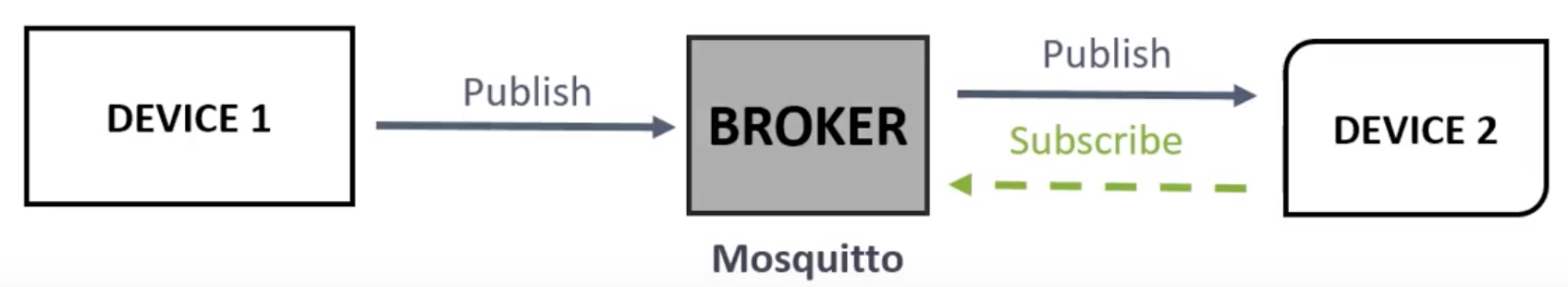
# MQTT

*~MQTT is een M2M publish/subscriber model*

## Basisconcepten



### Publish/subscriber

Vanuit een cliënt kan een bericht uitgestuurd worden naar de broker. Andere devices kunnen subscriben op deze data waardoor deze naar hun gestuurd wordt. (data kan ook bv. een opdracht zijn)

### Topic

* Dit is waarop effectief kan gesubscribed worden. Een bericht dat gepublished wordt moet deze ook bevatten.
* Worden voorgesteld als strings met slashes die het topic-niveau bepalen

Vb: home/office/lamp

Hier zijn home en office topic niveaus

* Hoofdlettergevoelig

### Broker

* Ontvangt de berichten
* Filtert berichten en published de berichten naar de subscribers van het topic beschreven in het ontvangen bericht (Hiervoor wordt meestal de Mosquitto Broker gebruikt, geleverd door Hass.io)
* Kent de verschillende cliënts (ook degene die in het verleden verbonden geweest zijn), samen met hun status en hun topics waarop ze gesubscribed zijn.

### Cliënt

*~Node-RED meest gebruikt | Node.js heeft ook 1 | Eclipse PAHO ook maar outdated*

* Birth message: wordt uitgestuurd elke keer gereconnect wordt
* LWT (Last Will Testament): wordt uitgestuurd wanneer hij offline gaat. Wordt hierom gecached in de broker.

## MQTT transmission

Een transmission bestaat uit 3 delen:

1. Message
2. Topic
3. QoS

### QoS (Quality of Service)

0: gaat het bericht sturen maar trekt er zich voor de rest niets van aan.

1 (at least once): gaat het bericht blijven sturen tot het minstens 1 keer uitgevoerd is.

2 (just once): gaat het bericht sturen tot het juist 1 keer is uitgevoerd. Er worden berichten uitgewisseld om te bevestigen dat het is uitgevoerd.