

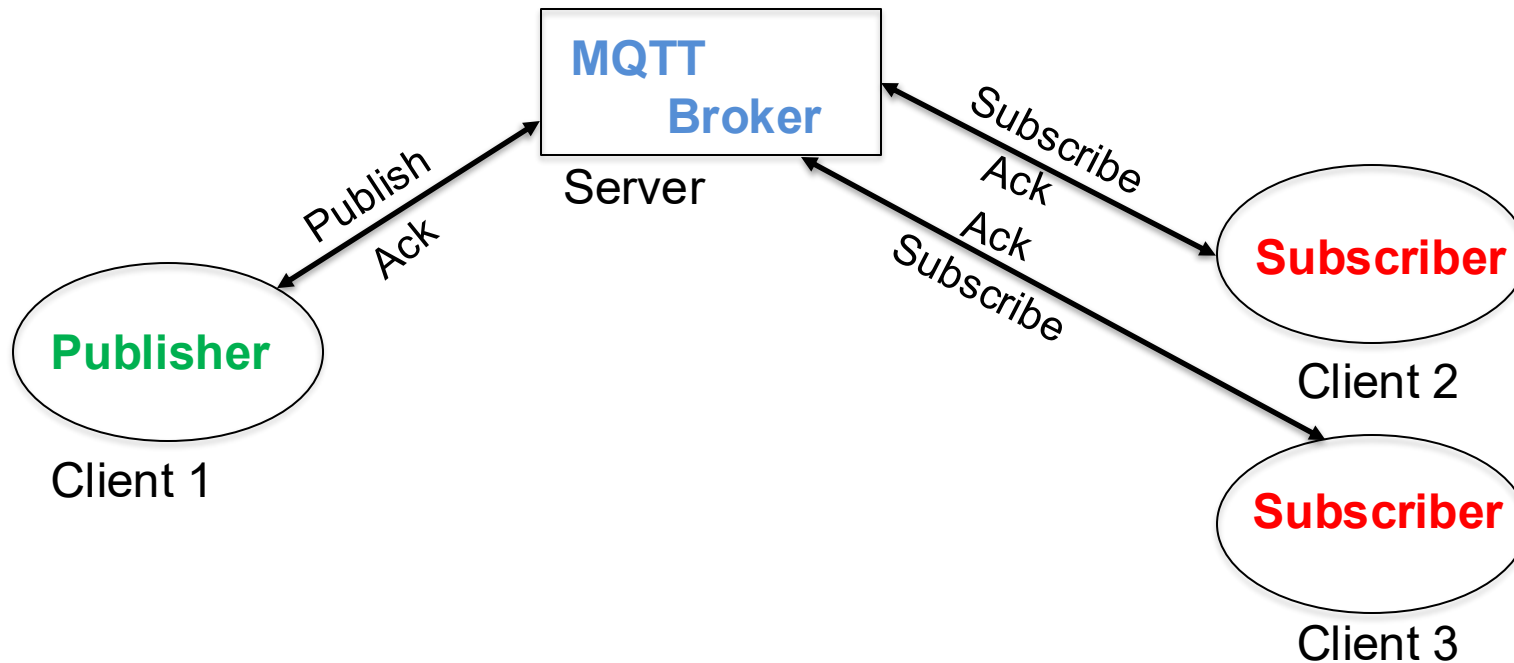


MQTT (Sparkplug) Protocol MQ TELEMETRY TRANSPORT

“Machine-to-Machine” (M2M)/“Internet of Things”
connectivity protocol

MQTT BASICS

- **MQTT** is a Publisher/Subscriber based messaging transport protocol. It is light weight, simple, and designed so as to be easy to implement.

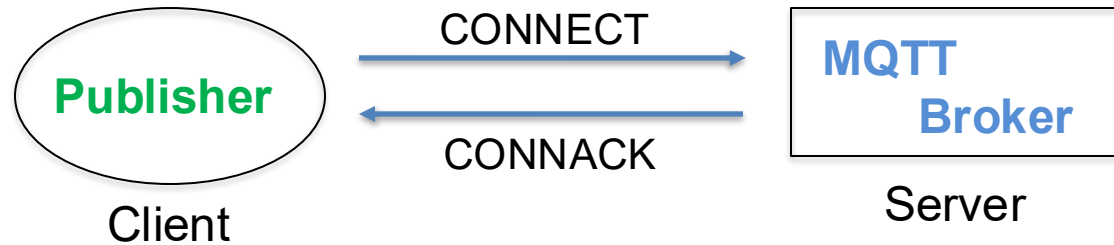


MQTT BASICS

- The publishers and subscribers never contact each other directly.
- The connection between them is handled by a third component (**The Broker**).
- MQTT uses subject-based filtering of messages. Every message contains a **topic** (subject) that the broker can use to determine whether a subscribing client gets the message or not.
 - Individual Topic
 - E.g. myfactory/sector1/motor1_speed , myfactory/sector1/temperature
 - Wild Card Topic (*Subscribers Only*)
 - E.g. myfactory/# , myfactory/sector1/#
 - myfactory/+ /motor1_speed , myfactory/+ /temperature

MQTT BASICS

- To initiate a connection, the client sends a CONNECT message to the broker. The broker responds with a CONNACK message.

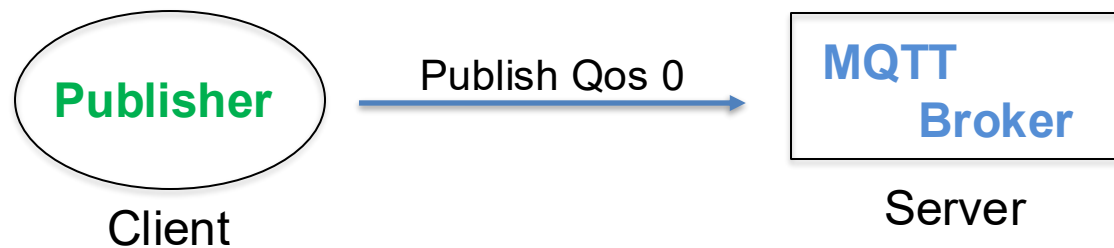


- The client identifier (ClientID) identifies each MQTT client that connects to an MQTT broker.
- The broker uses the ClientID to identify the client and the current state of the client.
- Therefore, this ID should be unique for client per broker.

MQTT BASICS

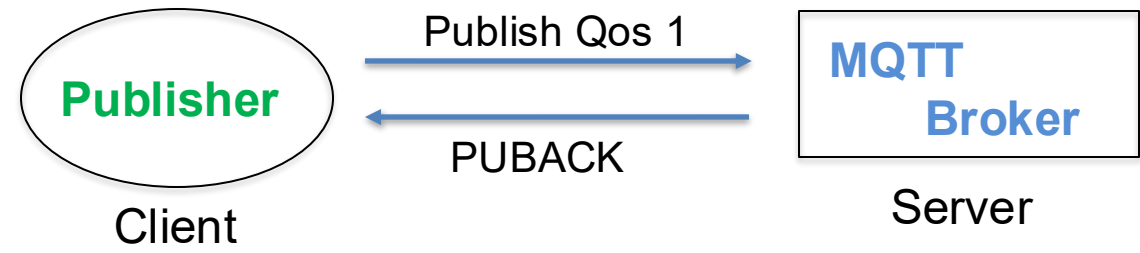
- The **Quality of Service** (QoS) level is an agreement between the sender of a message and the receiver of a message that defines the guarantee of delivery for a specific message.
- There are 3 QoS levels in MQTT

At most once (0)

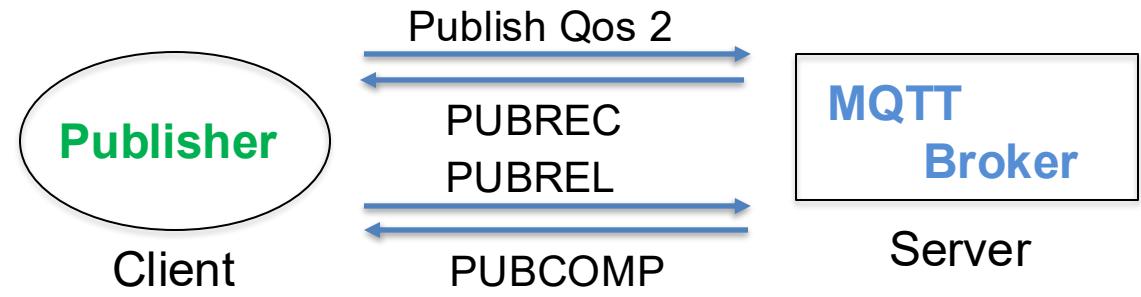


MQTT BASICS

At least once (1)



Exactly once (2)



MQTT BASICS

- When a client connects to a broker, it may inform the broker that it has a Last will.
- This is a message that it wishes the broker to send when the client disconnects unexpectedly.
- The broker will store the message until it detects that the client has disconnected ungracefully.
- If the client disconnect abruptly, the broker sends the message to all subscribed client on the topic, which was specified in the last will message.
- The stored LW message will be discarded if a client disconnects gracefully by sending a DISCONNECT message

MQTT + SparkPlug B

- For MQTT to be interoperable with the IIOT sector some implementation information must be defined. This is what Sparkplug tries to achieve in order to optimise MQTT for Scada/IIOT.
- Below are 3 implementation details that Sparkplug seeks to address:
 - ❖ MQTT Topic Namespace
 - ❖ MQTT State Management
 - ❖ MQTT Payload

MQTT + SparkPlug B

- **MQTT TopicNamespace**

The topicnamespace needs to be understood by every client participating in the data exchange. All MQTT clients using Sparkplug Specification will use the following structure:

namespace/group_id/message_type/edge_node_id/[device id]

Namespace is the root element, group_id element is logical grouping of edge of network nodes, message_type specifies the type of message being sent, edge_node_id uniquely identifies the MQTT edge of network node, device_id is optional and could be a sensor or devices behind a node which could be identified.

MQTT + SparkPlug B

- **MQTT State Management**

In industrial systems the connection state of the network is extremely important, there has to be continuous polling to know all devices are online and can provide data.

Sparkplug specification defines the use of “Last Will and Testament” that enables the broker to provide information to clients when a publisher goes offline(Death Certificate) or when its connection is restored(Birth Certificate).

- **MQTT Payload**

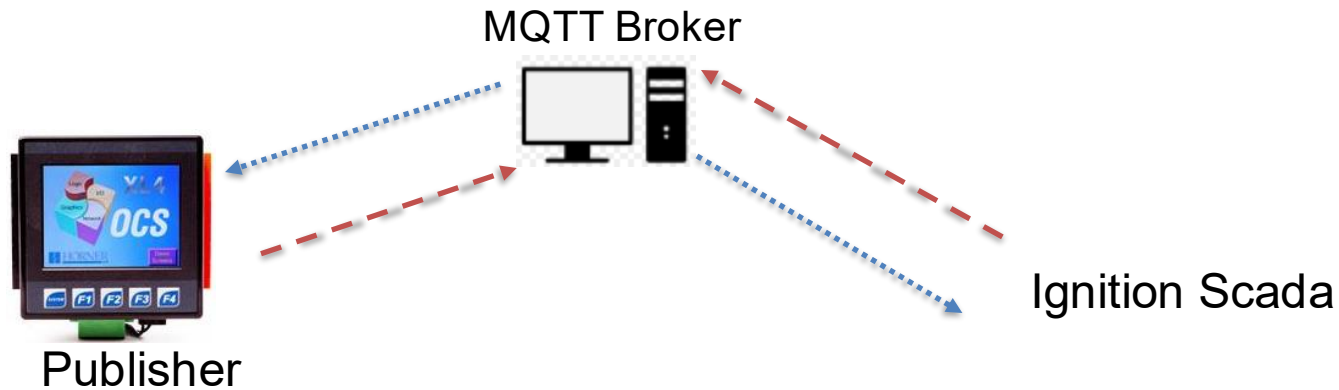
SparkplugB Payload is encoded using Google Protocol Buffers technology. It supports Complex data types, Datasets, Historical data, File data and metrics.

Horner MQTT SparkplugB Demo

- Cscape Configuration for MQTT Sparkplug protocol.
- Horner OCS supports MQTT Sparkplug client and is both Publisher/Subscriber.
- Publisher can publish data either periodically, by trigger, change of state or change of value.
- Licensing for MQTT Sparkplug (using both registered and un-registered Cscape).

Horner MQTT SparkplugB Demo

- **XL4e** and Ignition Scada are the MQTT Clients and the MQTT broker is Mosquitto.



- - - Data Send to Broker
..... Data Receive from Broker

Licensing of MQTT Sparkplug

- There is only a single permanent license available for MQTT.
- The licensing procedure varies a bit depending on whether the user is using a Registered or Un-registered version of Cscape.
- For registered version the user can directly apply the voucher code in Cscape itself and activate the license.
- For un-registered version the user is re-directed to the licensing website where the voucher can be redeemed.

Thank you..