

Everything you need to know: MQTT

Message

Queue

Telemetry

Transport



General Idea

- Client – Server
- publish/subscribe protocol
- Lightweight
- Open
- Simple
- Easy to Implement



Best for:

Machine to Machine
(M2M)

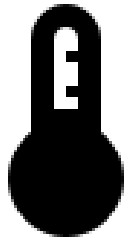
Internet of Things
(IoT)

Short History

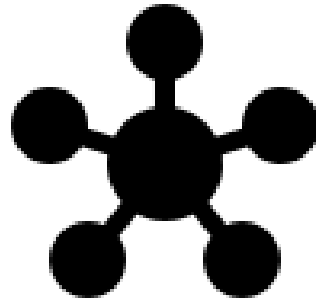
- Andy Stanford-Clark (IBM) and Arlen Nipper (Arcom Controls Systems Inc.)
- Pipeline (oil and gas) control and monitoring
 - Many proprietary protocols and systems
 - No intercommunication
- Idea of MQTT was born
- Still in use: MQTT-SN
- Widely adopted: MQTT 3.1.1
- Newest standard: MQTT 5.0

So what is MQTT?

General Principle – Basic Scenario



Client A



MQTT Broker

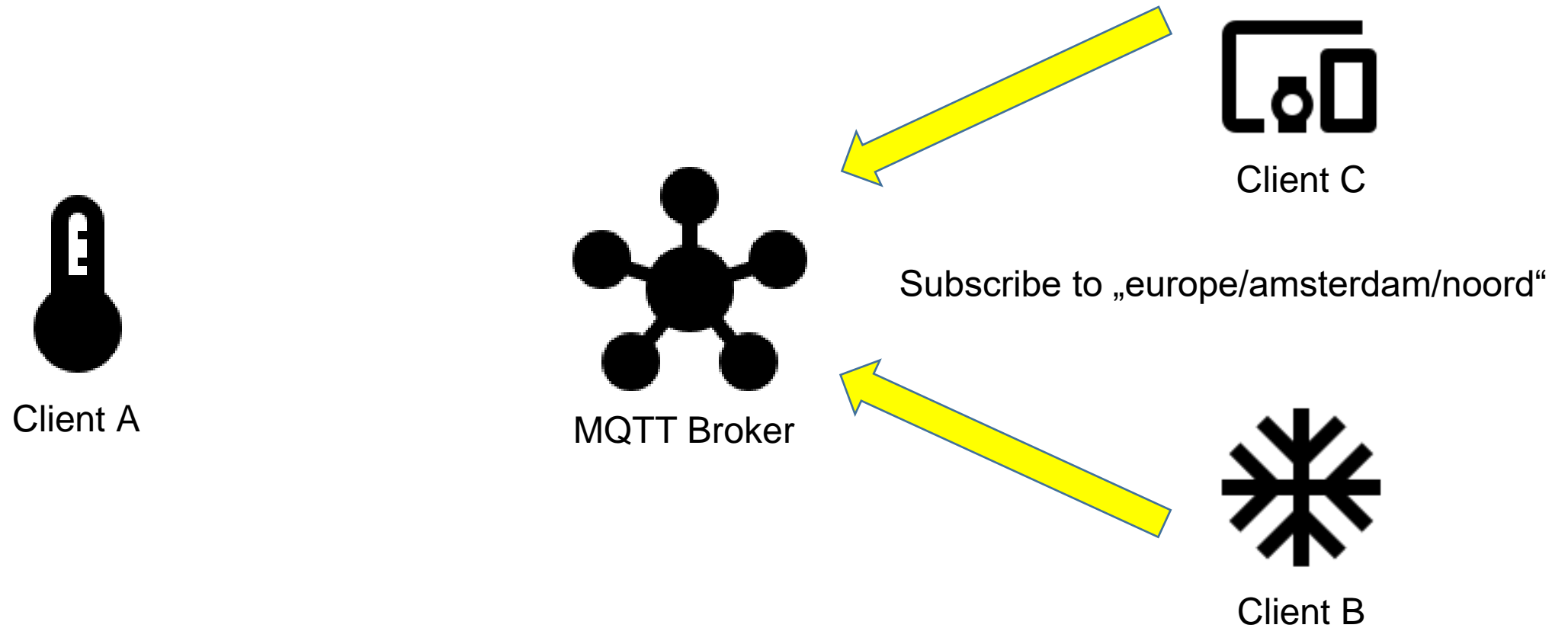


Client C

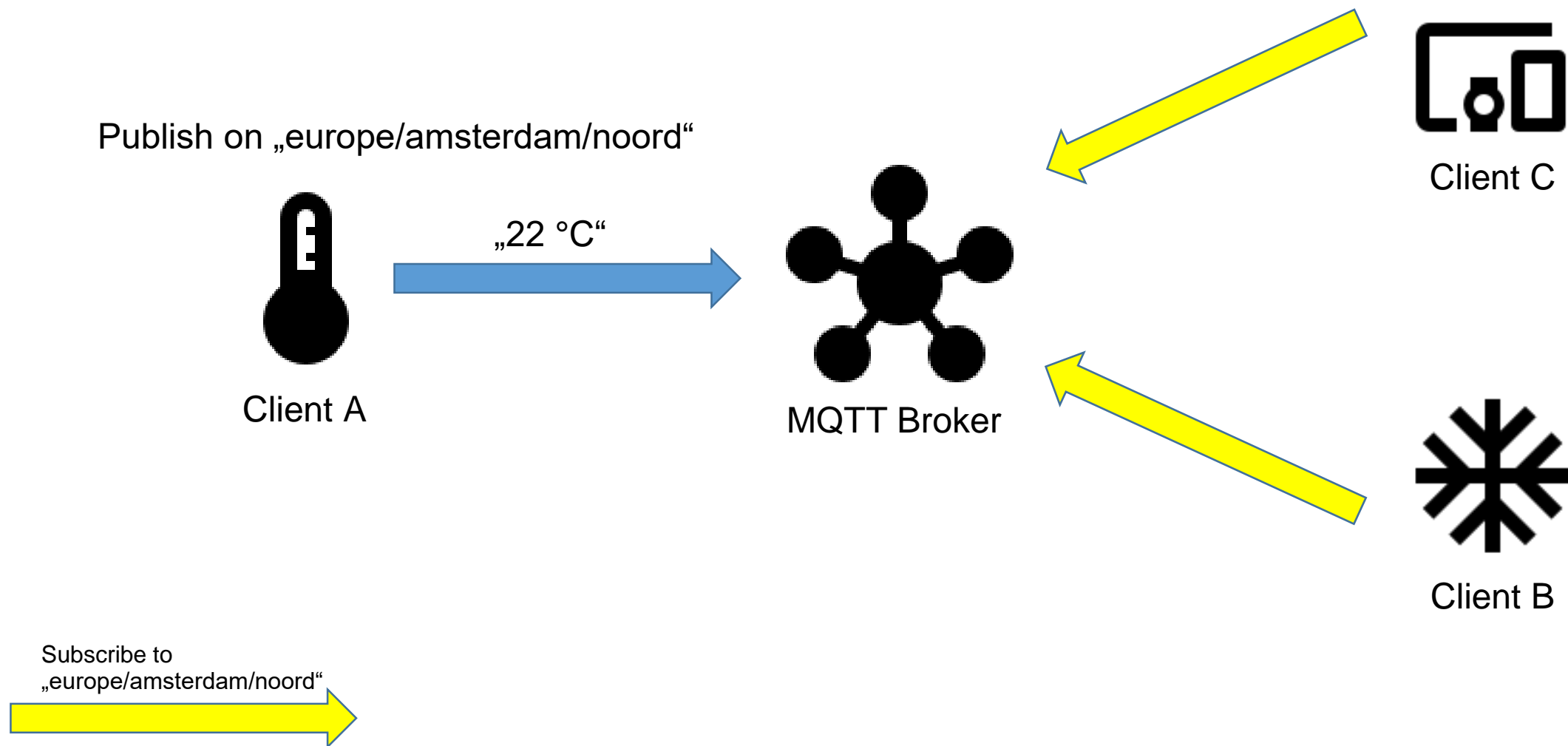


Client B

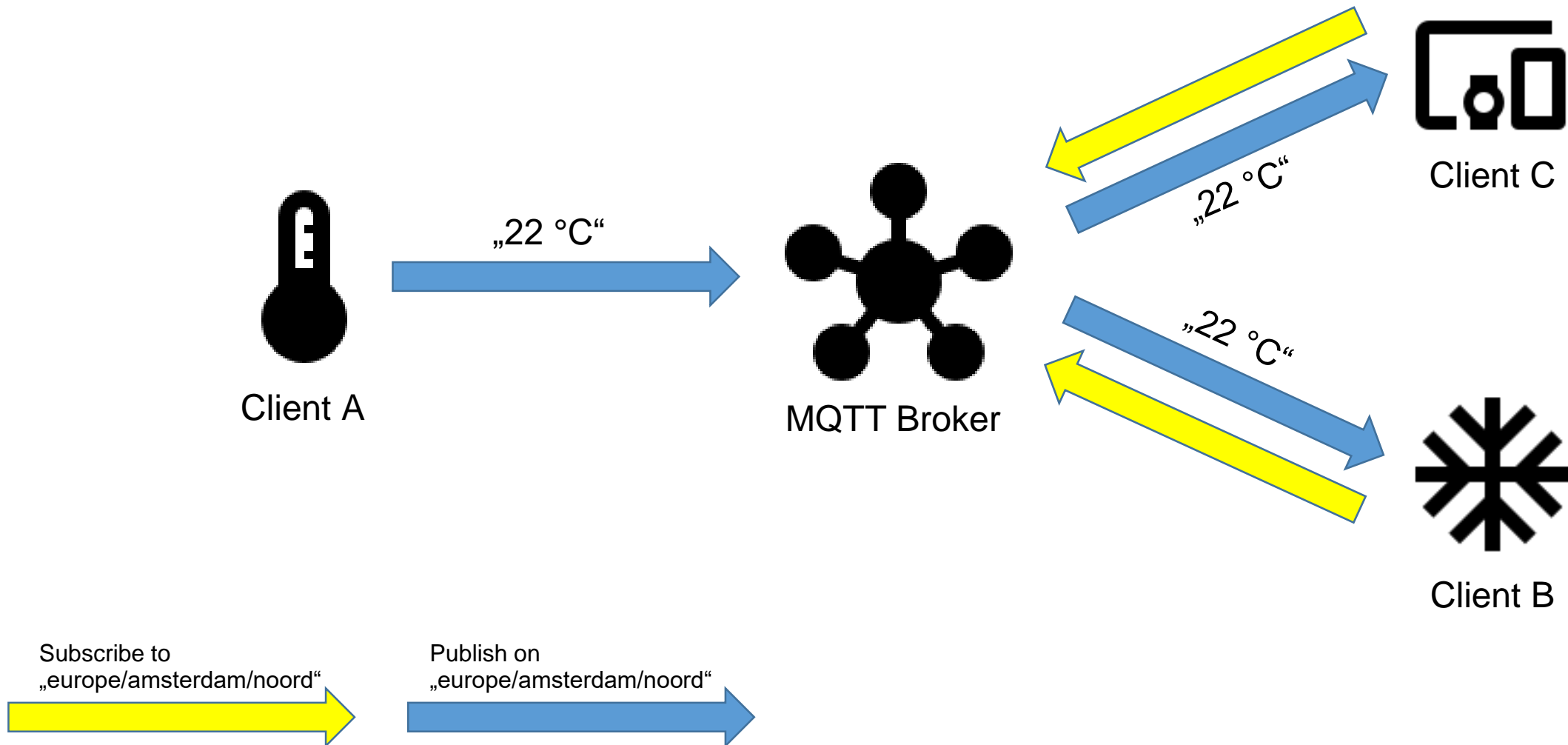
General Principle – Basic Scenario



General Principle – Basic Scenario



General Principle – Basic Scenario



General Principle - Topics

europa/amsterdam/noord

europa/amsterdam/oost

europa/amsterdam

north-america/usa/washington

General Principle - Topics

- Case-Sensitive
 - europe/amsterdam/noord ≠ Europe/Amsterdam/Noord
- Good practices:
 - No „/“ at the beginning or end
 - No „ “ (spaces)
 - No special characters – ä ö ü § % }
 - No „\$“ at the beginning

General Principle - Topics

europa/amsterdam/noord/temperatuur

europa/amsterdam/noord/humidity

europa/amsterdam/oost/temperatuur

europa/amsterdam/oost/humidity

General Principle - Topics - Wildcards

- Multi-Level Wildcard „#“
 - Matches any number of levels
 - MUST be last character!
 - Can only be used once
- Single-Level Wildcard „+“
 - Matches one specific level
 - Can be used at any level
 - Can be used more than once

General Principle - Topics - Wildcards

- Multi-Level Wildcard: „europe/amsterdam/noord/#“
- Matches:
 - „europe/amsterdam/noord/temperature“
 - „europe/amsterdam/noord/temperature/c“
 - „europe/amsterdam/noord/humidity“
 - „europe/amsterdam/noord“

General Principle - Topics - Wildcards

- Single-Level Wildcard: „europe/amsterdam/+ /temperature“
- Matches:
 - „europe/amsterdam/noord/temperature“
 - „europe/amsterdam/oost/temperature“
 - „europe/amsterdam//temperature“
- Doesn't match:
 - „europe/amsterdam/noord/temperature/c“
 - „europe/amsterdam/noord/humidity“

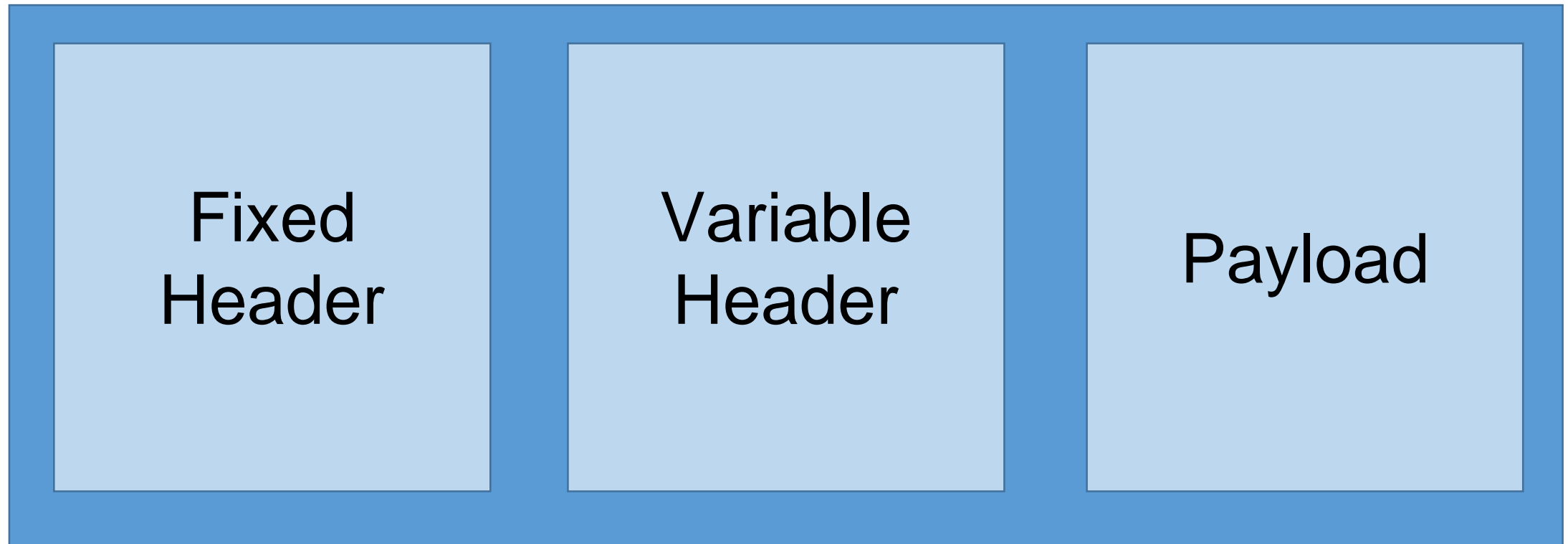
What are we actually
sending?

Details - Packets

- 15 different packet types
- 6 bigger groups
- Very simple
- Only small overhead
- Fixed and variable part

Details - Packets

- MQTT Control Packet 3 Parts:



Details - Packets - Fixed Header

Bit	7	6	5	4	3	2	1	0
Byte 1	Control Packet Type				Flags			
Byte 2...	Remaining Length							

Details - Packets - Variable Header

- Only used by some packet types
- 2 bytes used for packet identifier
- Variable length for properties, for example:
 - Topic Alias
 - Authentication Data
 - Maximum Packet Size

Details - Packets - Payload

- Required by some packets
 - Connect
 - Userdata
 - Last will
 - Subscribe
 - List of topics
 - Unsubscribe
 - List of topics

Details - Packets

Connection	Publish	Subscribe	Unsubscribe	Ping	Auth
CONNECT CONNACK DISCONNECT	PUBLISH PUBACK PUBREC PUBREL PUBCOMP	SUBSCRIBE SUBACK	UNSUBSCRIBE UNSUBACK	PINGREQ PINGRESP	AUTH

- ACK = Acknowledgment
- REC = Received
- REL = Released
- COMP = Completed
- REQ = Request
- RESP = Response

Details - QoS

- QoS = Quality of Service

QoS 0	QoS 1	QoS 2

Details - QoS

- QoS = Quality of Service

QoS 0	QoS 1	QoS 2
At most once delivery (fire and forget)		
Lowest priority		
Best for cumulative readings		

Details - QoS

- QoS = Quality of Service

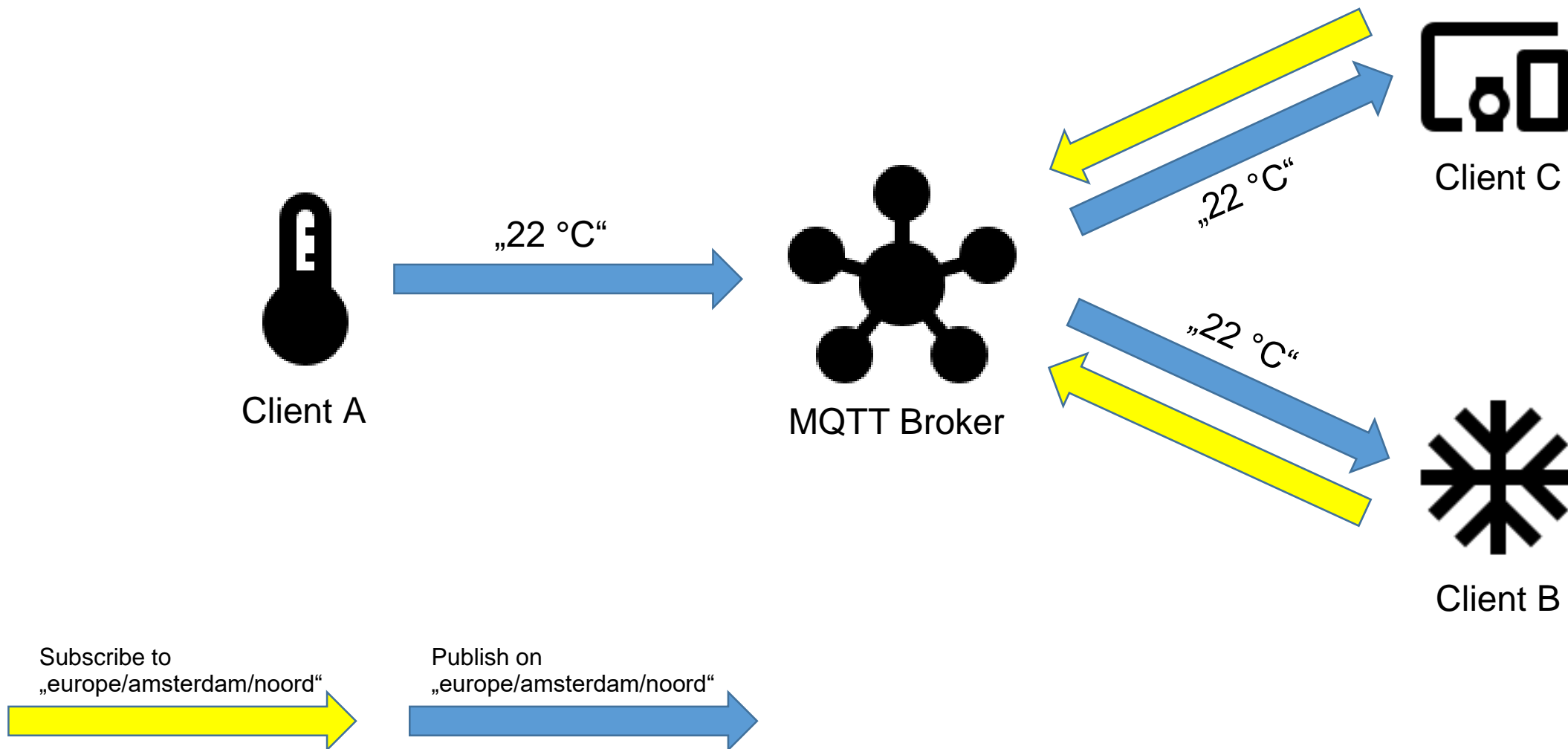
QoS 0	QoS 1	QoS 2
At most once delivery (fire and forget)	At least once delivery	
Lowest priority	Medium Priority	
Best for cumulative readings	Best if reliable power at client	

Details - QoS

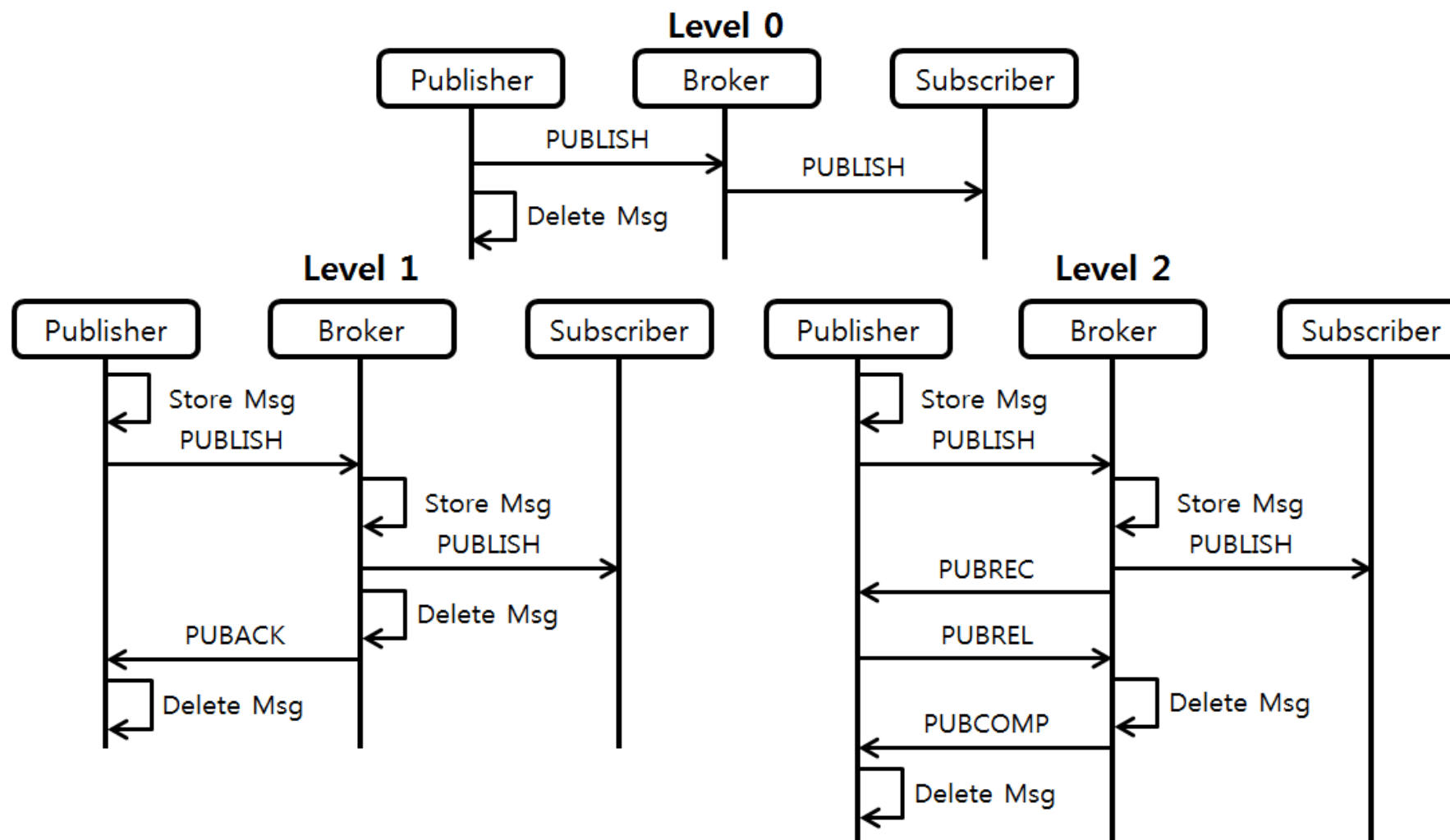
- QoS = Quality of Service

QoS 0	QoS 1	QoS 2
At most once delivery (fire and forget)	At least once delivery	Exactly once delivery
Lowest priority	Medium priority	High priority
Best for cumulative readings	Best if reliable power at client	No loss or duplicate acceptable

Details - QoS



Details - QoS



Details - Last Will

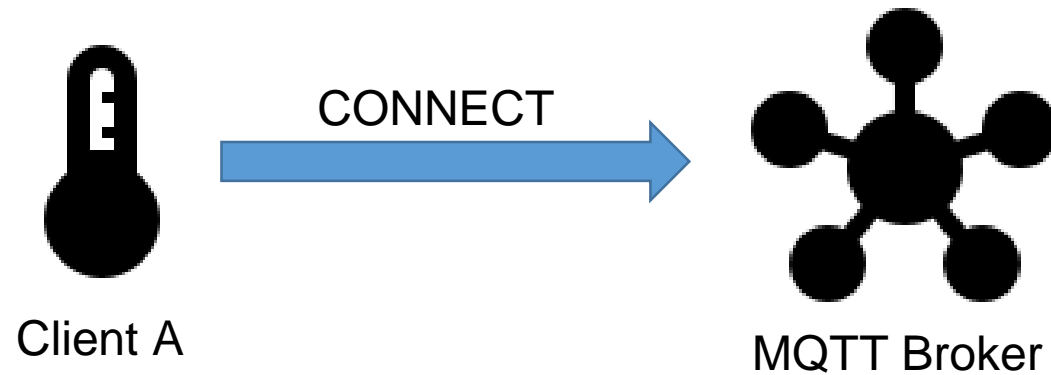
- Testament
- On CONNECT
 - topic, payload, QoS
- Two disconnect option:
 - graceful
 - Sending DISCONNECT packet
 - not graceful (by spec)
 - An I/O error or network failure detected by the Server
 - The Client fails to communicate within the Keep Alive time (*more on that later*)
 - The Client closes the Network Connection without first sending a DISCONNECT packet with a Reason Code 0x00 (Normal disconnection)
 - The Server closes the Network Connection without first receiving a DISCONNECT packet with a Reason Code 0x00 (Normal disconnection)

Details - Retained Messages

- Flag
- Broker saves Message as is
- Only the newest message per topic
- Gets send on subscription

→ last known good value

Details - LWT + Retained Message

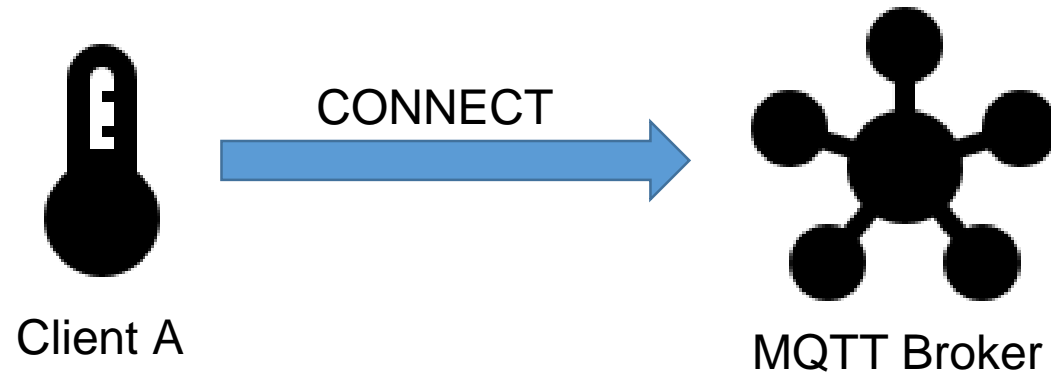


Details - LWT + Retained Message

Broker-Storage:

LWT:

- status/client-A
- „Offline“
- Retained = true

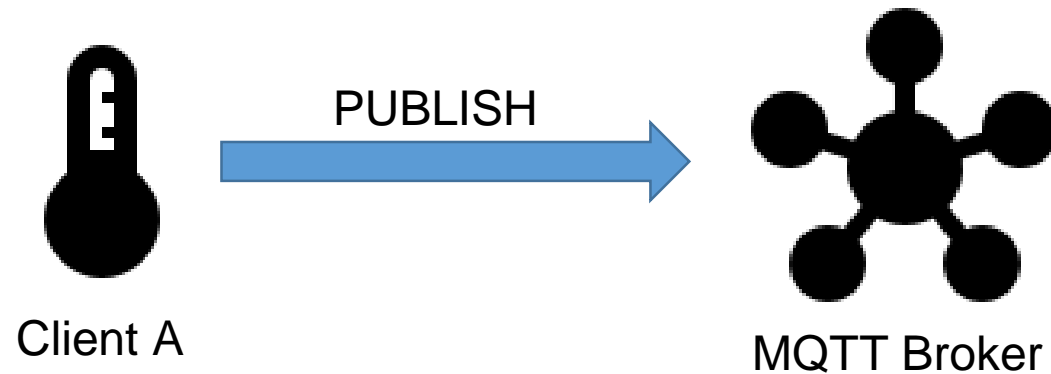


Details - LWT + Retained Message

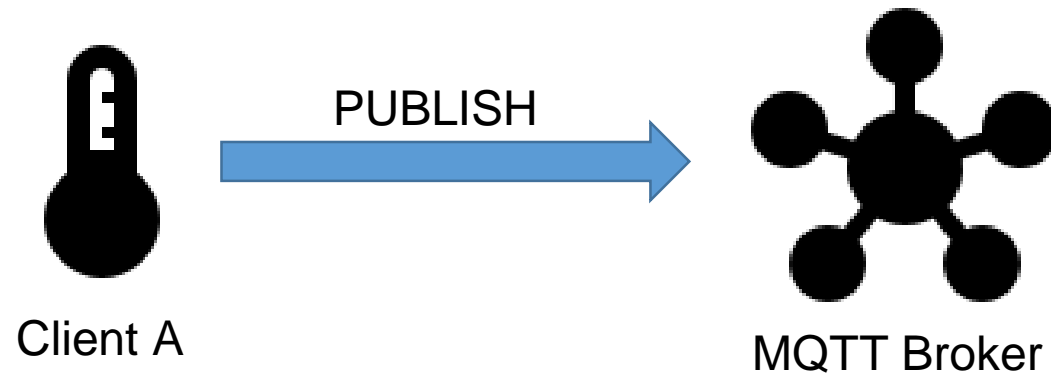
Broker-Storage:

LWT:

- status/client-A
- „Offline“
- Retained = true



Details - LWT + Retained Message



Broker-Storage:

LWT:

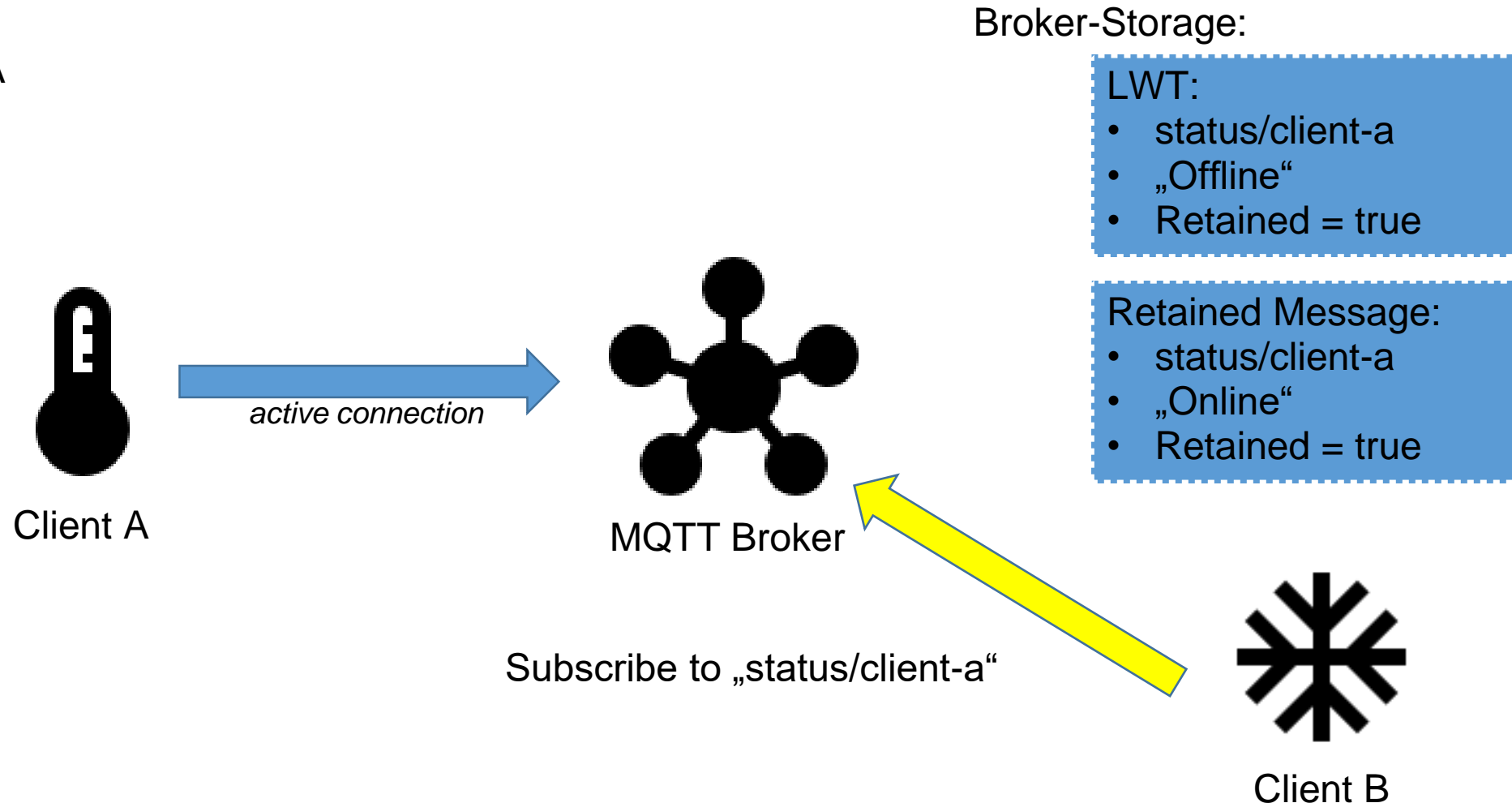
- status/client-A
- „Offline“
- Retained = true

Retained Message:

- status/client-A
- „Online“
- Retained = true

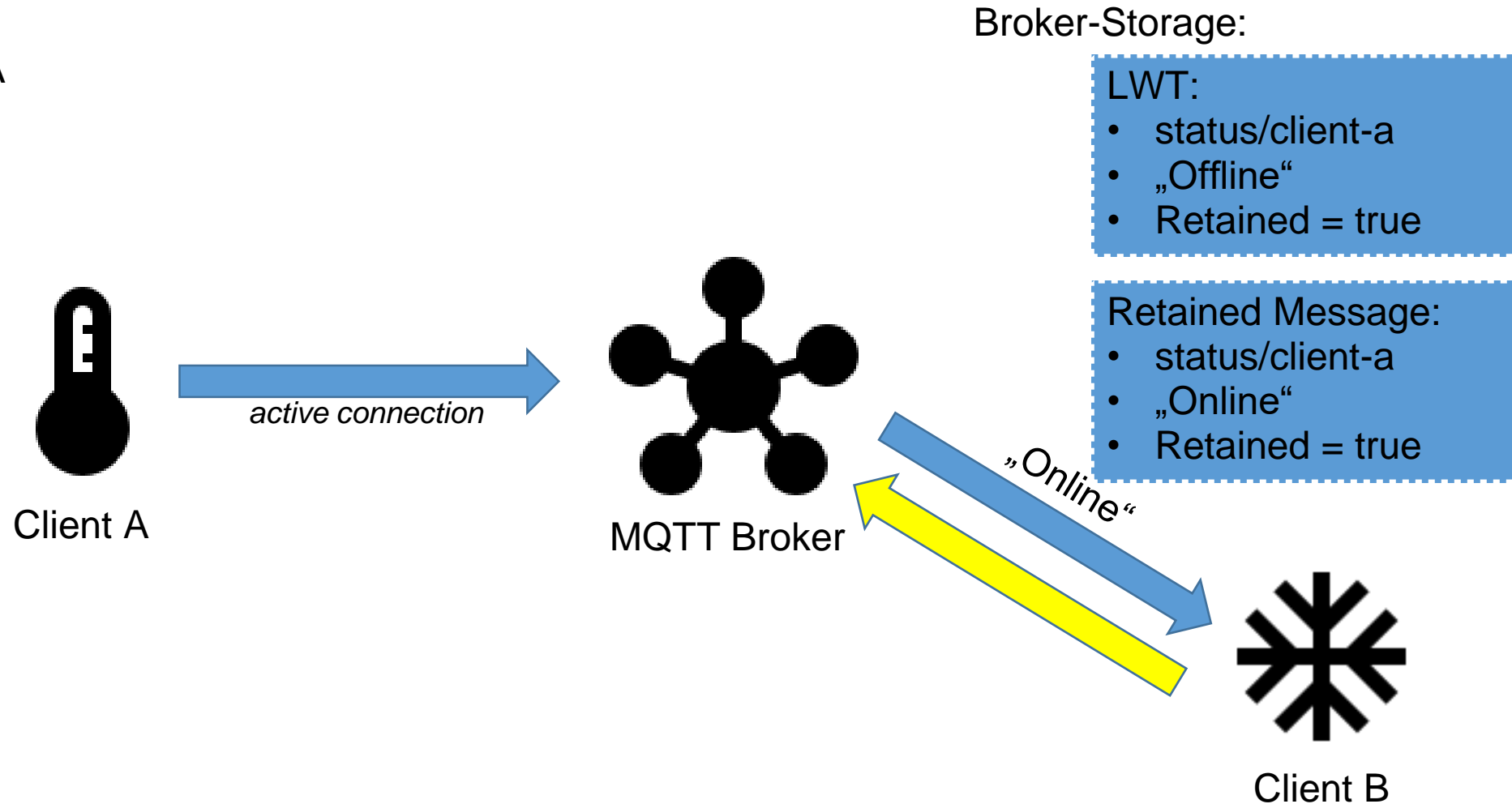
Details - LWT + Retained Message

Option A



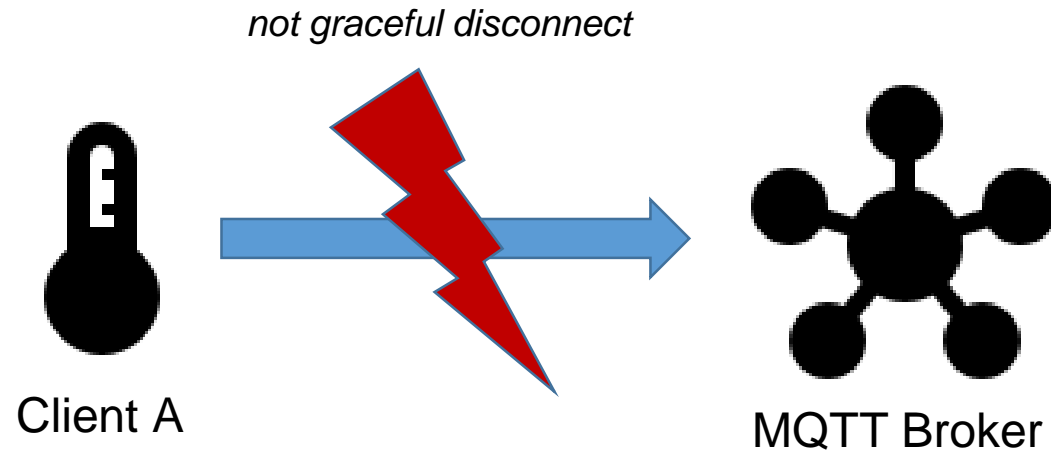
Details - LWT + Retained Message

Option A



Details - LWT + Retained Message

Option B



Broker-Storage:

LWT:

- status/client-a
- „Offline“
- Retained = true

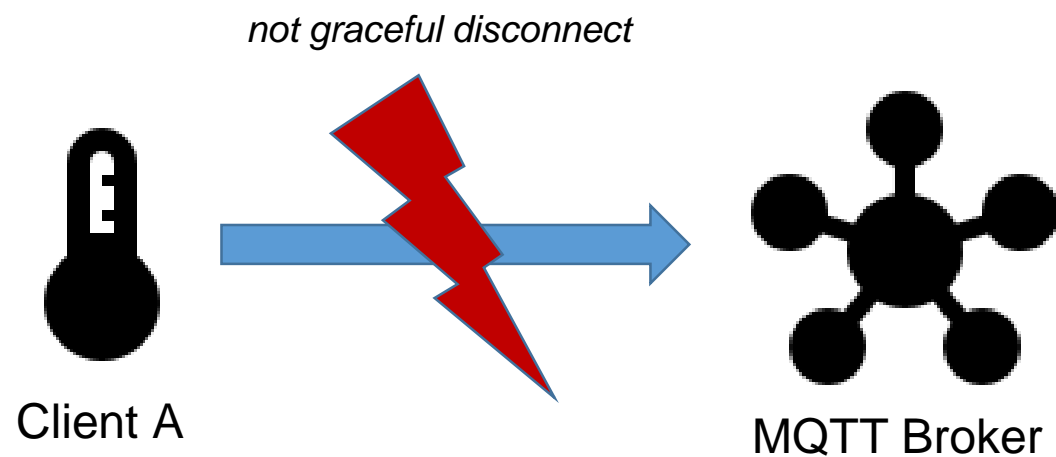
Retained Message:

- status/client-a
- „Online“
- Retained = true

Details - LWT + Retained Message

Broker-Storage:

Option B



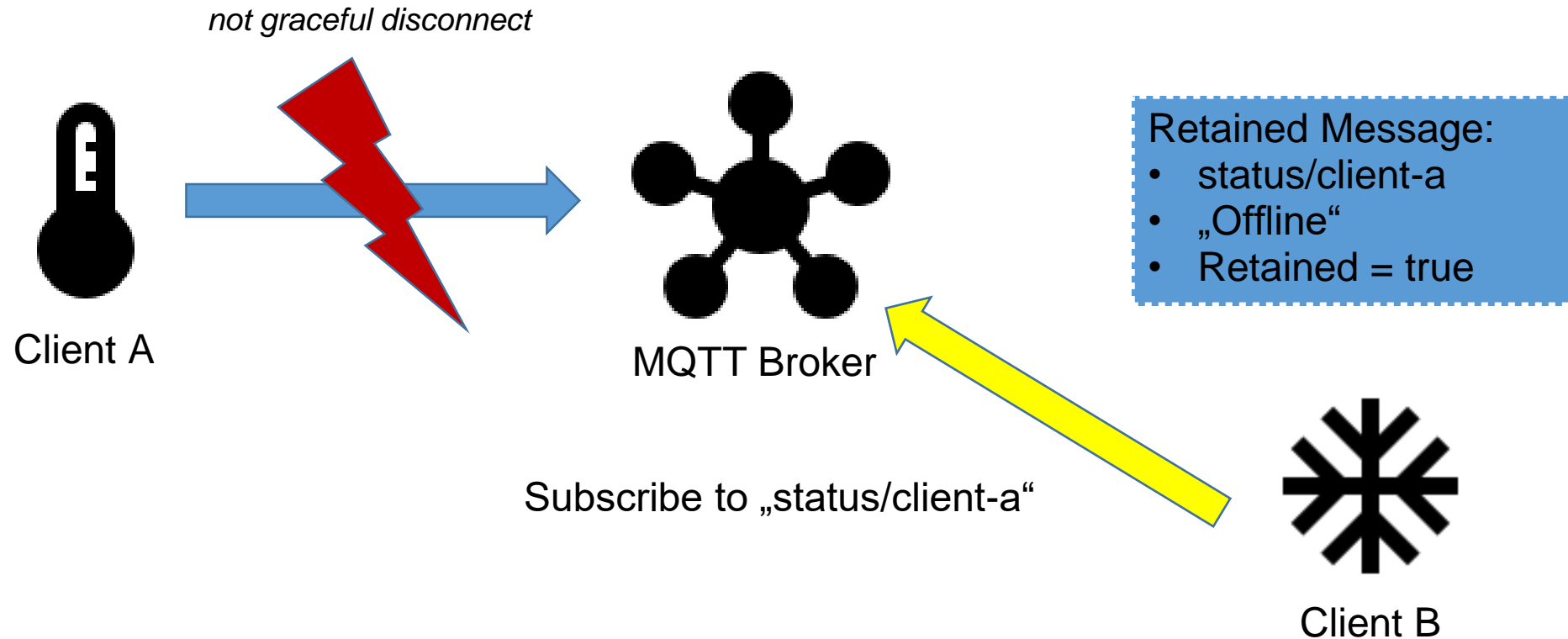
Retained Message:

- status/client-a
- „Offline“
- Retained = true

Details - LWT + Retained Message

Option B

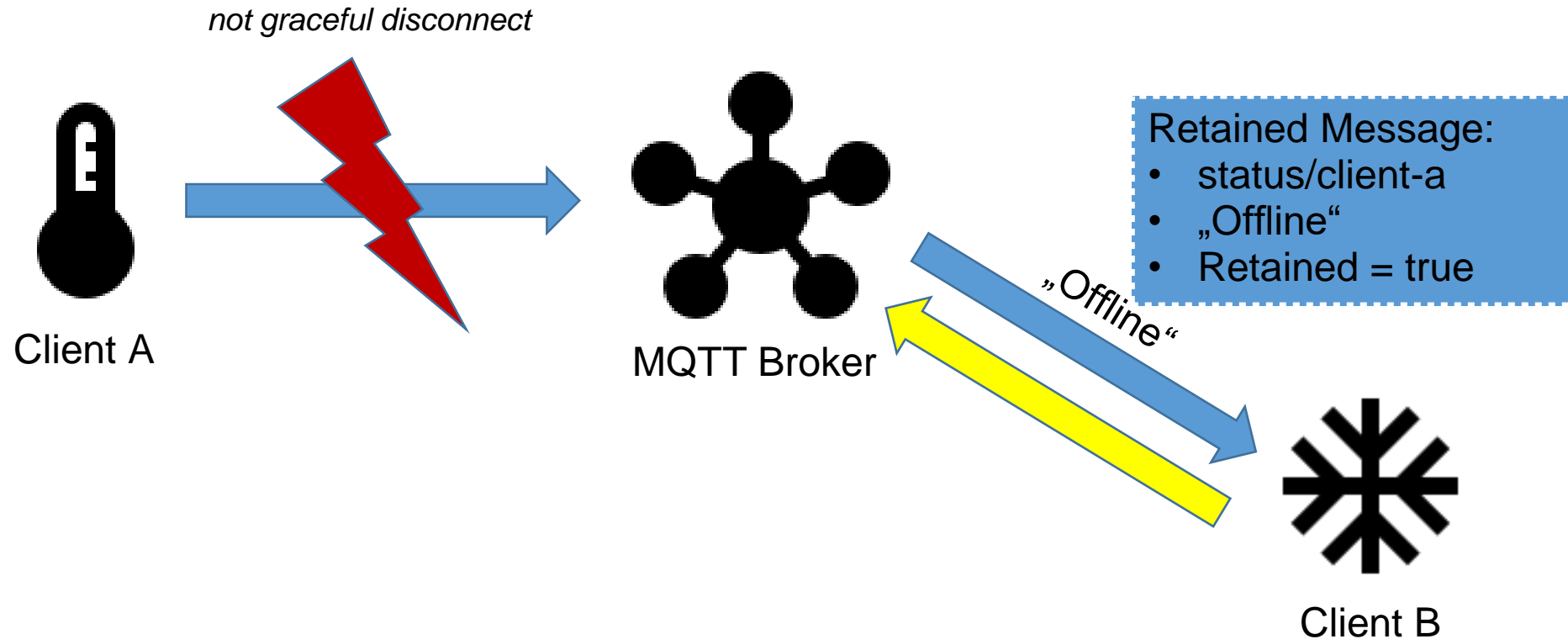
Broker-Storage:



Details - LWT + Retained Message

Option B

Broker-Storage:



Details - Persistent Session

- Saves all details of a session
 - Saves all unfinished communication
 - Saves all missed communication
 - Subscriptions

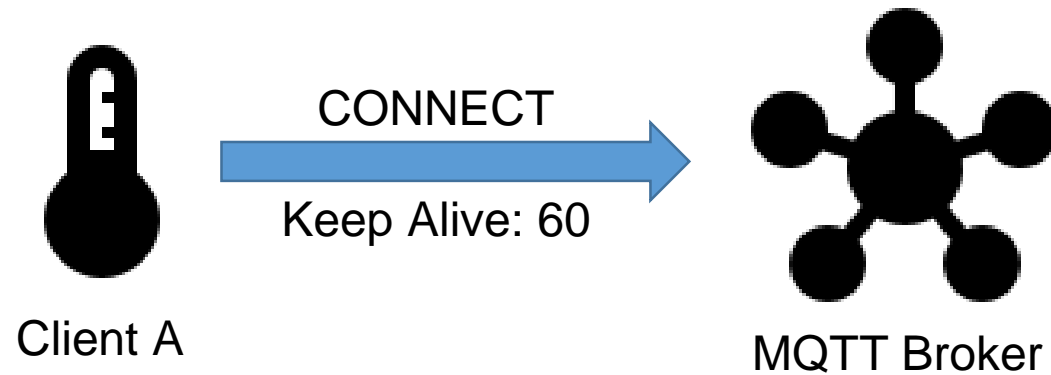
- Can be ignored by Client on CONNECT
 - Clean Start / Clean Session

Details - Keep Alive

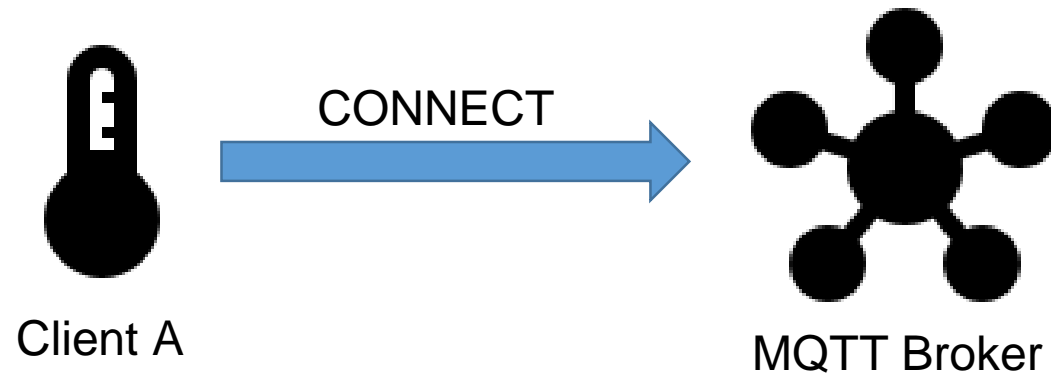
- Interval
- Checks Connection
- Interval is exceeded → Not graceful disconnect

Details – Keep Alive

Broker-Storage:



Details – Keep Alive

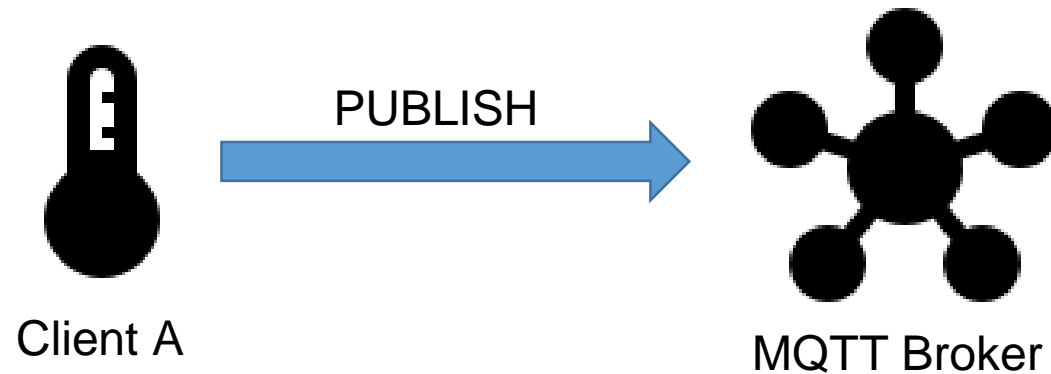


Broker-Storage:

- Client-A-Keep-Alive:
- Interval: 60
 - Last Message: 0

Details – Keep Alive

1. After 55 seconds



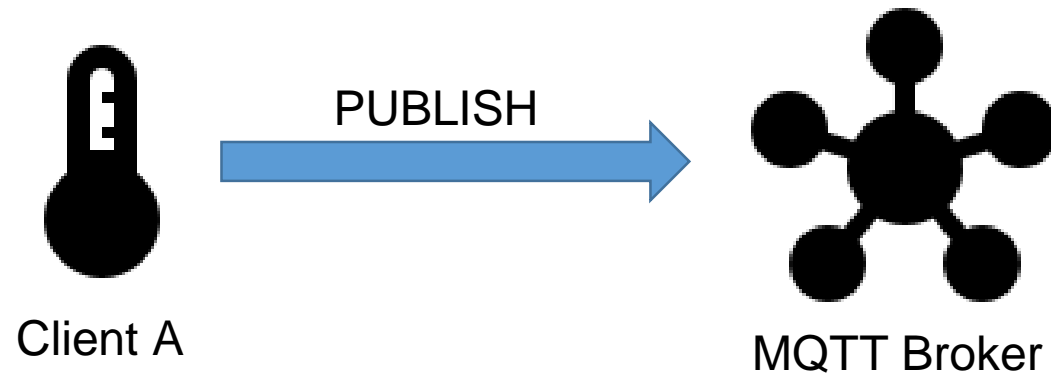
Broker-Storage:

Client-A-Keep-Alive:

- Interval: 60
- Last Message: 55

Details – Keep Alive

1. After 55 seconds



Broker-Storage:

Client-A-Keep-Alive:

- Interval: 60
- Last Message: 0

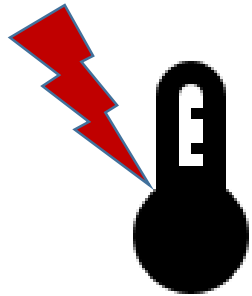
Details – Keep Alive

2. After another 55 seconds

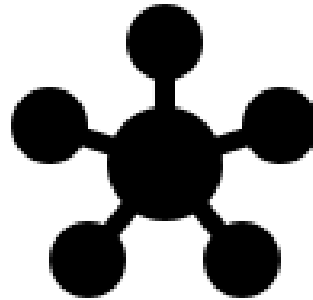
Broker-Storage:

- Client-A-Keep-Alive:
- Interval: 60
- Last Message: 55

measurement error



Client A

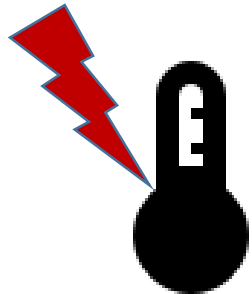


MQTT Broker

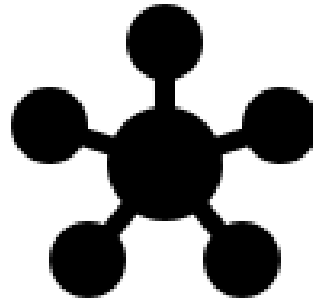
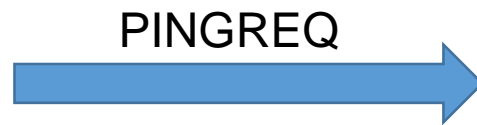
Details – Keep Alive

2. After another 55 seconds

measurement error



Client A



MQTT Broker

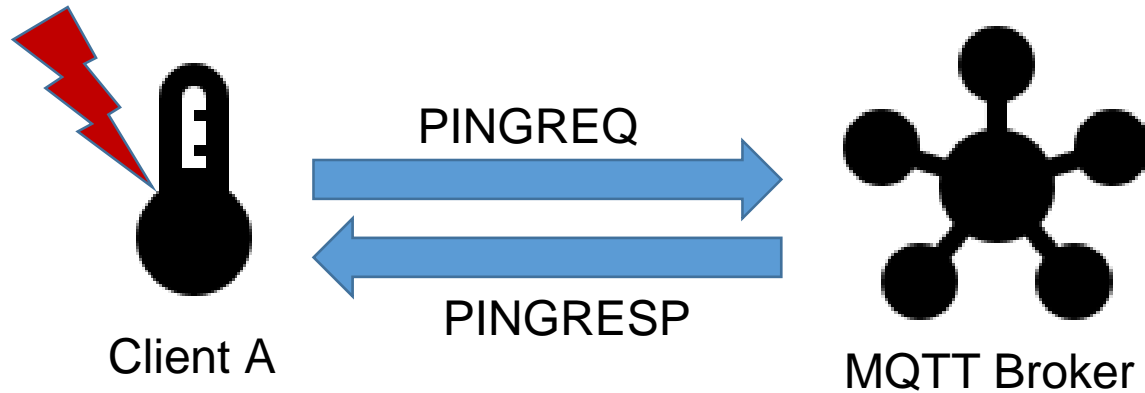
Broker-Storage:

- Client-A-Keep-Alive:
 - Interval: 60
 - Last Message: 55

Details – Keep Alive

2. After another 55 seconds

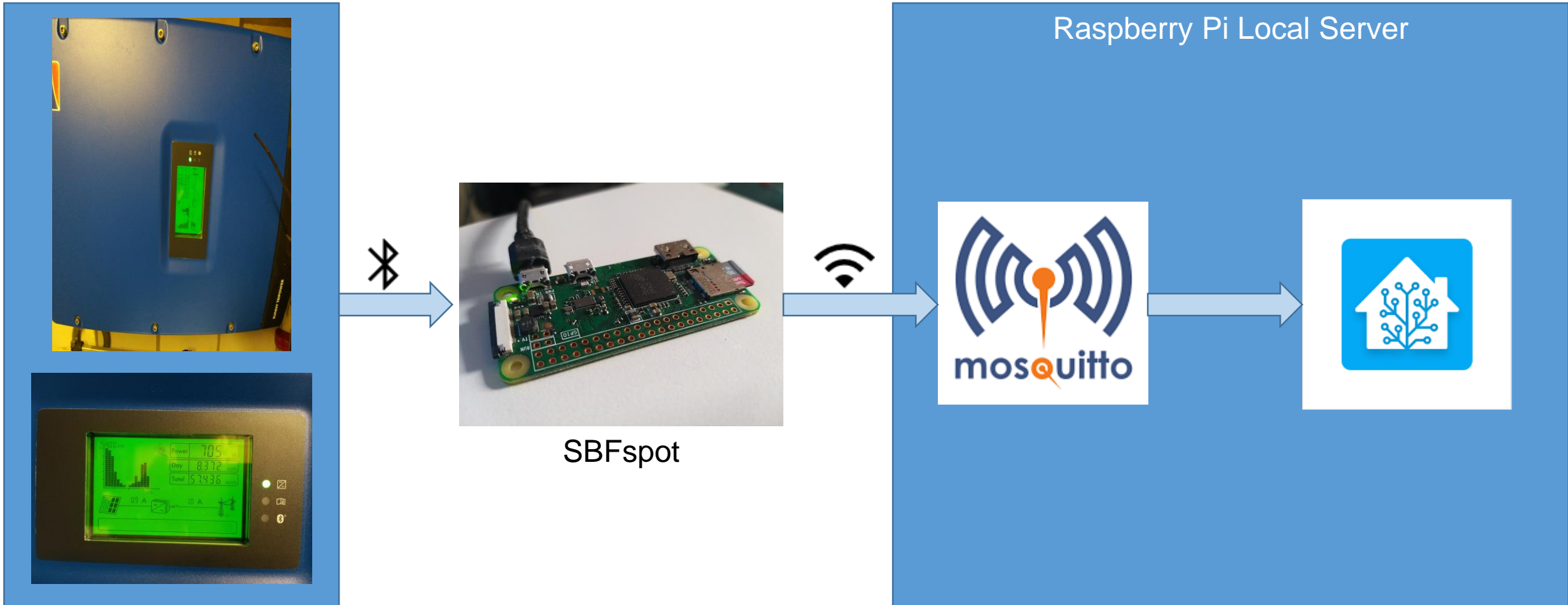
measurement error



Broker-Storage:

- Client-A-Keep-Alive:
 - Interval: 60
 - Last Message: 0

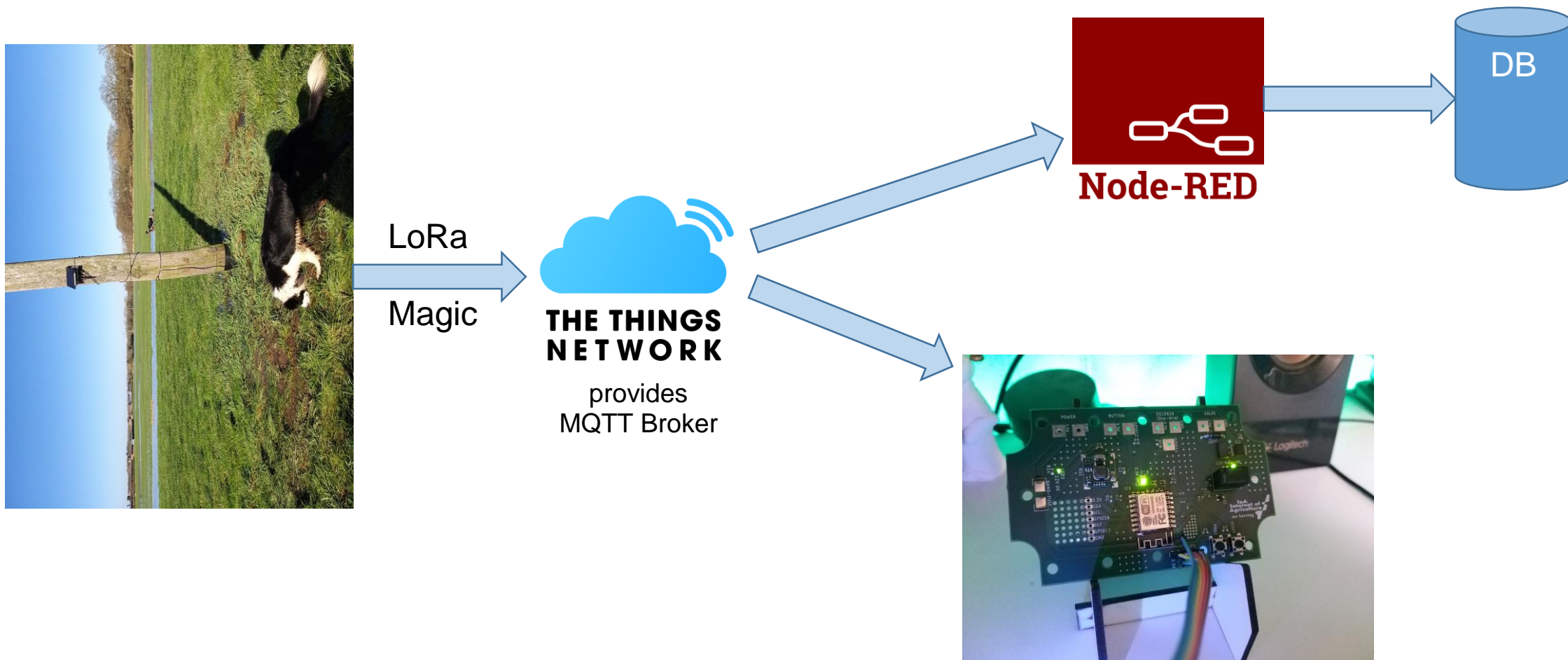
Examples – Solar MQTT



Examples - TTN

- The Things Network
- LoRaWAN
- Multiple kilometer distance
- Low power consumption
- Very good for sensor nodes

Examples - TTN



Practice - Private Broker

- Eclipse Mosquitto
- Easy to setup
- Can run locally for testing

- Cedalo Management Center
- UI for Mosquitto configuration
- Better overview

Practice

- 3 different kind of groups
 - Embedded
 - Java
 - Broker and MQTT Explorer / MQTTeX
- Subscribe and Publish
- Class-wide communication
- DIY

- For Testing: <https://www.hivemq.com/public-mqtt-broker/>