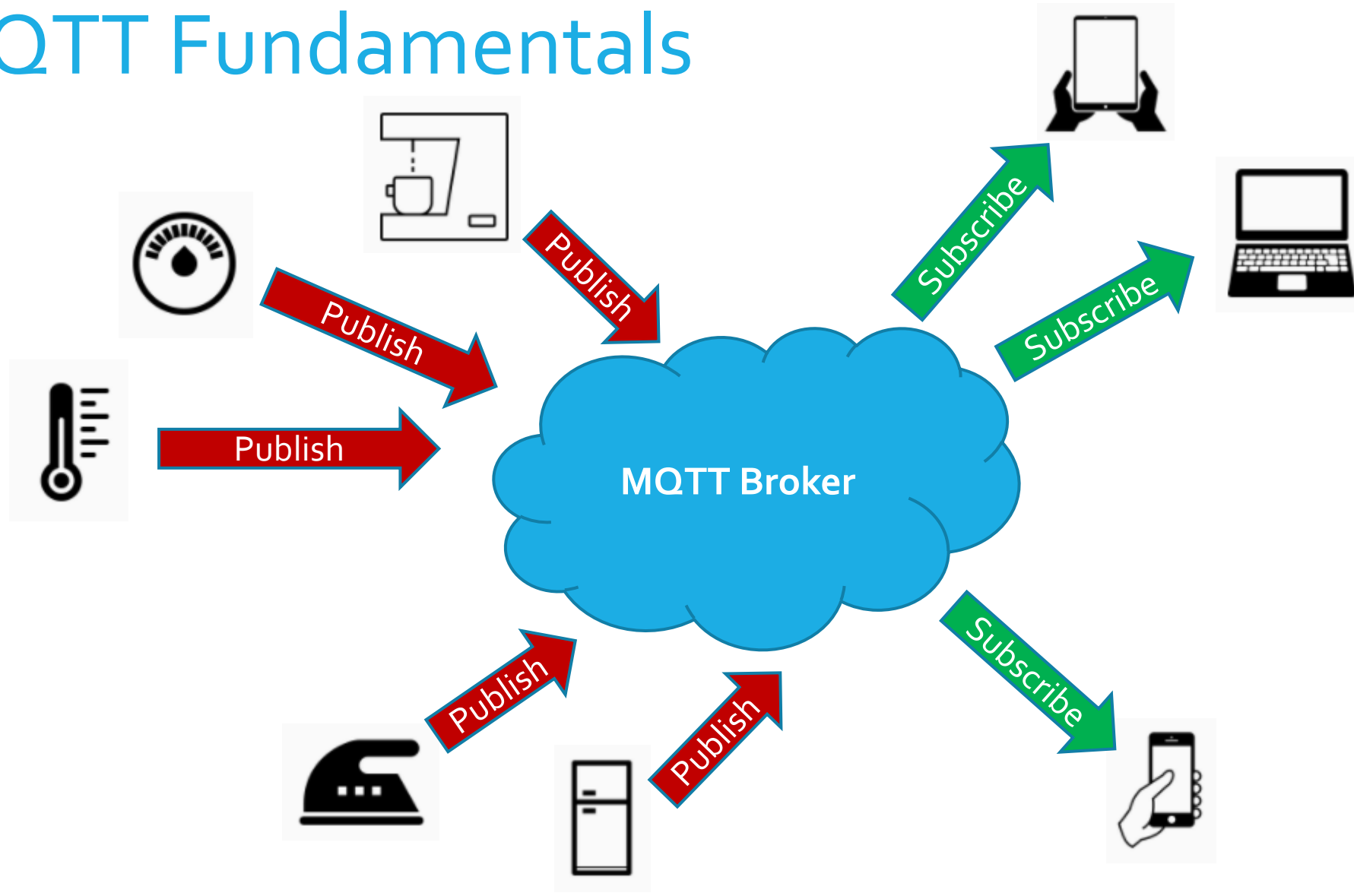


Analysing

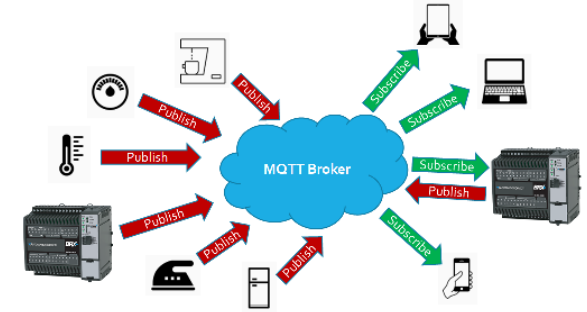
MQTT (IOT)

Message Queuing Telemetry Transport
(Internet of Things)

MQTT Fundamentals



MQTT Terminology (1 of 2)



- **MQTT Broker**

- Receives published topics
- Distributes topics to subscribers
- Keeps Client connections alive
- Sends Last Will & Testament (LWT) to subscribers if a Client “ungracefully disconnects”

- **MQTT Client**

- Can publish topic(s), keep-alive time, Retain bit, QoS, Last Will & Testament
- Can subscribe to topic(s)

- **Topic**

- Name of the data

- **Payload**

- Actual data

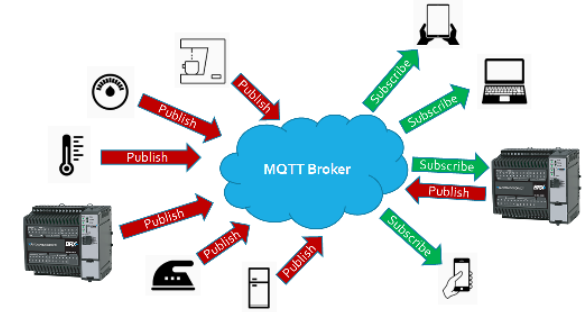
- **Message**

- Topic + Payload

- **QoS (Quality of Service)**

- 0 = At most once : transmits message once (relies on TCP)
- 1 = At least once : transmits message until it is acknowledged by receiver (may receive more than one)
- 2 = Exactly once : transmits message, needs “received” message, asks if it can be “released,” needs “complete” message

MQTT Terminology (2 of 2)



- **Publish**

- To send a Topic w/Payload to MQTT Broker

- **Subscribe**

- To request a Topic w/Payload update from MQTT Broker

- **Retain**

- Asks MQTT Broker to save the Topic w/Payload even after sending it to all the subscribing Clients

- **Keep-alive Time**

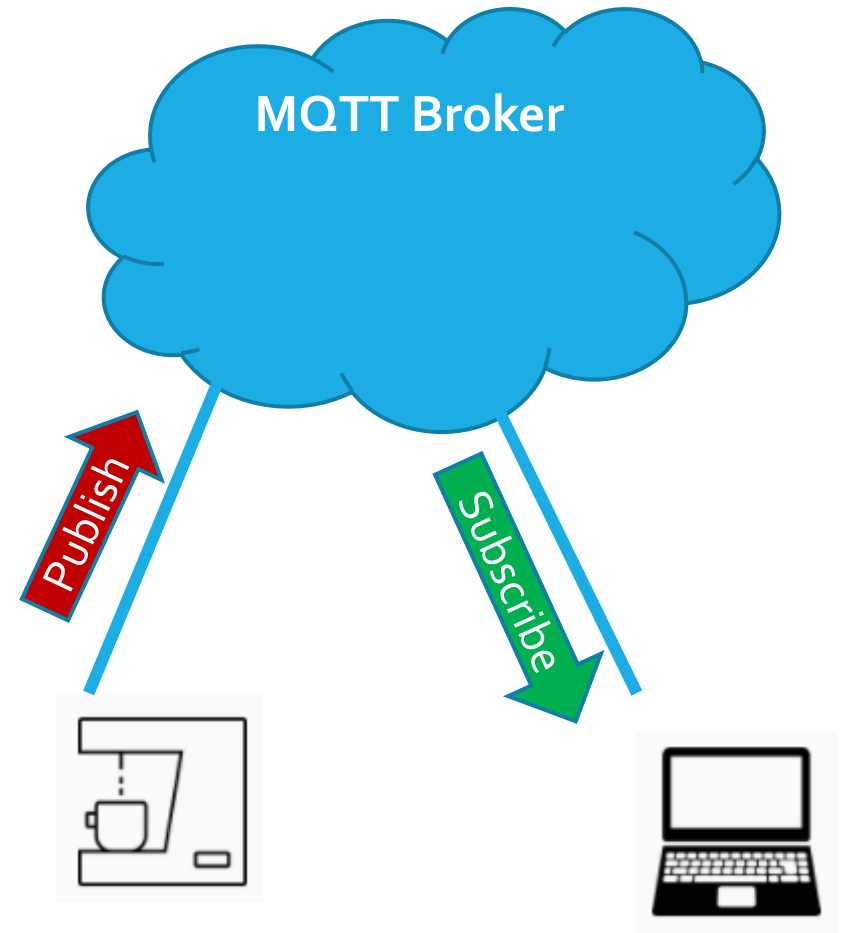
- How often Broker “pings” client to see if he’s there

- **Last Will & Testament (LWT)**

- Topic w/Payload initially sent by an MQTT Client to the MQTT Broker for the Broker to send to other Clients if he is “ungracefully disconnected”

MQTT Data Exchange

- Publishers are fundamentally separate from Subscribers
 - Publishers only care about getting data to Broker
 - Broker is fully responsible for getting data to Subscribers
- Clients connect to an MQTT Broker (TCP/IP, MQTT)
- Clients can publish data to topics, e.g.
 - `host/office/greg/temp, 72.3`
- Clients subscribe to topics, e.g.
 - `host/office/greg/temp`



DEMO
