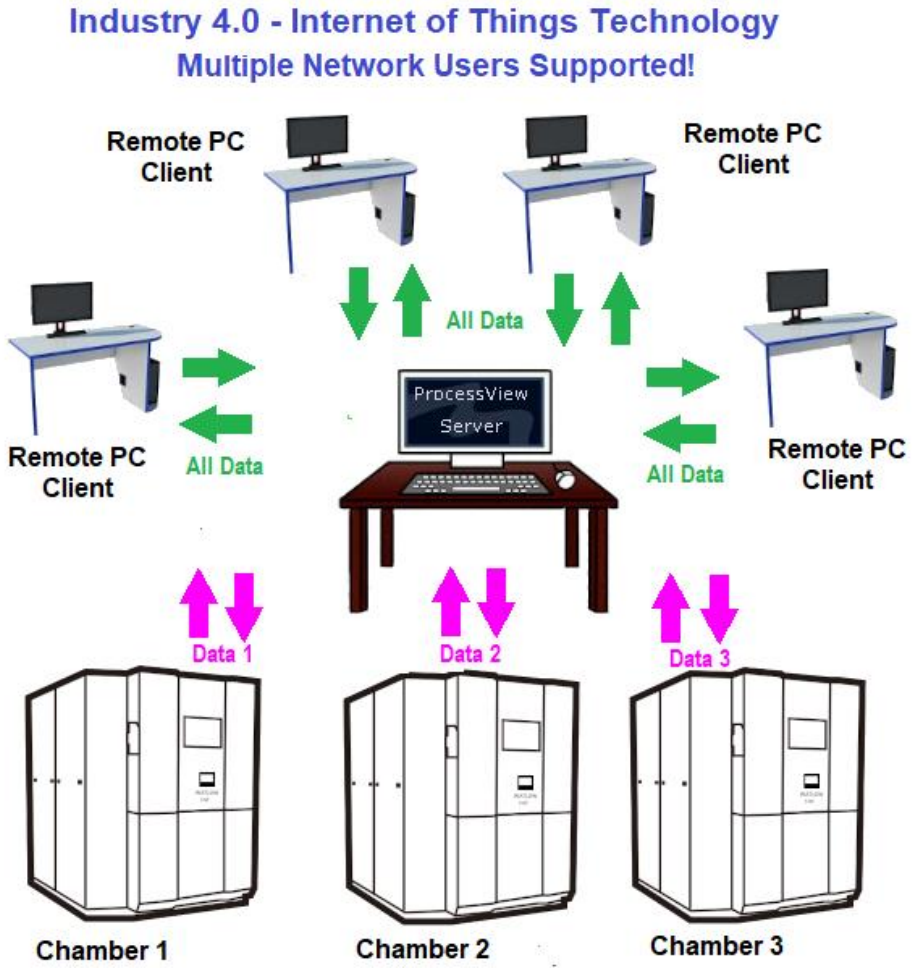
ProcessView® MQTT Interface Specification Rev 1.6

Introduction:

This document details the structure and implementation requirements for ProcessView compliant MQTT Client implementations for ProcessView Server and ProcessView software. It is assumed that the reader of this document is familiar with MQTT protocol and it's requirements. Below is an illustration of how a MQTT network is structured. The center PC has both ProcessView software and ProcessView Server software installed on it. ProcessView software reads and writes all data to and from Watlow Controllers via Modbus TCP/IP over Ethernet. The ProcessView software is also a MQTT client which publishes messages to the ProcessView MQTT Server which other remote MQTT Clients have access to via MQTT. The ProcessView server software can also reside on a remote PC as well instead of the same PC that the ProcessView software resides on.



Introduction to MQTT:

"MQTT is a Client Server publish/subscribe messaging transport protocol. It is light weight, open, simple, and designed so as to be easy to implement. These characteristics make it ideal for use in many situations, including constrained environments such as for communication in Machine to Machine (M2M) and Internet of Things (IoT) contexts where a small code footprint is required and/or network bandwidth is at a premium."

*Citation from the official **MQTT 3.1.1 specification***

The abstract of the MQTT specification does a good job describing what MQTT is all about. It is a very light weight and binary protocol, and due to its minimal packet overhead, MQTT **excels when transferring data over the wire in comparison to protocols like HTTP**. Another important aspect of the protocol is that MQTT is extremely easy to implement on the client side. Ease of use was a key concern in the development of MQTT and makes it a perfect fit for constrained devices with limited resources today.

Client Connection Message Parameters:

The following are the elements in the connection message that are required to establish a connection with the MQTT server:

Client ID: The client identifier (Client ID) **identifies each MQTT client** that connects to an MQTT server. The server uses the Client ID to identify the client and the current state of the client. Therefore, this Id should be unique per client and server. In MQTT 3.1.1 you can send an empty Client ID. if you don't need a state to be held by the server. The empty Client ID results in a connection without any state. In this case, the clean session flag must be set to true or the server will reject the connection.

Server Port Number: The port used by the MQTT server.

Username: The username used by the server.

Password: The password used by the server.

Remote Server Address: The URL or IP address where the server resides.

General Subscribe Structure:

"**F4T / C# / Topic**"

(# is the controller number on the network, or the row number for the Controller in the ProcessView Network Overview Window)

Topic is the Topic part of the message which are specific to the parameter you are trying to read.

Example 1: Read the Event Output Status for Event Output 8 from F4T Controller 2 on the network:

"F4T/C2/EvStat8"

Example 1: Read the Current Batch Operator Name for F4T Controller 1 on the network:

"F4T/C1/BatchOp"

SubscribeTopics:

Communication Topics:

PV\Stat – Read if the ProcessView Client is online (ProcessView MQTT Server only function, True-Online, False-Offline)

Profile Data Topics:

PStat – Read Profile Status (0-None, 1-Running, 2-Paused, 3-Completed, 4-Terminated
PStep-Profile Step Number
PType-Profile Step Type (0-Unknown, 1-Ramp Time, 2-Soak, 3-Ramp Rate, 4-Jump, 5-End, 6-Wait For, 7-Inst Change)
PSTime- Read Profile Step Time Remaining
PTTime -Read Profile Total Time Remaining
Pact -Read the Active Profile Name

Internal Alarm Data Topics:

Alm1 -Read Alarm 1 State (Safe, Startup, Alarm)
ALbl1 -Read Alarm 1 Name
Alm2 -Read Alarm 2 State (Safe, Startup, Alarm)
ALbl2 -Read Alarm 2 Name

Process Data Topics:

PV1 – Read Process Value For Control Loop 1

PVLbl1 – Read Process Name for Control Loop 1
SP1 – Read Setpoint Value for Control Loop 1
HPWR1 – Read Heat Power Output for Control Loop 1
CPWR1 – Read Cool Power Output for Control Loop 1

PV2 – Read Process Value for Control Loop 2
PVLbl2 – Read Process Name for Control Loop 2
SP2 – Read Setpoint Value for Control Loop 2
HPWR2 – Read Heat Power Output for Control Loop 2
CPWR2 – Read Cool Power Output for Control Loop 2

PV3 – Read Process Value for Control Loop 3
PVLbl3 – Read Process Name for Control Loop 3
SP3 – Read Setpoint Value for Control Loop 3
HPWR3 – Read Heat Power Output for Control Loop 3
CPWR3 – Read Cool Power Output for Control Loop 3

PV4 – Read Process Value for Control Loop 4
PVLbl4 – Read Process Name for Control Loop 4
SP4 – Read Setpoint Value for Control Loop 4
HPWR4 – Read Heat Power Output for Control Loop 4
CPWR4 – Read Cool Power Output for Control Loop 4

LV1 – Read Limit Process Value for Limit Control 1
Lim1 – Read Limit State for Limit Control 1 (Safe, Alarm)
LLbl1 – Read Limit Name for Limit Control 1

Analog Input Topics:

Ain11 – Read Analog Value for Input 1, Slot 1
AinLbl11 – Read the Name for Analog Input 1, Slot 1

Ain12 – Read Analog Value for Input 2, Slot 1
AinLbl12 – Read the Name for Analog Input 2, Slot 1

Ain13 – Read Analog Value for Input 3, Slot 1
AinLbl13 – Read the Name for Analog Input 3, Slot 1

Ain14 – Read Analog Value for Input 4, Slot 1
AinLbl14 – Read the Name for Analog Input 4, Slot 1

Ain21 – Read Analog Value for Input 1, Slot 2
AinLbl11 – Read the Name for Analog Input 1, Slot 2

Ain22 – Read Analog Value for Input 2, Slot 2
AinLbl12 – Read the Name for Analog Input 2, Slot 2

Ain23 – Read Analog Value for Input 3, Slot 2
AinLbl13 – Read the Name for Analog Input 3, Slot 2

Ain24 – Read Analog Value for Input 4, Slot 2
AinLbl14 – Read the Name for Analog Input 4, Slot 2

Ain31 – Read Analog Value for Input 1, Slot 3

AinLbl11 – Read the Name for Analog Input 1, Slot 3

Ain32 – Read Analog Value for Input 2, Slot 3
AinLbl12 – Read the Name for Analog Input 2, Slot 3

Ain33 – Read Analog Value for Input 3, Slot 3
AinLbl13 – Read the Name for Analog Input 3, Slot 3

Ain34 – Read Analog Value for Input 4, Slot 3
AinLbl14 – Read the Name for Analog Input 4, Slot 3

Ain41 – Read Analog Value for Input 1, Slot 4
AinLbl41 – Read the Name for Analog Input 1, Slot 4

Ain42 – Read Analog Value for Input 2, Slot 4
AinLbl42 – Read the Name for Analog Input 2, Slot 4

Ain43 – Read Analog Value for Input 3, Slot 4
AinLbl43 – Read the Name for Analog Input 3, Slot 4

Ain44 – Read Analog Value for Input 4, Slot 4
AinLbl44 – Read the Name for Analog Input 4, Slot 4

Ain51 – Read Analog Value for Input 1, Slot 5
AinLbl11 – Read the Name for Analog Input 1, Slot 5

Ain52 – Read Analog Value for Input 2, Slot 5
AinLbl12 – Read the Name for Analog Input 2, Slot 5

Ain53 – Read Analog Value for Input 3, Slot 5
AinLbl13 – Read the Name for Analog Input 3, Slot 5

Ain54 – Read Analog Value for Input 4, Slot 5
AinLbl14 – Read the Name for Analog Input 4, Slot 5

Ain61 – Read Analog Value for Input 1, Slot 6
AinLbl61 – Read the Name for Analog Input 1, Slot 6

Ain62 – Read Analog Value for Input 2, Slot 6
AinLbl62 – Read the Name for Analog Input 2, Slot 6

Ain63 – Read Analog Value for Input 3, Slot 6
AinLbl63 – Read the Name for Analog Input 3, Slot 6

Ain64 – Read Analog Value for Input 4, Slot 6
AinLbl64 – Read the Name for Analog Input 4, Slot 6

Event Output Name Topics:

EV1 – Read Event Output 1 Name
EV2– Read Event Output 2 Name
EV3– Read Event Output 3 Name
EV4– Read Event Output 4 Name
EV5– Read Event Output 5 Name
EV6– Read Event Output 6 Name
EV7– Read Event Output 7 Name
EV8– Read Event Output 8 Name

Output Event State Topics:

EVStat1– Read Event Output 1 Status (Off, On)
EVStat2– Read Event Output 2 Status (Off, On)
EVStat3– Read Event Output 3 Status (Off, On)
EVStat4– Read Event Output 4 Status (Off, On)
EVStat5– Read Event Output 5 Status (Off, On)
EVStat6– Read Event Output 6 Status (Off, On)
EVStat7– Read Event Output 7 Status (Off, On)
EVStat8– Read Event Output 8 Status (Off, On)

Batch Topics:

BatchOp – Read Operator Name
BatchUOp– Read Unload Operator Name
BatchP1– Read Product ID 1
BatchP2– Read Product ID 2
BatchB1– Read Batch ID

Profile Names Topics:

P1 – Read Profile 1 Name
P2 – Read Profile 2 Name
P3– Read Profile 3 Name
P4– Read Profile 4 Name
P5– Read Profile 5 Name
P6– Read Profile 6 Name
P7– Read Profile 7 Name
P8– Read Profile 8 Name
P9– Read Profile 9 Name
P10– Read Profile 10 Name
P11– Read Profile 11 Name
P12– Read Profile 12 Name
P13– Read Profile 13 Name
P14– Read Profile 14 Name
P15– Read Profile 15 Name
P16– Read Profile 16 Name
P17– Read Profile 17 Name
P18– Read Profile 18 Name
P19– Read Profile 19 Name
P20– Read Profile 20 Name
P21– Read Profile 21 Name
P22– Read Profile 22 Name
P23– Read Profile 23 Name
P24– Read Profile 24 Name
P25– Read Profile 25 Name
P26– Read Profile 26 Name
P27– Read Profile 27 Name
P28– Read Profile 28 Name
P29– Read Profile 29 Name
P30– Read Profile 30 Name
P31– Read Profile 31 Name
P32– Read Profile 32 Name
P33– Read Profile 33 Name
P34– Read Profile 34 Name
P35– Read Profile 35 Name
P36– Read Profile 36 Name
P37– Read Profile 37 Name

P38– Read Profile 38 Name
P39– Read Profile 39 Name
P40-Read Profile 40 Name

Control Mode Topics:

RCMode1—Read the Control Mode for PID 1 Control/Cascade Loop
RCMode2—Read the Control Mode for PID 2 Control/Cascade Loop
RCMode3—Read the Control Mode for PID 3 Control Loop
RCMode3—Read the Control Mode for PID 4 Control Loop

Below is a table showing how the PID Loop numbers are assigned for an F4T

PID LOOP#	1	2	3	4	6	7	8	9	A	B	C
1	PID 1	PID 1	PID 1	PID 1	CAS 1	CAS 1	CAS 1	CAS 1	CAS 1	CAS 1	CAS 1
2		PID 2	PID 2	PID 2		PID 1	PID 1	PID 1	CAS 2	CAS 2	CAS 2
3			PID 3	PID 3			PID 2	PID 2		PID 1	PID 1
4				PID 4				PID 3			PID 2

*CAS – Cascade PID Loop

*Column Headers are the Control Algorithm Code from the F4T Part Number

F4T-Part Number: F4TXXXXAAXXXXXX

Profile Publish Topics

Publish Communication Parameters:

All payloads should be set to: “True”

Quality of Service (QoS) should be set to 1.

RETAIN flag should be set to “True”.

Example 1: Start the Active Profile for the F4T Controller 1 on the Network:

“F4T/C1/Start”

Payload = True
QoS = 1
Retain Flag = True

Publish Topics:

Start – Start Profile
Stop – Stop Profile
Pause-Pause Profile
Resume – Resume Profile

Res – Read All Data that has changed from the ProcesView Client

CMode1 – Set the Control Mode for PID 1 Control/Cascade Loop
CMode2– Set the Control Mode for PID 2 Control/Cascade Loop
CMode3– Set the Control Mode for PID 3 Control Loop
CMode4– Set the Control Mode for PID 4 Control Loop

Payloads: Auto, Man, Off

Example: Set the Control Mode for PID Control Loop 2 to Manual Mode on Controller 2:

Topic: “F4T/C2/Cmode2
Payload: Man