CaoWorkspace = null;
        private CCaoController m_CaoController = null;

                private void executSample()
        {
                   // CAO engine generation
        m_CaoEngine = new CCaoEngine();
        m_CaoWorkspaces = m_CaoEngine.Workspaces;
        m_CaoWorkspace = m_CaoWorkspaces[0];


       // Controller argument
```

```
 var opstionStr = string.Format("BaseURL={0},TenantID={1},AccessCode={2}", BASE_URL, TENANT_ID,
EVENT_ACCESS_CODE);

 if (m_CaoController == null)
 {
// Connect it with the controller.
 m_CaoController = m_CaoWorkspace.AddController("Test", "CaoProv.FUJITSU.IOTPLATFORM", null,
opstionStr);
 }
 var paramList = new List<object>();
 var paramStrList = new List<string>();
// Event control
// With the TODO IOTPLATFORM dashboard
// the resource: Make "sensors" "sensors/temperature".
// access code: "sensors" and "CDL" authority giving of "TemperatureCDL" and object resource
// access code: "sensors/temperature" and "RU" authority giving of "TemperatureRU" and object
resource

// Event registration
 // RegistEventData
 var registEventDataJson = "{¥"event¥": {¥"notification¥": {¥"smtp¥": {The
temperature..¥"Subject¥"/¥"¥ that the temperature exceeds 30 times" and ¥"Body¥"/¥ ".. exceeds
30 degrees. Confirm it. ¥",¥"send_to¥": ¥"hoge@piyo.fujitsu.com¥"}},¥"conditions¥":
{¥"targets¥": {¥"read_access_code¥": ¥"TemperatureRU¥",¥"operations¥":
[¥"create¥",¥"update¥"],¥"resource_path¥":
¥"sensors/temperature¥"},¥"notification_condition¥": {¥"body_conditions¥": {¥"path¥":
¥"sensors/temperature¥",¥"path_type¥": ¥"JSONPath¥",¥"comparing_operator¥":
¥"ge¥",¥"value¥": 30.0}}}}}";
 var registEventID = (String)m_CaoController.Execute("RegistEventData", registEventDataJson);

// Reference to event information
 // ReferenceEventData
 paramList.Clear();
 paramList.Add(registEventID);
 var referenceEventData = m_CaoController.Execute("ReferenceEventData", paramList.ToArray());

// Update of event information
```

```
        // UpdateEventData
        var updateEventDataJson = "{¥"event¥": {¥"notification¥": {¥"smtp¥": {The
      temperature ..¥"Subject¥"/¥"¥ that the temperature exceeds 31 times" and ¥"Body¥"/¥ ".. exceeds
      31 degrees. Confirm it. ¥",¥"send_to¥": ¥"hoge@piyo.fujitsu.com¥"}},¥"conditions¥":
      {¥"targets¥": {¥"read_access_code¥": ¥"TemperatureRU¥",¥"operations¥":
      [¥"create¥",¥"update¥"],¥"resource_path¥":
      ¥"sensors/temperature¥"},¥"notification_condition¥": {¥"body_conditions¥": {¥"path¥":
      ¥"sensors/temperature¥",¥"path_type¥": ¥"JSONPath¥",¥"comparing_operator¥":
      ¥"ge¥",¥"value¥": 31.0}}}}}";
        paramStrList.Clear();
        paramStrList.Add(registEventID);
        paramStrList.Add(updateEventDataJson);
        m_CaoController.Execute("UpdateEventData", paramStrList.ToArray());

      // Deletion of event
        // DeleteEventData
        m_CaoController.Execute("DeleteEventData", registEventID);
}
        }
      }
```