NOVEMBER 15, 2022

Lessons Learned from Implementing MQTT

With OSIsoft Pi and AVEVA Historian

Xavier Mesrobian

AVEVA

Xavier Mesrobian

Vice President Sales & Marketing

Skkynet

xavier.mesrobian@skkynet.com

Agenda

What is MQTT

Advantages / Disadvantages

Lessons Learned from Implementing MQTT

Other Implementation Examples

AVEVA

About Skkynet

Critical Infrastructure for over 25 years

- 25,000+ installations in 86 countries
- Used by the 10 top automation providers worldwide
- OEM relationships with hardware and software providers
- Certified AVEVA Technology Partner
- Covering energy, discreet manufacturing, water/wastewater, building automation, oil & gas, and minerals & mining to name just a few

What is MQTT

Understanding

- MQTT is an OASIS standard network messaging protocol for the Internet of Things (IoT)
  - Publish/subscribe messaging transport
  - Lightweight and efficient
  - Bi-directional communications
  - Offers three defined quality of service levels (0, 1, 2)
  - MQTT is a transport, not a protocol
- Most common flavors of MQTT
  - MQTT JSON
  - MQTT Sparkplug B (predictable payload format)
  - MQTT JSON Schema (unpredictable payload format)

What is MQTT

Advantages of using MQTT

- Publish/subscribe messaging transport
  - The MQTT client/EoN establishes an outbound connection
- Typically uses less bandwidth, unless there is a disconnect
- Scalable
- Typically, well suited for low power devices - Battery friendly

What is MQTT

Disadvantages

- MQTT is a messaging protocol, not a data protocol
- MQTT does not offer a standard data protocol
- The broker cannot perform intelligent routing
- Does not maintain message order
- No guaranteed consistency
- Daisy chaining is not reliable, QoS does not chain
- It is difficult to restrict access on topics, read-only, or data subsets
- QoS can be misleading

Water Wastewater

Global Engineering and Construction

CDM Smith

Northeast
Ohio Regional
Sewer District

Challenge

Replace existing system of communication to AVEVA Galaxy to support store and forward, while maintaining the Galaxy configuration templates.

New system needs to be cost effective.

Solution

Implement MQTT clients, which support store and forward. Enhancing and replacing Ethernet/IP.

Benefits

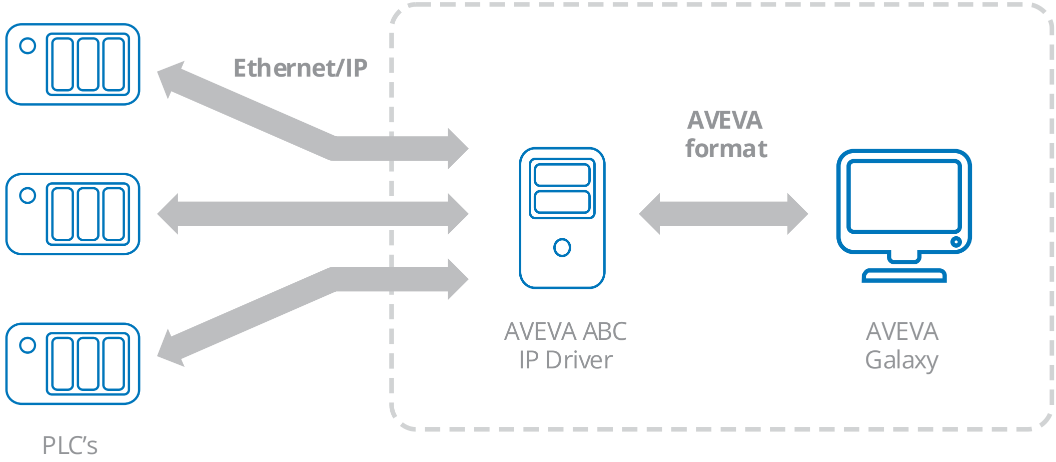
Cost effective, scalable solution. No changes to the Galaxy configuration templates. More accurate reporting.

AVEVA

Lessons learned from implementing MQTT

CD Smith MQTT Journey – Current design

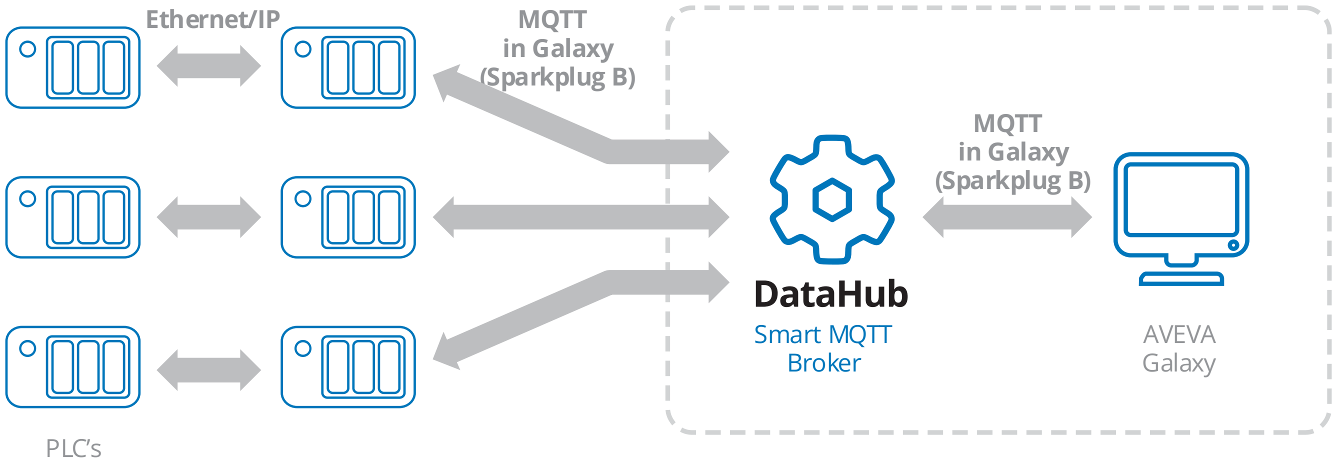
- PLC's are connected via Ethernet/IP
- If the connections fails, data is lost as there is no facility to transfer historical data



Lessons learned from implementing MQTT

CD Smith MQTT Journey – MQTT Sparkplug B Trial

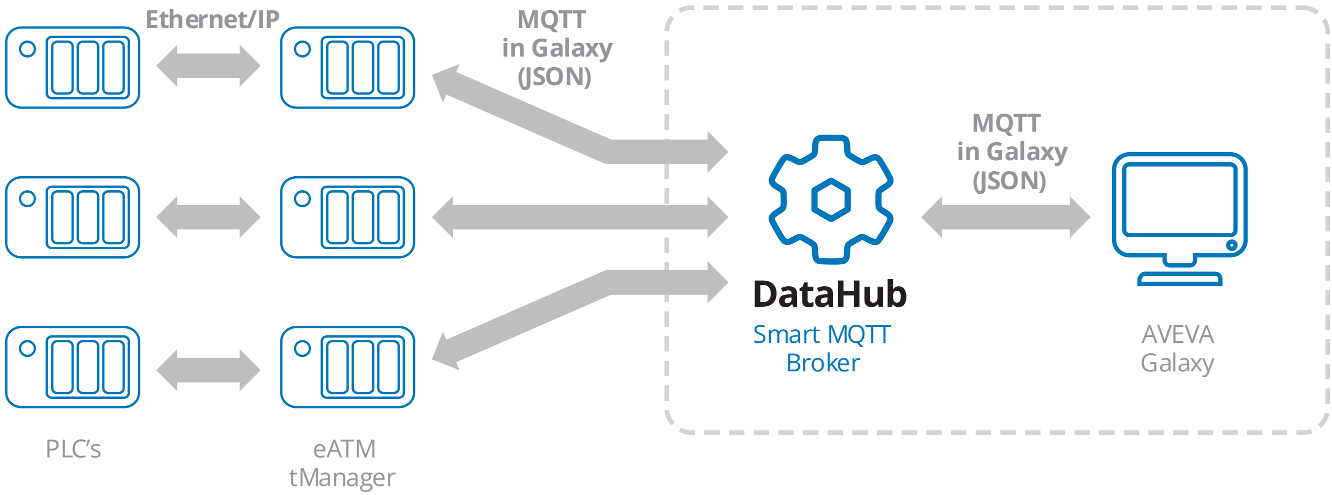
- Conceptually this should have worked
- Real-time data worked fine
- Historical data did not work



Lessons learned from implementing MQTT

CD Smith MQTT Journey – MQTT JSON

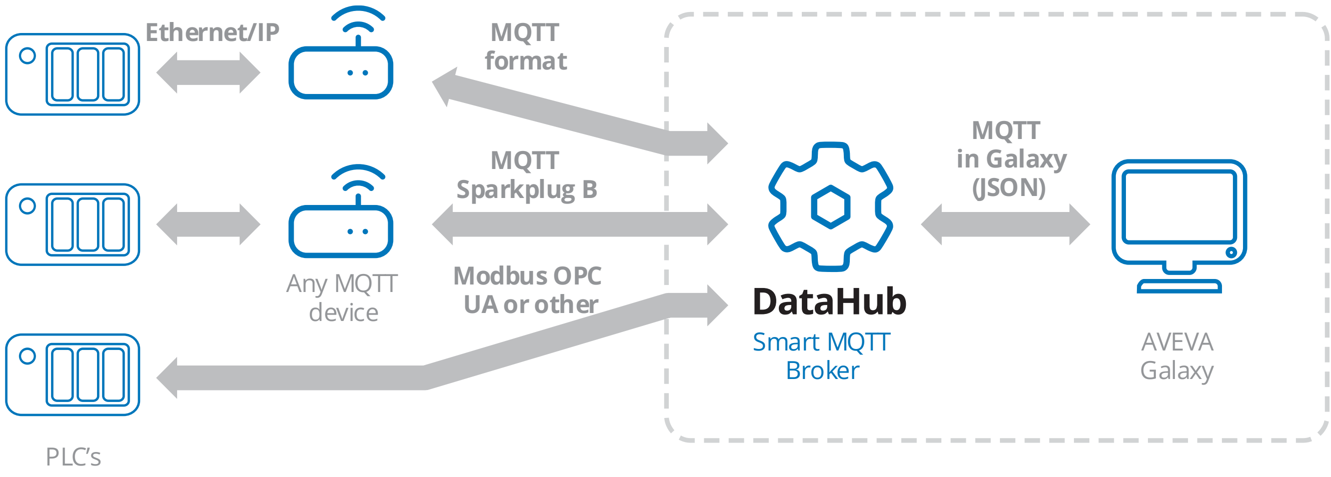
- Galaxy expects data to arrive in a specific JSON format in order to ingest historical MQTT data
- Real-time data worked as expected
- Historical data also worked as expected
- Disadvantage of this approach is that your choice of MQTT clients are limited as the client must publish in a specific JSON format.



Lessons learned from implementing MQTT

CD Smith MQTT Journey – MQTT with DataHub® Smart Broker

- DataHub supports any number of client MQTT connection types
  - MQTT JSON(S)
  - MQTT Sparkplug B
  - MQTT JSON Schema
- Conversion is done at the Broker
  - Any MQTT client can connect
  - Store and Forward enabled
  - Cost Effective



> “One of the things with DataHub that I've been impressed throughout the pilot process is that it just plain works. We tested it to its max. I've been really impressed, and it takes a lot for me to say that”

Jason Davis – Automation Engineer

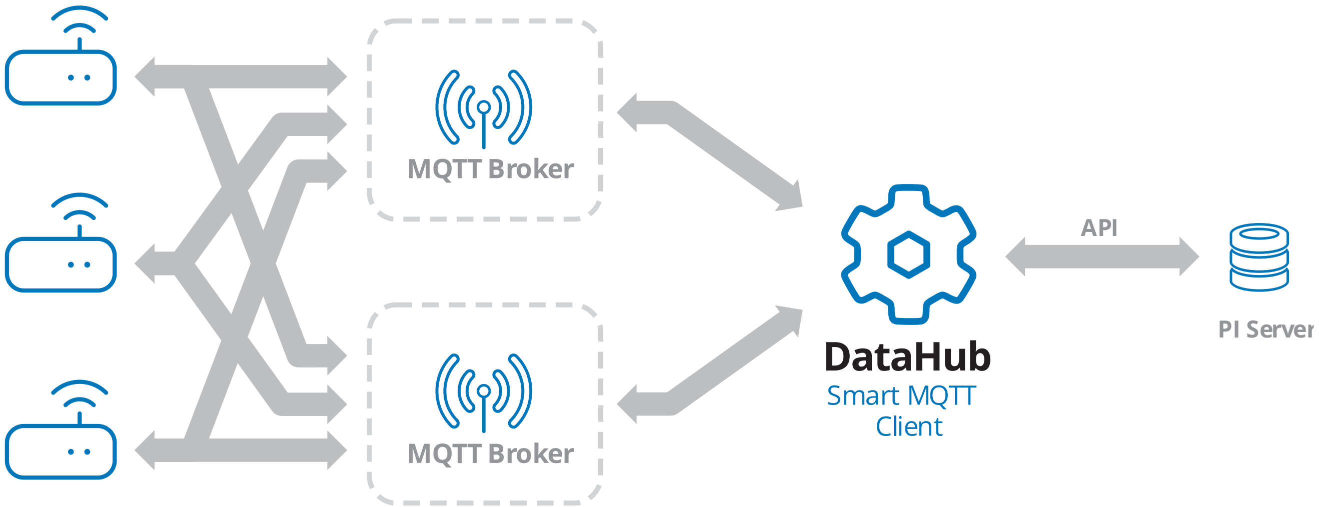
MQTT Implementation With Redundancy

Use Case Worth mentioning

Implementing MQTT Redundancy for Brokers

MQTT Client Redundancy

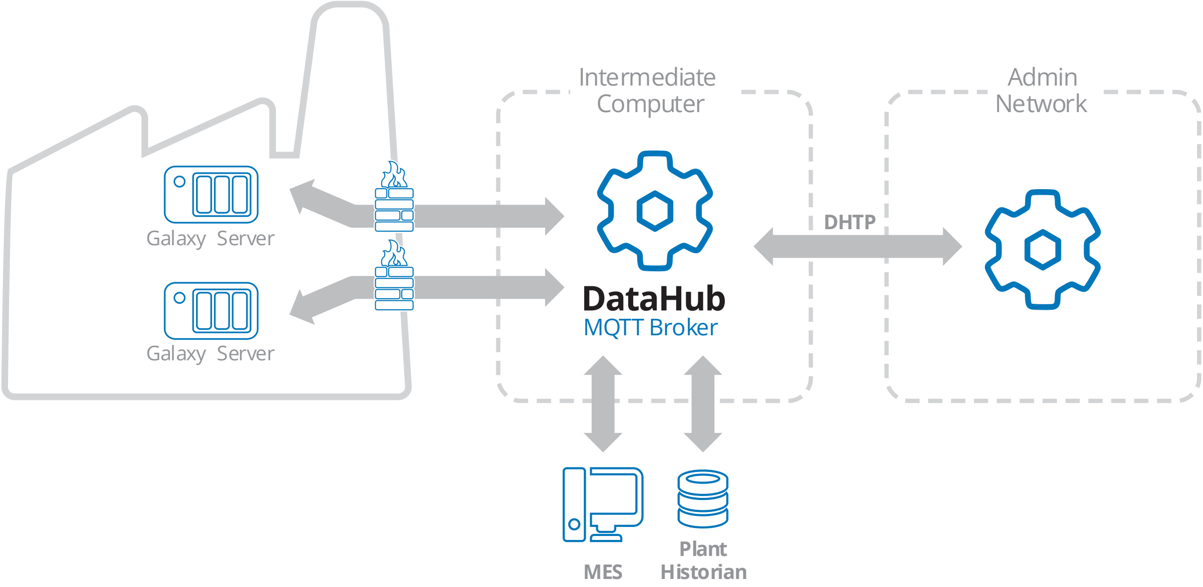
- Here we have multiple devices connected to two MQTT brokers
- DataHub is connecting to both
- Resolving Redundancy
- Publishing directly to OSIsoft Pi



Implementing MQTT Redundancy for Clients

MQTT Client Redundancy

- Here we have two Galaxy Servers publishing MQTT to the DataHub
- DataHub is managing redundancy across the two data sources
- Collecting MES and plant Historical data
- Mirroring the data to the Admin network



Questions?

Please wait for the microphone.

State your name and company.

Please remember to...

Navigate to this session in the mobile app to complete the survey.

Thank you!