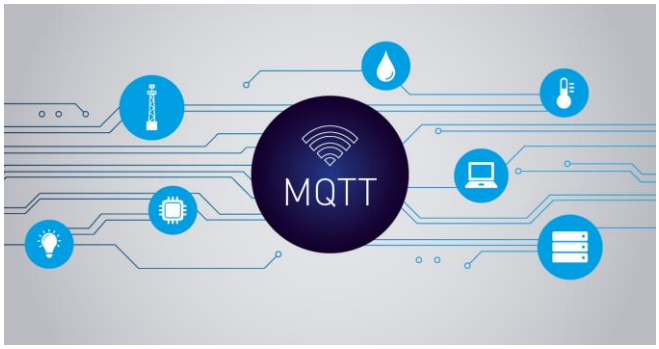


1

MQTT

A practical protocol for the **Internet of Things**

TV Sets

Pacemakers

Ovens

Vehicles

Cows

Smartphones

2

The Internet is (in) everything

- vehicles
- children
- cows
- smartphones
- ovens
- pacemakers

By the year 2020…

57,000 /sec
new objects connecting

212 BILLION
Total number of available sensor enabled objects

30 BILLION
sensor enabled objects **connected to networks**

Data source IDC

3

The world is getting smarter

Smarter Vehicles
- realtime telemetry
- predictive maintenance
- look-ahead alerting
- pay-as-you-drive

Smarter Homes
- energy tracking
- automation
- remote monitoring
- smart appliances

Smarter Logistics
- end-to-end tracking
- theft prevention
- real-time updates
- fleet monitoring

Smarter Healthcare
- smart scales
- in-home monitoring
- assisted living
- physician messaging

4

MQTT a lightweight protocol for IoT messaging

- **open** — open spec, standard — 40+ client implementations
- **lightweight** — minimal overhead — efficient format — tiny clients (kb)
- **reliable** — QoS for reliability on unreliable networks
- **simple** — 43-page spec — connect + publish + subscribe

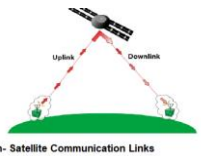
Invented	Published	Eclipse M2M	Standard
EUROTECH IBM	MQTT	paho	OASIS
Late 1990s	Aug 2010	Nov 2011	Sep 2014

5

MQTT – message queuing telemetry transport

- 1999: Andy Stanford-Clark (IBM) and Arlen Nipper (Cirrus Link)





Pipelines in Desert. Low Bandwidth- Satellite Communication Links

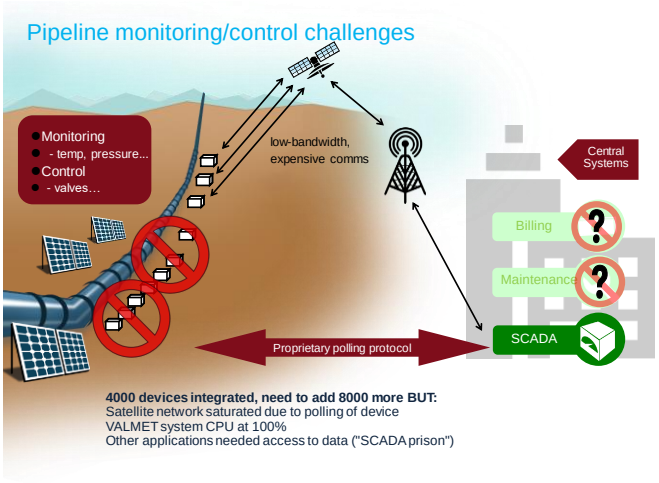
"MQTT is a Client Server publish/subscribe messaging transport protocol. It is light weight, open, simple, and designed so as to be easy to implement. These characteristics make it ideal for use in many situations, including constrained environments such as for communication in Machine to Machine (M2M) and Internet of Things (IoT) contexts where a small code footprint is required and/or network bandwidth is at a premium."

Citation from the official MQTT 3.1.1 s

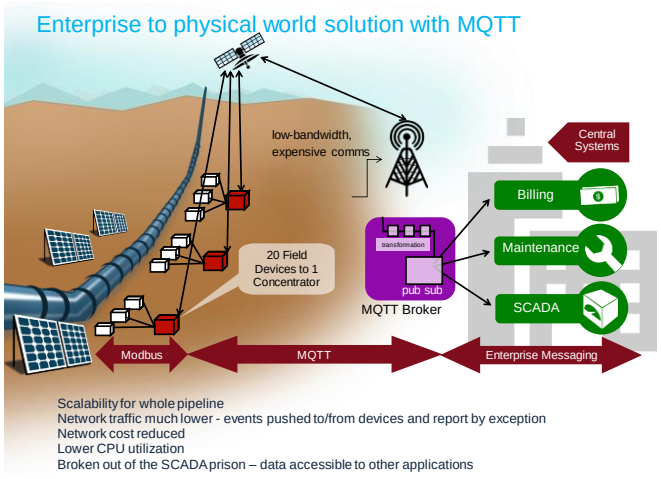
6

Photo credit: Dave Houseknecht, USGS

7



8



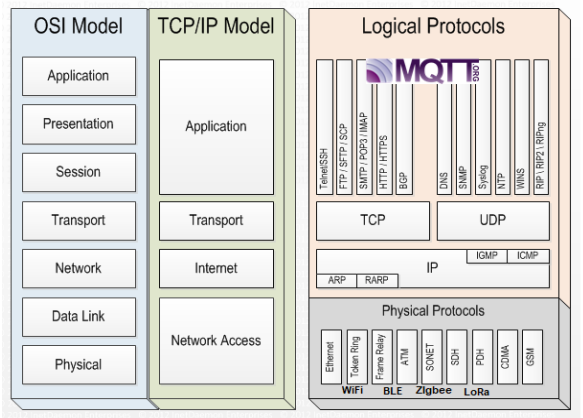
9

MQTT – message queuing telemetry transport

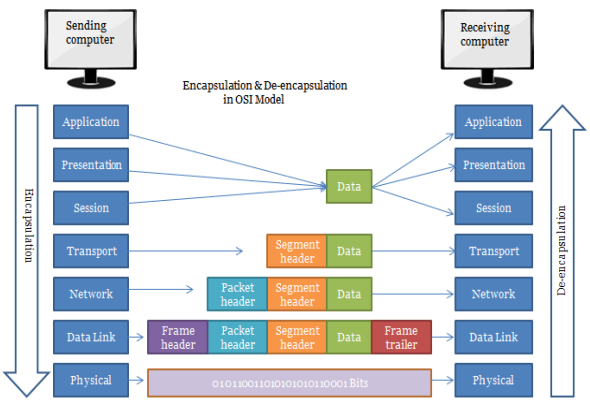
Requirements:

- Simple implementation
- Quality of Service data delivery
- Lightweight and bandwidth efficient
- Data agnostic
- Continuous session awareness

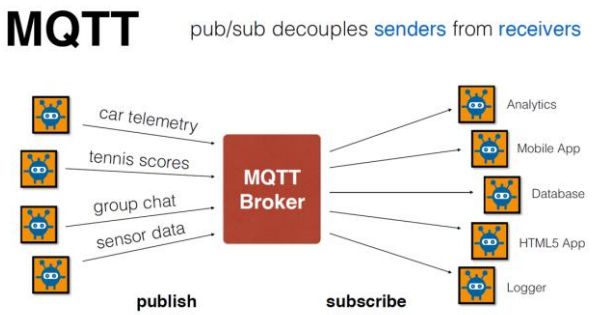
10



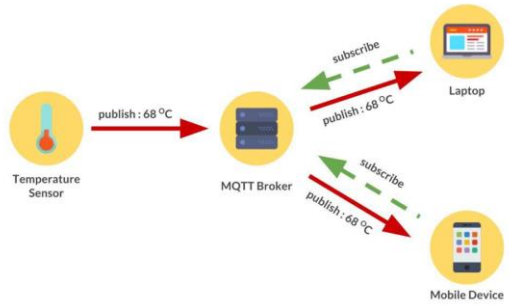
11



12

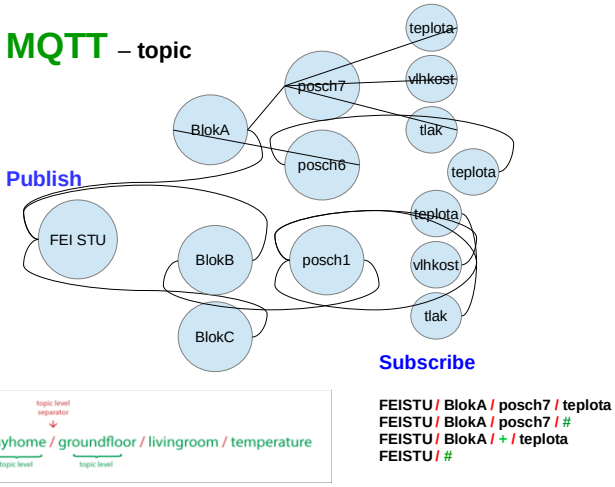


13



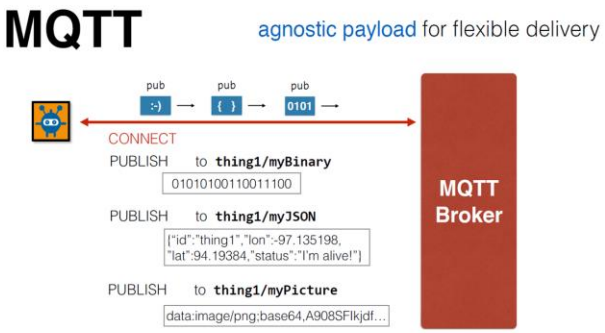
Hermanudin, Aldwin & Ekadiyanto, Fransiskus & Sari, Riri. (2019). Performance Evaluation of CoAP Broker and .

14

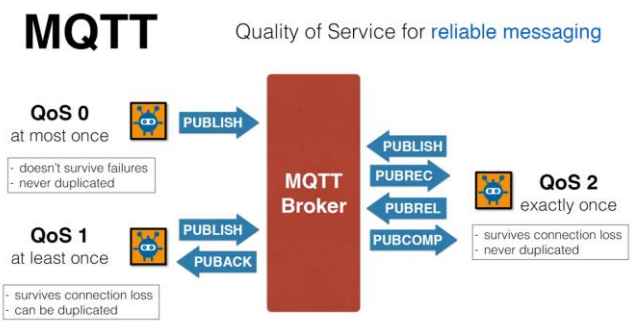


FEISTU / BlokA / posch7 / teplota
FEISTU / BlokA / posch7 / #
FEISTU / BlokA / + / teplota
FEISTU / #

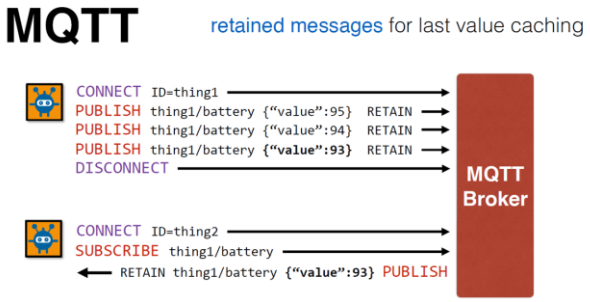
15



16



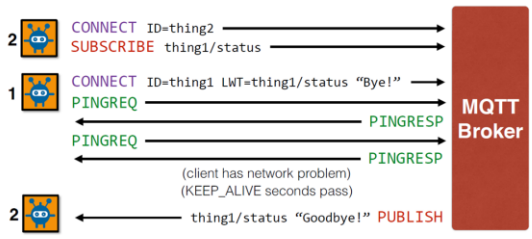
17



18

MQTT

last will and testament for presence



19

Popular Users

Facebook Messenger

openHAB
empowering the smart home

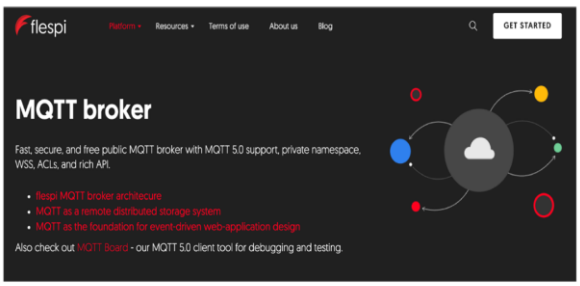
EVRYTHNG

amazon web services

20

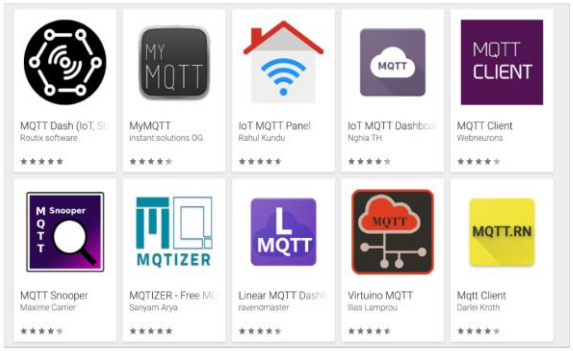
Cloud based brokers: flespi

https://flespi.com/mqtt-broker



21

MQTT clients: Android



22

MQTT – disadvantages

- **If the broker fails…**
- Does not define a standard client API, so application developers have to select the best fit.
- Does not include many features that are common in Enterprise Messaging Systems like:
  - expiration, timestamp, priority, custom message headers, …
- Does not have a point-to-point (aka queues) messaging pattern
  - Point to Point or One to One means that there can be more than one consumer listening on a queue but only one of them will be get the message
- Maximum message size 256MB

23

MQTT – príklad v Processingu

```
import mqtt.*;
MQTTClient client;

void setup() {
  client = new MQTTClient(this);
  client.connect("mqtt://public:public@public.cloud.shiftr.io",
"userName");
}

void draw() { /* draw nothing */}

void keyPressed() {
  client.publish("/FEISTU", "myMessage");
}
```

24

MQTT – príklad v Processingu

```
void clientConnected() {
  println("client connected");
  client.subscribe("/hello");
}

void messageReceived(String topic, byte[] payload) {
  println("new message: " + topic + " - " + new String(payload));
}


void connectionLost() {
  println("connection lost");
}
```

25

26

JSON – JavaScript Object Notation

Douglas Crockford

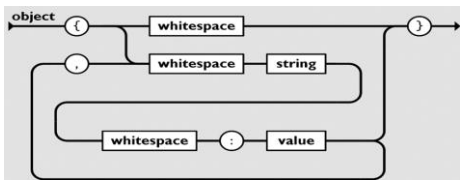
Developed by Douglas Crockford

Standard ISO/IEC 21778:2017

The simplest supported data formats are:

```
    {"key1":"value1", "key2":"value2"}
```

```
{"stringKey":"value1", "booleanKey":true, "doubleKey":42.0, "longKey":73}
```



27

XML vs. JSON

```
<person>
  <name>John Smith</name>
  <age>25</age>
  <address>
    <street>21 2nd Street</street>
    <city>New York</city>
    <state>NY</state>
    <postalCode>10021</postalCode>
  </address>
  <sex>
    <type>male</type>
  </sex>
</person>
```

```
{
  "name": "John Smith",
  "age": 25,
  "address": {
    "street": "21 2nd Street",
    "city": "New York",
    "state": "NY",
    "postal code": "10021"
  },
  "sex": {"type": "male"}
}
```

28

JSON – príklad v Processingu

```
JSONObject message;
void setup()
{  message = new JSONObject();
   message.setFloat("temperature", 10.0);
   message.setInt("state",2);
   message.setString("name", "Lion");

   saveJSONObject(message, "data/new.json");
    int aktualnyStav = message.getInt("state");
   float aktualnaTeplota = message.getFloat("temperature");
   String realName = message.getString("name");
   println("Stav: " + aktualnyStav
  + ", Teplota: " + aktualnaTeplota + ",  Meno: " + realName);
}
```

29

JSON – príklad v Processingu - pokračovanie

```
void draw() { /* nic nekreslime */ }

void keyPressed() {

  temperature = random(-10, 32.5);
  message.setFloat("temperature", temperature);
  println(message.toString());
}
```

30

Úloha – zadanie

Vyskúšajte si posielanie protokolom MQTT. Pošlite jednoduchú správu

MQTT server `mqtt://try:try@broker.shiftr.io`

topic **`/feistu/misa/2024/XXX`**

a potom na

MQTT server `mqtt://9RYd7rPhakMm9CCwPBJG@demo.thingsboard.io`

topic **`v1/devices/me/telemetry`**

Správa vo formáte JSON má vyzerať takto:

```
{"XXXY-Lat": 49.1634, "XXXY-Lon": 20.1349, "XXXY-Temp": 18.2}
```

kde

`XXX` sú prvé tri písmená vášho priezviska a `Y` prvé z krstného mena

`Lat` je zemepisná šírka na štyri desatinné miesta

`Lon` je zemepisná dĺžka na štyri desatinné miesta

`Temp` aktuálna vonkajšia teplota