

ECloud MQTT

Design Specification for Device

Version : 1.2.3

Table of Contents

1. Connect to MQTT Broker of ECloud	4
1.1 Connection Parameter.....	4
1.1.1 Host Name & Port	4
1.1.2 Client ID	4
1.1.3 QoS	5
1.1.4 Username / Password.....	5
1.1.5 SSL/TLS	5
1.1.6 Conclusion	6
1.2 Topic Define	6
1.2.1 Data.....	6
1.2.2 File	6
1.2.3 Configuration.....	6
1.2.4 Command.....	6
1.2.5 Connection.....	7
1.3 Root Json Format	7
2. MQTT Communication Format.....	7
2.1 Data Format	8
2.1.1 Tag Value	8
2.1.2 The Rules of Device Data Communication	10
2.1.3 Data Recovery	11
2.2 File Format	12
2.2.1 File Upload	13
2.2.2 File Download	14
2.3 Configuration Format	15
2.3.1 Introduction.....	15
2.3.2 Device.....	16
2.3.3 Tag Common Part	17
2.3.4 Analog Tag Advance Part.....	18
2.3.5 Digit Tag Advance Part.....	19
2.3.6 Configuration Download	20
2.3.7 Add Tag Example.....	21
2.3.8 Update Tag Example.....	25
2.3.9 Delete Tag Example.....	25
2.3.10 Devices Plug and Play Mechanism.....	26
2.4 Command Format	27
2.4.1 Write Value	27
2.4.2 Write Config	28
2.4.3 Data On.....	29
2.4.4 Data Off	30
2.4.5 Backup Configuration.....	30
2.4.6 Firmware Upgrade	31

2.4.7 File Download	31
2.4.8 Time Synchronization	32
2.4.9 Command List	33
2.5 Connection Format	33
2.5.1 Connect.....	34
2.5.2 Disconnect	34
2.5.3 Unexpected Disconnect	35
2.5.4 Heartbeat.....	36
2.5.5 Time Synchronization	37
2.6 Action Format	38
2.6.1 Connect.....	38
2.6.2 Unexpected Disconnect	39
2.6.3 Config	39
2.6.4 File.....	40

1. Connect to MQTT Broker of ECloud

Devices which want to connect to ECloud must go with MQTT.

1.1 Connection Parameter

1.1.1 Host Name & Port

Host Name: Domain name or IP address with "/MyMQTT/" of MQTT broker

Ex: 192.168.0.1/MyMQTT/

Port: Port of MQTT broker. Default port of WS(WebSocket), WSS(Secure WebSocket) and TCP are 80, 443, 1883. It is strongly recommended devices go with MQTT over WSS for security.

1.1.2 Client ID

Client ID must be unique for each device and the format is:

d:group_id:type_id:dev_id

1.1.2.1 d

It means devices.

1.1.2.2 group_id

group_id is composed by ECloud project name and SCADA name. It is separated by a under line, for example: MyProject_MySCADA.

The maximum length of **group_id** is 65, and each of project name and SCADA name is 32.

1.1.2.3 type_id

It means device type.

type_id	Device Type
01	Temperature
02	Humidity
03	Wind power
04	Rainfall
05	PM2.5
06	Smoke
07	Wind energy
08	Solar energy
09	Video
10	LED
11	Infrared

type_id	Device Type
...	Other equipment (expandable)

1.1.2.4 dev_id

Device Type

It means device ID. The maximum length is 32. **dev_id** must be unique under same **group_id**.

1.1.3 QoS

The quality of service (QoS) level is an agreement between sender and receiver of a message regarding the guarantees of delivering a message.

1.1.3.1 QoS0

At most once.

1.1.3.2 QoS1

At least once. (QoS1 is default suggestion)

1.1.3.3 QoS2

Exactly once. (not support yet)

1.1.4 Username / Password

User name and password for MQTT broker

Each of maximum length is 32.

1.1.5 SSL/TLS

ECloud MQTT broker is support SSL/TLS for WSS only.

1.1.6 Conclusion

With those connection parameters for ECloud MQTT broker, there must be some UI for user to input relative information on the devices.

1.2 Topic Define

There are several MQTT topic as the following:

1.2.1 Data

mqtt-main/evt/**mydata**/fmt/**group_id**

This topic is for devices sending their real-time data.

1.2.2 File

mqtt-main/evt/**myfile**/fmt/**group_id**

This topic is for devices sending their file.

1.2.3 Configuration

mqtt-main/evt/**mycfg**/fmt/**group_id**

This topic is for devices sending their configuration.

1.2.4 Command

mqtt-main/evt/**mycmd**/fmt/**group_id**

mqtt-main/evt/**mycmd**/fmt/**group_id**/**dev_id**

This two topics are for devices receiving command from cloud.

1.2.5 Connection

mqtt-main/evt/**myconn**/fmt/**group_id**

This topic is for devices sending their connection status.

1.3 Root Json Format

There are only two property in root json format.

Format:

```

    {
      "d": {

      },
      "ts": "2017-04-04T23:26:10+08:00"
    }
  
```

Property:

Name	Limitation	Description
d	Json	User define data
	Format	Data Recovery
ts		UTC Time stamp, support format as the following: <ol style="list-style-type: none"> 2017-05-13T02:52:51\mathbf{Z} (seconds) 2017-05-13T02:52:51.742\mathbf{Z} (milliseconds) 2017-05-13T10:52:51+08:00 (seconds with time zone) 2017-05-13T10:52:51.742+08:00 (milliseconds with time zone)

2. MQTT Communication Format

In this chapter, it will define json format of data, file, configuration, command and connection topics.

There are several representation styles as the following:

- Red means primary key
- Green means necessary

- Blue means optional but nice to have
- Black means optional
- ... means user define config space

It is case sensitive for all json property.

2.1 Data Format

Devices publish real-time data to this topic:

mqtt-main/evt/mydata/fmt/group_id

2.1.1 Tag Value

It is important for saving bandwidth that devices should publish real-time data which the tag values are changed. It is recommended to have deadband setting for each tag when difference between current and last published value is over some percentage.

Format:

```

{
  "unique string": {
    "Val": {
      "unique string1": number,
      "unique string2": "string",
      "unique string3": {
        "0": number,
        "1": number,
        "2": number,
        ...
      }
    }
  },
  ...
}

```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under <u>same group_id</u>)
DRec		Data Recovery
From		Seconds since 00:00 hours, Jan 1, 1970 UTC
unique string1~N	21	Tag ID, must be unique under same EAccess

		<p>project and can not be "(", "&", ",", ":", ".", "%", "=", "# and space.</p> <p>(Tag ID is mapping to EAccess Tag Name and Tag Address)</p> <p>Tag Value can be number, string or array (index from 0). One single "*" is for bad value. The value should be bad when devices or I/O can not be reached.</p>
offset 1~N		<p>Time Offset with "From" in seconds. Value can be number, string or array (index from 0)</p>

*The maximum length of EAccess Tag Name is limited to 21 bytes.

Example for normal value:

```

    {
      "d": {
        "WISE4070-7F28A7": {
          "Val": {
            "W4070-26A7_Fz1Volt": 1213.48,
            "W4070-26A7_Fz1Temp": 63.81,
            "W4070-26A7_Fz1Desc": "Freezer 1",
          }
        }
      },
      "ts": "2017-04-17T10:19:51+08:00"
    }

```

Example for array value:

```

    {
      "d": {
        "WISE4070-7F28A7": {
          "Val": {
            "W4070-26A7_Fz1Volt": {
              "0": 963.56,
              "3": 1147.38,
              "4": 1038.84
            }
          }
        }
      }
    }

```

```

    }
  }
},
"ts": "2017-04-17T10:19:51+08:00"
}

```

Example for bad value:

```

{
  "d": {
    "WISE4070-7F28A7": {
      "Val": {
        "W4070-26A7_Fz1Volt": "*",
        "W4070-26A7_Fz1Temp": "*",
        "W4070-26A7_Fz1Desc": "*"
      }
    }
  },
  "ts": "2017-04-17T10:19:51+08:00"
}

```

2.1.2 The Rules of Device Data Communication

It supports primary and backup devices mechanism on ECloud. The backup devices is switched to data communication with cloud when the primary devices are lost. And it will switch back to primary when the primary devices is come back.

For saving network bandwidth(2.1.1):

All devices publish real-time data to cloud when value of tags is changed.

For primary and backup devices(2.3.2):

All devices can not publish real-time data at the start.

For command of data on(2.4.4) and data off(2.4.5):

The devices should publish real-time data when receiving command of data on, and publish all values only at first time.

The devices should stop publishing real-time data when receiving command of data off.

2.1.3 Data Recovery

Cloud sides can collect data from devices and record those to database for query or analytics usage. However, it gets data lost when devices is offline unexpectedly. Therefore, It will be necessary to recover the lost data as the following when devices found that they didn't publish some period of data successfully.

Format:

```

    {
      "unique string": {
        "DRec": {
          "From": number,
          "Tags": {
            "unique string1": {
              "offset1": number,
              "offset2": number,
              ...
            },
            "unique string2": {
              "offset1": "string",
              "offset2": "string",
              ...
            }
          }
        }
      },
      ...
    }
  
```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under same <u>group_id</u>)
DRec		Data Recovery
From		Seconds since 00:00 hours, Jan 1, 1970 UTC
unique string1~N	21	Tag ID, must be unique under same EAccess project and can not be "(", "&", ",", ":", ".", "%", "=", "# and space.

Name	Limitation	Description
		(Tag ID is mapping to EAccess Tag Name and Tag Address) Tag Value can be number, string or array (index from 0). One single "*" is for bad value. The value should be bad when devices or I/O can not be reached.
offset 1~N		Time Offset with " From " in seconds. Value can be number, string or array (index from 0)

Example:

```

{
  "d": {
    "WISE4070-7F28A7": {
      "DRec": {
        "From": 1460129890,
        "W4070-26A7_Fz1Volt": {
          "3": 147.32,
          "18": 149.18,
          "87": 148.93
        },
        "W4070-26A7_Fz2Volt ": {
          "0": 73.83,
          "118": 84.49,
          "124": 68.17
        }
      }
    }
  }
  "ts": "2017-04-17T10:19:51+08:00"
}

```

2.2 File Format

Devices publish file to this topic:

mqtt-main/evt/myfile/fmt/group_id

2.2.1 File Upload

Upload file from devices to cloud

Format:

```

    {
      "unique string": {
        "FPut": {
          "Name": "string",
          "Ind": number,
          "Type": number,
          "Size": number,
          "Bin": binary
        }
      },
      ...
    }
  
```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under same <u>group_id</u>)
FPut		File Upload to Cloud
Name	256	{File Path}/{File Name}
Ind	0 or 1	Is Individual for Root Path: 0: ... \EAccess\Node\mqtt\ 1: ... \EAccess\Node\config\{ group_id }\
Type	0 ~ 2	0: ASCII 1: UTF-8 2: Binary
Size	<=256MB	File Size In Bytes (Up to 4GB)
Bin		File Content

Example for file upload:

```

    {
      "d": {
        "WISE4070-7F28A7": {
          "FPut": {
            "Name": "W4070-26A7/image.bin",
            "Ind": 1,
            "Type": 2,
            "Size": 3576,
            "Bin": "content..."
          }
        }
      },
      "ts": "2017-04-17T10:19:51+08:00"
    }

```

2.2.2 File Download

Download file from cloud to device

Format:

```

    {
      "unique string": {
        "FGet": {
          "Name": "string",
          "Ind": number
        }
      },
      ...
    }

```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under same <code>group_id</code>)
FGet		File Download from Cloud
Name	256	{File Path}/{File Name}
Ind	0 or 1	Is Individual for Root Path: 0: ...\\EAccess\\Node\\mqtt\ 1: ...\\EAccess\\Node\\config\\{ group_id }\\

Example for file download:

```

    {
      "d": {
        "WISE4070-7F28A7": {
          "FGet": {
            "Name": "W4070-26A7/image.bin",
            "Ind": 1
          }
        }
      },
      "ts": "2017-04-17T10:19:51+08:00"
    }
  
```

(file download response from command "Fdl" at [2.4.8](#))

2.3 Configuration Format

Devices publish configuration to this topic:

mqtt-main/evt/mycfg/fmt/**group id**

2.3.1 Introduction

On ECloud, it has hierarchy configuration as the following:

```

+Project
  +SCADA
    +Port
      +Device
        +Tag
  
```

It can be configured automatically on ECloud if devices publish sufficient configuration information. Devices should keep a last uploaded configuration profile, and sending differential part to cloud every time by "UTg" or "DTg". In this way, devices can add, modify or delete tag configuration on ECloud.

2.3.2 Device

Format:

```

{
  "unique string": {
    "TID": number,
    "Dsc": "string",
    "Hbt": number,
    "PRI": "string",
    "UTg": {tag format},
    "DTg": {tag format},
    "Del": 1,
    ...
  }
}

```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under same <u>group_id</u>)
TID	1~11	Device Type (1.1.2.3) ...
Dsc	64	Description
Hbt	1~65535	Interval of Heartbeat (2.5.4), Suggest 5 or 10 Seconds Should be Good
BID	31	Backup Device ID left empty if the device is backup device left backup device ID if the device is backup
UTg		Add or Update Tag
DTg		Delete Tag
Del	1	Delete Device and All Tags on Cloud

PS: Command order is "Del" → "DTg" → "UTg" if they show at same time

2.3.3 Tag Common Part

Format:

```

    "unique string": {
        "TID": number,
        "Dsc": "string",
        "Ary": number,
        "RO": number,
        ...
    }

```

Property:

Name	Limitation	Description	Default
uniquestring	21	Tag ID, must be unique under same EAccess project and can not be "(", "&", ",", ":", ".", "%", "=", "# and space. (Tag ID is mapping to EAccess Tag Name and Tag Address)	
TID	1~5	Tag Type 1: Analog, 2: Digit, 3: Text	
Dsc	64	Description	
Ary	0~32767	Array Size	0
RO	0 or 1	Read Only (0:false, 1:true)	0

*The maximum length of EAccess Tag Name is limited to 21 bytes.

2.3.3.1 Tag ID

It is recommend to use abbreviation because the length is limitation. EAccess Cloud is tag based architecture so it is suggested that using the format as thefollowing will be good:

{Device Unique Abbreviation}_{Tag Abbreviation}

For example, there is a WISE4012E with partial MAC address A924 and one of analog input channel can be show:

W4012-A924_AI0

Notice that devices should keep Tag ID to be unique under the same EAccess project.\

2.3.4 Analog Tag Advance Part

Format:

```

{
    "Log": number,
    "SH": number,
    "SL": number,
    "EU": "string",
    "DSF": "string",
    "Alm": number,
    "HHP": number,
    "HHA": number,
    "HiP": number,
    "HiA": number,
    "LoP": number,
    "LoA": number,
    "LLP": number,
    "LLA": number,
    ...
}

```

Property:

Name	Limitation	Description	Default
Log	0 or 1	Log Data to RTDB (0:disabled, 1:enabled), RTDB means EAccess Real-time Database	0
SH	double	Span High	1000
SL	double	Span Low	0
EU	10	Engineer Unit	
DSF	xx.xx	Display Format (Integer.Fraction) (xx: 0~15)	4.2
Alm	0 or 1	Alarm Enabled (0:false, 1:true)	0
HHP	0~99	HH Priority (0:disabled)	0
HHA	double	HH Alarm Limit	
HiP	0~99	High Priority (0:disabled)	0
HiA	double	High Alarm Limit	

Name	Limitation	Description	Default
LoP	0~99	Low Priority (0:disabled)	0
LoA	double	Low Alarm Limit	
LLP	0~99	LL Priority (0:disabled)	0
LLA	double	LL Alarm Limit	

Note1: HHP >= HiP, LLP >= LoP

Note2: Alm must be taken at same time when some relative alarm setting is updated

2.3.5 Digit Tag Advance Part

Format:

```

{
    "Log": number,
    "S0": "string",
    "S1": "string",
    "S2": "string",
    "S3": "string",
    "S4": "string",
    "S5": "string",
    "S6": "string",
    "S7": "string",
    "Alm": number,
    "S0L": number,
    "S1L": number,
    "S2L": number,
    "S3L": number,
    "S4L": number,
    "S5L": number,
    "S6L": number,
    "S7L": number,
    ...
}

```

Property:

Name	Limitation	Description	Default
Log	0 or 1	Log Data	0

Name	Limitation	Description	Default
S0	12	State 0	0
S1	12	State 1	1
S2	12	State 2	NotUsed
S3	12	State 3	NotUsed
S4	12	State 4	NotUsed
S5	12	State 5	NotUsed
S6	12	State 6	NotUsed
S7	12	State 7	NotUsed
Alm	0 or 1	Alarm Enabled (0:false, 1:true)	0
S0P	0~99	State 0 Alarm Priority	0
S1P	0~99	State 1 Alarm Priority	0
S2P	0~99	State 2 Alarm Priority	0
S3P	0~99	State 3 Alarm Priority	0
S4P	0~99	State 4 Alarm Priority	0
S5P	0~99	State 5 Alarm Priority	0
S6P	0~99	State 6 Alarm Priority	0
S7P	0~99	State 7 Alarm Priority	0

2.3.6 Configuration Download

Devices should keep a up-to-date configuration profile. If devices miss or mess up the profile, it can be restored by this request command. For example: device replacement, the new one need to fill up the device ID of the broken one, then using this request command to get back up-to-date configuration profile from cloud.

Format:

```

{
  "unique string": {
    "Cdl": 1
  }
}

```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under <u>same group id</u>)
Cdl	1	Config Download

It will return configuration from [2.4.2](#) after cloud receive this request commnad.

Example:

```
{
  "d": {
    "WISE4070-7F28A7": {
      "Cdl": 1
    }
  },
  "ts": "2017-04-17T10:19:51+08:00"
}
```

2.3.7 Add Tag Example

```
{
  "d": {
    "WISE4070-7F28A7": {
      "TID": 3,
      "Dsc": "Factory A",
      "Hbt": 5,
      "PRI": "",
      "UTg": {
        "W4070-26A7_Fz1Volt": {
          "TID": 1,
          "Dsc": "Freezer1 Voltage",
          "RO": 1,
          "Log": 1,
          "SH": 2500,
          "SL": 0,
          "EU": "V",
          "DSF": "4.2",
          "Alm": 1,
          "HHP": 0,
          "HHA": 0,
          "HiP": 1,
          "HiA": 1800,
          "LoP": 1,
          "LoA": 500,
          "LLP": 0,

```

```
"LLA": 0
},
"W4070-26A7_Fz2Volt": {
  "TID": 1,
  "Dsc": "Freezer2 Voltage",
  "RO": 1,
  "Log": 1,
  "SH": 2500,
  "SL": 0,
  "EU": "V",
  "DSF": "4.2",
  "Alm": 1,
  "HHP": 0,
  "HHA": 0,
  "HiP": 1,
  "HiA": 1800,
  "LoP": 1,
  "LoA": 500,
  "LLP": 0,
  "LLA": 0
},
"W4070-26A7_Fz1Temp": {
  "TID": 1,
  "Dsc": "Freezer1 Temperature",
  "RO": 1,
  "Log": 1,
  "SH": 150,
  "SL": -50,
  "EU": "°C",
  "DSF": "3.2",
  "Alm": 1,
  "HHP": 0,
  "HHA": 0,
  "HiP": 1,
  "HiA": 90,
  "LoP": 1,
  "LoA": -10,
  "LLP": 0,
```

```
        "LLA": 0
    },
    "W4070-26A7_Fz2Temp": {
        "TID": 1,
        "Dsc": "Freezer2 Temperature",
        "RO": 1,
        "Log": 1,
        "SH": 150,
        "SL": -50,
        "EU": "°C",
        "DSF": "3.2",
        "Alm": 1,
        "HHP": 0,
        "HHA": 0,
        "HiP": 1,
        "HiA": 90,
        "LoP": 1,
        "LoA": -10,
        "LLP": 0,
        "LLA": 0
    },
    "W4070-26A7_Fz1Switch": {
        "TID": 2,
        "Dsc": "Freezer1 Switch",
        "RO": 0,
        "Log": 1,
        "S0": "On",
        "S1": "Off",
        "S2": "",
        "S3": "",
        "S4": "",
        "S5": "",
        "S6": "",
        "S7": "",
    },
    "W4070-26A7_Fz1Mode": {
        "TID": 2,
```

```
"Dsc": "Freezer1 Mode Switch",
"RO": 0,
"Log": 1,
"S0": "Save",
"S1": "Safe",
"S2": "Lv1",
"S3": "Lv2",
"S4": "Lv3",
"S5": "Manual",
"S6": "",
"S7": ""
},
"W4070-26A7_Fz2Switch": {
"TID": 2,
"Dsc": "Freezer2 Switch",
"RO": 0,
"Log": 1,
"S0": "On",
"S1": "Off",
"S2": "",
"S3": "",
"S4": "",
"S5": "",
"S6": "",
"S7": ""
},
"W4070-26A7_Fz2Mode": {
"TID": 2,
"Dsc": "Freezer2 Mode Switch",
"RO": 0,
"Log": 1,
"S0": "Save",
"S1": "Safe",
"S2": "Lv1",
"S3": "Lv2",
"S4": "Lv3",
"S5": "Manual",
```



```

        "S6": "",
        "S7": ""
    },
    },
    },
    "ts": "2017-04-17T10:19:51+08:00"
}

```

2.3.8 Update Tag Example

```

{
  "d": {
    "WISE4070-7F28A7": {
      "TID": 3,
      "Dsc": "Factory A",
      "Hbt": 5,
      "PRI": "",
      "UTg": {
        "W4070-26A7_Fz1Volt": {
          "SH": 5000,
          "SL": 500,
        },
        "W4070-26A7_Fz2Volt": {
          "SH": 7500,
          "SL": 2000,
        }
      }
    }
  },
  "ts": "2017-04-17T10:19:51+08:00"
}

```

2.3.9 Delete Tag Example

```

{
  "d": {
    "WISE4070-7F28A7": {

```

```

        "TID": 3,
        "Dsc": "Factory A",
        "Hbt": 5,
        "PRI": "",
        "DTg": {
            "W4070-26A7_Fz2Temp": 1,
            "W4070-26A7_Fz2Switch": 1
        }
    },
    "ts": "2017-04-17T10:19:51+08:00"
}

```

2.3.10 Devices Plug and Play Mechanism

It still need to setup project, SCADA, ports , devices and tags because there is also a EAccess software on cloud side. To reduce this configuration effort, auto-configuration from each device is a must feature, as we call "Plug and Play". Devices have control right to add, modify and delete configuration of EAccess on cloud side.

Devices must support these two way for Plug and Play:

1. Add full configuration: through the way like [2.3.7](#). Usually, it is used in scenario as the following:
 - connect to the group id at first time
 - device configuration was deleted by cloud side
2. Partial update or delete configuration: through the way like [2.3.8](#) and [2.3.9](#). Devices should keep a up-to-date configuration profile, and update or delete configuration based on this profile.

Device replacement scenario:

Assume A is broken device, and B is new one.

1. Find A and ready to replace with B.
2. Set device ID of A to B
3. Let B connect to cloud and get back up-to-date configuration profile through the way [at 2.3.6](#). B should setup well according to this profile.
4. Start to publish real-time data through way [at 2.1.1](#) when B all gets ready.

2.4 Command Format

Devices should subscribe these two topics to receive commands from cloud:

mqtt-main/evt/mycmd/fmt/group id

mqtt-main/evt/mycmd/fmt/group id/dev id

2.4.1 Write Value

Write values from cloud to the device

mqtt-main/evt/mycmd/fmt/group id/dev id

Format:

```

{
  "Cmd": "WV",
  "Val": {
    "unique string1": number,
    "unique string2": "string"
    "unique string3": {
      "0": number,
      "1": number,
      "2": number,
      ...
    }
  }
}

```

Property:

Name	Limitation	Description
unique string1~3	21	Tag ID, must be unique under same EAccess project and can not be "(", "&", ",", ":", ".", "%", "=", "# and space. (Tag ID is mapping to EAccess Tag Name and Tag Address)

Example:

```

{
  "d": {
    "Cmd": "WV",

```

```

        "Val": {
            "W4070-26A7_Fz1Volt": 1213.48,
            "W4070-26A7_Fz1Temp": 63.81
        }
    },
    "ts": "2017-04-17T10:19:51+08:00"
}

```

2.4.2 Write Config

Write configuration from cloud to the device
mqtt-main/evt/mycmd/fmt/group_id/dev_id

Format:

```

{
    "Cmd": "WC",
    "UTg": {

    },
    "DTg": {

    },
    "Del": 1
}

```

Property:

Name	Limitation	Description
UTg		Add or Update Tags
DTg		Delete Tags
Del	1	Delete All Tags

Example: (modify tag span high and span low, and delete tag)

```

{
    "d": {
        "Cmd": "WC",
        "UTg": {
            "W4070-26A7_Fz2Volt": {
                "SH": 2000,
                "SL": 200
            }
        }
    }
}

```

```

    },
    "W4070-26A7_Fz2Temp": {
        "SH": 150,
        "SL": -20
    }
},
"DTg": {
    "W4070-26A7_Fz1Volt": 1,
    "W4070-26A7_Fz1Temp": 1
},
},
"ts": "2017-04-17T11:58:07+08:00"
}

```

2.4.3 Data On

Indicate devices to start publishing real-time data

mqtt-main/evt/mycmd/fmt/group_id

mqtt-main/evt/mycmd/fmt/group_id/dev_id

Format:

```

{
    "Cmd": "DOn"
}

```

Property:

Cmd Name	Limitation	Description
DOn		start sending real-time data. publish all tag value at first time, then just publish values with changed.

Example:

```

{
    "d": {
        "Cmd": "DOn"
    },
    "ts": "2017-04-17T13:33:29+08:00"
}

```

2.4.4 Data Off

Indicate devices to stop publishing real-time data

mqtt-main/evt/mycmd/fmt/group_id

mqtt-main/evt/mycmd/fmt/group_id/dev_id

Format:

```
{
  "Cmd": "DOf"
}
```

Property:

Cmd Name	Limitation	Description
DOf		stop sending real-time data.

Example:

```
{
  "d": {
    "Cmd": "DOF"
  },
  "ts": "2017-04-17T13:33:29+08:00"
}
```

2.4.5 Backup Configuration

Indicate the device to backup the current device configuration file to cloud

mqtt-main/evt/mycmd/fmt/group_id/dev_id

Format:

```
{
  "Cmd": "BkC"
}
```

Property:

Cmd Name	Limitation	Description
RsC		resotre device configuration file

Example:

```

{
  "d": {
    "Cmd": "RsC",
  },
  "ts": "2017-04-17T13:33:29+08:00"
}

```

2.4.6 Firmware Upgrade

Indicate the device to upgrade firmware from cloud

mqtt-main/evt/mycmd/fmt/group_id/dev_id

Format:

```

{
  "Cmd": "FwU"
}

```

Property:

Cmd Name	Limitation	Description
FwU		Command device update software

Example:

```

{
  "d": {
    "Cmd": "FwU",
    "URL": "http://myvm.cloudapp.net/firmware/WISE4070.bin"
  },
  "ts": "2017-04-17T13:33:29+08:00"
}

```

2.4.7 File Download

Receive download file from cloud

mqtt-main/evt/mycmd/fmt/group_id/dev_id

Format:

```

{
  "Cmd": "Fdl",
}

```

```

    "Name": "string",
    "Type": number,
    "Size": number,
    "Bin": binary
  }

```

Property:

Name	Limitation	Description
Fdl		Files Get from Cloud
Name	256	File Name
Type	0 ~ 2	0: ASCII 1: UTF-8 2: Binary
Size	<=256MB	File Byte Size (Up to 4GB)
Bin		File Content

Example:

```

  {
    "d": {
      "Cmd": "Fdl",
      "File": {
        "Name": "image.bin",
        "Type": 2,
        "Size": 3576,
        "Bin": "content..."
      }
    },
    "ts": "2017-04-17T13:33:29+08:00"
  }

```

(reference request command "FGet" at [2.2.2](#))

2.4.8 Time Synchronization

Receive time synchronization from cloud

mqtt-main/evt/mycmd/fmt/group_id/dev_id

Format:

```

  {
    "Cmd": "TSyn",

```



```

    "UTC": number
  }

```

Property:

Name	Limitation	Description
TSyn		Time Synchronization
UTC		UTC Time from Cloud

Example:

```

{
  "d": {
    "Cmd": "TSyn",
    "UTC": 1460129890
  },
  "ts": "2017-04-17T13:33:29+08:00"
}

```

(reference request command "TSyn" at [2.5.5](#))

2.4.9 Command List

Cmd	Property	Description
WV	Val	Write Value
WC	Cfg	Write Config
DOn		Data On
DOf		Data Off
<u>BkC</u>		<u>Backup Config</u>
<u>RsC</u>		<u>Restore Config</u>
<u>FwU</u>		<u>Firmware Upgrade</u>
Fdl	File	File Download
TSyn	UTC	Time Synchronization

2.5 Connection Format

Devices publish connection status to this topic:

mqtt-main/evt/myconn/fmt/group id

2.5.1 Connect

Notify cloud side after connecting successfully.

Format:

```

{
  "unique string": {
    "Con": 1
  }
}

```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under same <u>group_id</u>)
Con	1	Connect

Example:

```

{
  "d": {
    "WISE4070-7F28A7": {
      "Con": 1
    }
  },
  "ts": "2017-04-17T10:19:51+08:00"
}

```

2.5.2 Disconnect

Notify cloud side after disconnecting successfully.

Format:

```

{
  "unique string": {
    "DsC": 1
  }
}

```

```

    }
}

```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under same <u>group_id</u>)
Dsc	1	Disconnect

Example:

```

{
  "d": {
    "WISE4070-7F28A7": {
      "DsC": 1
    }
  },
  "ts": "2017-04-17T10:19:51+08:00"
}

```

2.5.3 Unexpected Disconnect

Implement with MQTT Last Will mechanism to notify cloud side after unexpected disconnection happening.

Format:

```

{
  "unique string": {
    "UeD": 1
  }
}

```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under same <u>group_id</u>)
Ued	1	Unexpected Disconnect

Example:

```

{
  "d": {
    "WISE4070-7F28A7": {
      "UeD": 1
    }
  },
  "ts": "2017-04-17T10:19:51+08:00"
}

```

2.5.4 Heartbeat

Notify cloud side that the device is alive.

Format:

```

{
  "unique string": {
    "Hbt": 1
  }
}

```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under same <u>group_id</u>)
Hbt	1	Heartbeat for Keep Alive

Example:

```

{
  "d": {
    "WISE4070-7F28A7": {

```

```

        "Hbt": 1
      }
    },
    "ts": "2017-04-17T10:19:51+08:00"
  }

```

2.5.5 Time Synchronization

Synchronize time from cloud side

Format:

```

    {
      "unique string": {
        "TSyn": 1
      }
    }

```

Property:

Name	Limitation	Description
unique string	31	Device ID (unique under same <u>group_id</u>)
TSyn	1	Time Synchronization

Example:

```

    {
      "d": {
        "WISE4070-7F28A7": {
          "TSyn": 1
        }
      },
      "ts": "2017-04-17T10:19:51+08:00"
    }

```

(reference response command "UTC" at [2.4.9](#))

2.6 Action Format

Devices publish action to this topic for cloud:

mqtt-main/evt/myactc/fmt/group_id/dev_id

Devices should subscribe these two topics to receive action from cloud:

mqtt-main/evt/myactd/fmt/group_id

mqtt-main/evt/myactd/fmt/group_id/dev_id

Devices should log these action to help for problem tracking and analyzing.

2.6.1 Connect

Reveive the action after connected to cloud successfully (2.5.1).

mqtt-main/evt/myactd/fmt/group_id/dev_id ("Con": 1)

mqtt-main/evt/myactd/fmt/group_id ("Con": 2)

Format:

```
{
  "Con": 1
}
```

Property:

Name	Limitation	Description
Con	1	Device Connect Successfully
	2	Cloud Agent Connect Successfully

Example:

```
{
  "d": {
    "Con": 1
  },
  "ts": "2017-04-17T10:19:51+08:00"
}
```

2.6.2 Unexpected Disconnect

Receive the action after cloud agent is offline unexpectedly.

mqtt-main/evt/myactd/fmt/group_id

Format:

```
{
  "UeD": 1
}
```

Property:

Name	Limitation	Description
UeD	1	Cloud Agent is Offline Unexpectedly

Example:

```
{
  "d": {
    "Ued": 1
  },
  "ts": "2017-04-17T10:19:51+08:00"
}
```

2.6.3 Config

Receive the action after updated config to cloud successfully [\(2.5.1\)](#).

mqtt-main/evt/myactd/fmt/group_id/dev_id

Notify cloud after updated config from cloud successfully [\(2.4.2\)](#)

mqtt-main/evt/myactc/fmt/group_id/dev_id

Format:

```
{
  "Cfg": 1
}
```

Property:

Name	Limitation	Description
Cfg	1	Updated Config Successfully
	2	Failed to Updated Config

Example:

```

    {
      "d": {
        "Cfg": 1
      },
      "ts": "2017-04-17T10:19:51+08:00"
    }

```

2.6.4 File

Receive the action after uploaded file handling to cloud successfully [\(2.2.1\)](#).

mqtt-main/evt/myactd/fmt/group_id/dev_id

Notify cloud after downloaded file handling from cloud successfully [\(2.4.8\)](#)

mqtt-main/evt/myactc/fmt/group_id/dev_id

Format:

```

    {
      "File": 1
    }

```

Property:

Name	Limitation	Description
File	1	File Handling is Successfully
	2	Failed to Handle File

Example:

```

    {
      "d": {
        "File": 1
      },
      "ts": "2017-04-17T10:19:51+08:00"
    }

```