

The background is a close-up, high-angle shot of a complex electronic circuit board. The board is dark blue or black, with intricate gold and silver-colored traces and components. On the right side, there's a glowing effect of binary code (0s and 1s) in a light blue or cyan color, suggesting digital data flow. The overall lighting is dim, with some highlights on the metallic traces and components.

PSI3542 SISTEMAS EMBARCADOS PARA IOT 2019

PROF. SERGIO TAKEO KOFUJI

PSI-EPUSP

AULA 04 – 13/08/2019

MQTT

INTERNET DAS COISAS

AULA03 - RECAPITULAÇÃO

Parte Prática 01

Plataformas/Ferramentas

Dispositivos

- Notebook: Mouse x,y
- Smartphone: GPS, Acelerometro, medidor de luminosidade

Plataformas

- Dweet.io + Freeboard.io

Roteiro Geral

Atraves do notebook, acesse o site freeboard.io

Crie uma conta gratuita

Siga o roteiro sugerido

- USANDO SO O NOTEBOOK
 - Implemente uma tela mostrando a posicao (X,Y) do mouse
- USANDO O NOTEBOOK+SMARTPHONE
 - Implemente uma tela com o mapa GPS, indicacao de luminosidade, acelerometro, etc...

Atenção:

- Voce pode utilizar o smartphone como dispositivo IoT. Para isso instale o APP dweet
- Voce pode tambem utilizar o seu notebook como dispositivo IoT. Para isso, acesse o site dweet.io e crie um dispositivo
- EM AMBOS OS CASOS, A APLICAÇÃO DE SENSORIMANETO DEVE ESTAR EM FOREGROUND!!!

Tire um screen-shot do dashboard e mande via SLACK com a identificacao do GRUPO (nome dos components)

Parte Prática

1

Familiarização com o Freeboard do GITHUB

- <http://freeboard.github.io/freeboard/>

Crie um Data source do tipo “clock”

Crie um Painel do tipo texto que apresente os dados do Data Source

- So a Data
- So a Hora
- Data + Hora

Repita com o Freeboard.io

Parte Prática

2



Entre no site <https://freeboard.io> e siga o roteiro



Utilizando outra instancia do browser, entre no site <https://dweet.io> e inicie a geração de dados



No freeboard, utilize como fonte de dados a fonte dweet gerada no outro browser



Acrescente um painel



Apresente na forma textual, a posição do mouse



Experimente com outras formas de apresentacao

Parte Prática

3

Utilize o smartphone (conectada ou não na rede local wifi) como fonte de dados, instalando o APP “Dweet My Device”

Execute o APP e mantenha no foreground

Repita o tutorial do freeboard, usando como fonte de dados o smartphone

Exercícios

- Implemente um detetor de queda, utilizando as informações do acelerômetro (X,Y,Z)

AULA04

FREEBOARD+DWEET



COMO CONTROLAR DISPOSITIVOS?



COMO ACRESCENTER NOVOS WIDGETS?



COMO ACRESCENTAR NOVOS
PROTOCOLOS DE COMUNICAÇÃO, COMO
O MQTT?



COMO FAZER PROCESSAMENTO DE
DADOS E ARMAZENAMENTO DE DADOS
HISTÓRICOS?

MQTT

Implementation	Developed by	Open source	Software license	Copyright owner	Written in	Type
Adafruit IO	Adafruit	Yes	MIT License	Adafruit	Ruby on Rails , Node.js , Python	Client
EMQ X ^[8]	EMQ	Yes	Apache License version 2.0	EMQ Enterprise, Inc.	Erlang	Broker
flespi ^[9]	Gurtam	No	Commercial license	Gurtam	C	Broker
HiveMQ MQTT Client ^[8]	dc-square GmbH	Yes	Apache License version 2.0	dc-square GmbH	Java	Client
HiveMQ Community Edition ^[11]	dc-square GmbH	Yes	Apache License version 2.0	dc-square GmbH	Java	Broker
HiveMQ ^[12]	dc-square GmbH	No	Commercial license	dc-square GmbH	Java	Broker
IBM WIoT Message Gateway ^[13]	IBM	No	Commercial license	IBM	C	Broker
JoramMQ [#]	ScalAgent D.T. [#]	No, based on OW2/Joram [#] 5.15, open-source broker (LGPL)	Commercial license	ScalAgent D.T. [#]	Java	Broker
M2Mqtt ^[16]	Eclipse	Yes	Eclipse Public License 1.0	Eclipse	C#	Client
Machine Head ^[16]	ClojureWerkz ^[19]	Yes	Creative Commons Attribution 3.0 Unported License	Alexander Petrov, Michael Klishin, ClojureWerkz Team	Clojure	Client
moquette ^[20]	Andrea Selva	Yes	Apache License version 2.0	Andrea Selva	Java	Broker
Mosquitto ^[22]	Eclipse	Yes	Eclipse Public License 1.0 , Eclipse Distribution License 1.0 (BSD)	Eclipse	C	Client and Broker
MQTT-C [#] ^[24]	Liam Bindle	Yes	MIT License	Liam Bindle	C	Client
Paho MQTT ^[26]	Eclipse	Yes	Eclipse Public License 1.0 , Eclipse Distribution License 1.0 (BSD) ^[27]	Eclipse	C, C++, Java, JavaScript, Python, Go	Client
PubSub+ ^[29]	Solace	No	Commercial license, free version	Solace	C, C++	Broker
Thingstream ^[30]	Thingstream	No	Commercial licence	Thingstream	C, C++, Java, JavaScript, Python, Go	Client and broker
VerneMQ ^[32]	VerneMQ/Erlot	Yes	Apache License version 2.0	VerneMQ/Erlot	Erlang/OTP	Broker
wolfMQTT ^[33]	wolfSSL ^[34]	Yes	GNU Public License, version 2, Commercial License	wolfSSL	C	Client
Bevywise MQTTBroker ^[35]	Bevywise Networks [#]	No	Commercial license	Bevywise Networks [#]	C, Python	Broker

MQTT/MQTT-SN

Protocolo de Comunicação para IoT

<https://www.hivemq.com/mqtt-essentials/>

<https://fiware-tutorials.readthedocs.io/en/latest/iot-over-mqtt/index.html>

<http://www.steves-internet-guide.com/mqtt/>

<https://learn.sparkfun.com/tutorials/introduction-to-mqtt/all>

<https://1sheeld.com/mqtt-protocol/>

MQTT

<https://www.hivemq.com/mqtt-essentials/>

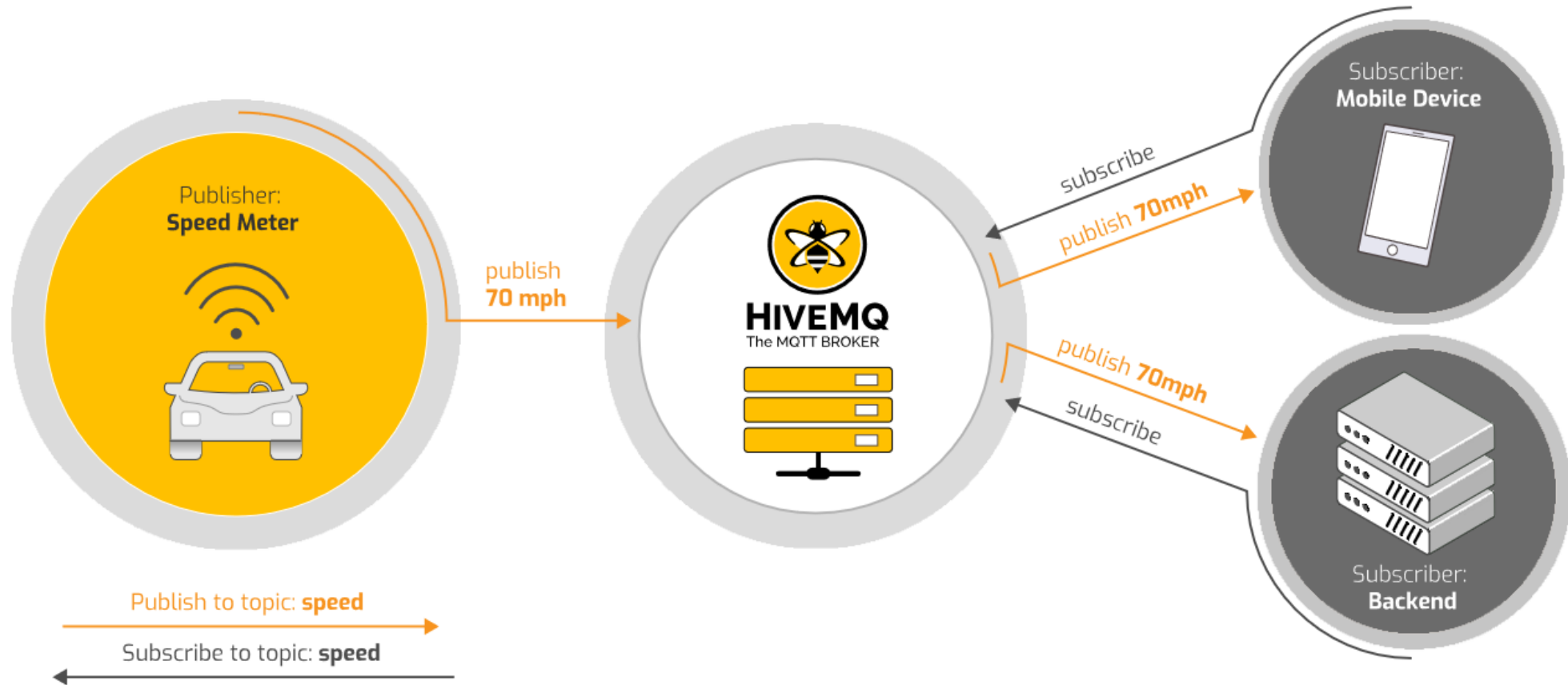
1999. Andy Stanford-Clark (IBM) and Arlen Nipper:
protocol for minimal battery loss and minimal bandwidth to connect with oil pipelines via satellite

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

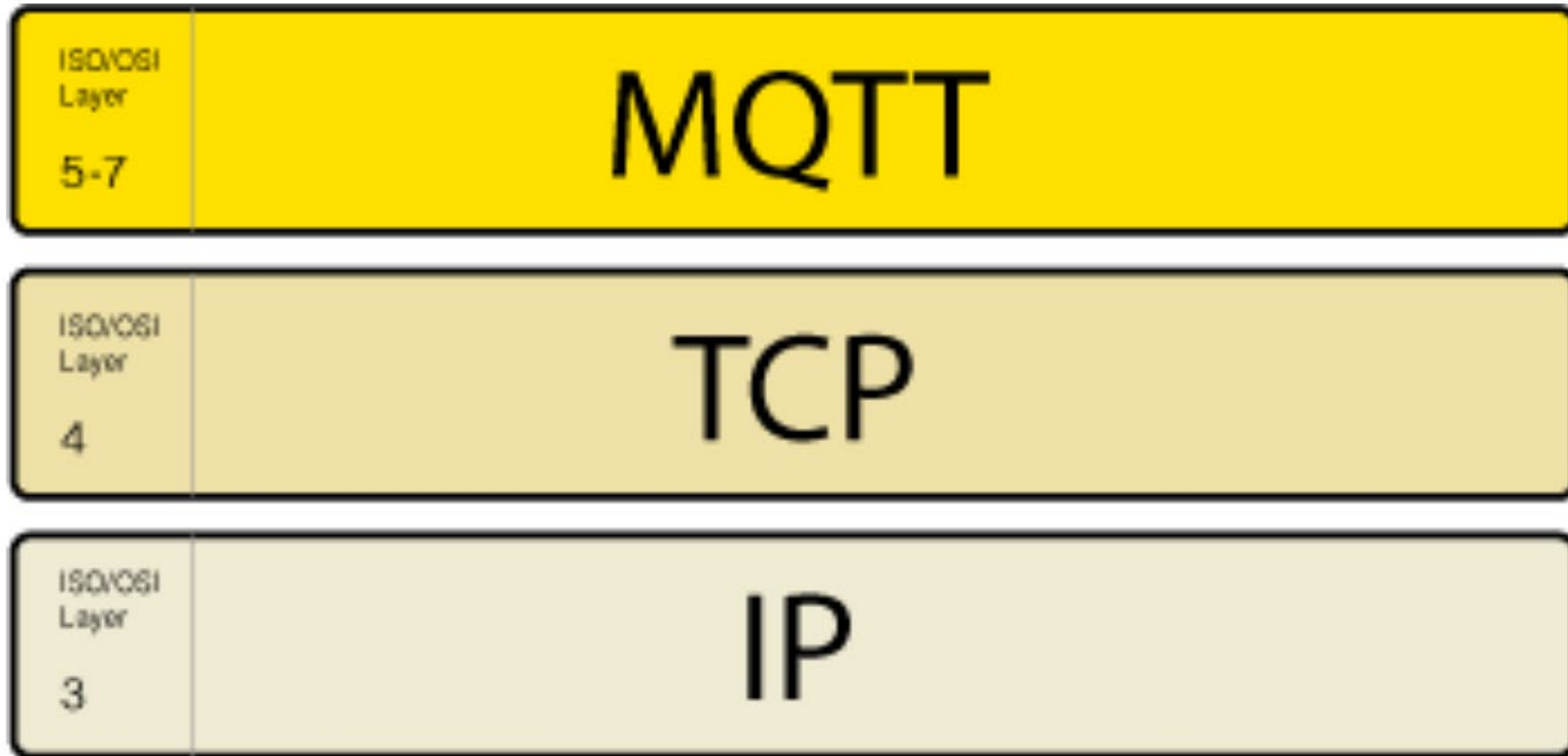
IBM, 2010: MQTT 3.1 royalty-free version

OASIS, 2014: MQTT 3.1.1 became an officially approved OASIS Standard

HIVEMQ



MQTT Connection



MQTT Connection



MQTT Client



MQTT Broker

MQTT Packets

MQTT-Packet:

CONNECT



contains:

Example

clientId

"client-1"

cleanSession

true

username (optional)

"hans"

password (optional)

"letmein"

lastWillTopic (optional)

"/hans/will"

lastWillQos (optional)

2

lastWillMessage (optional)

"unexpected exit"

lastWillRetain (optional)

false

keepAlive

60

MQTT PUBLISH

MQTT-Packet:

PUBLISH



contains:

packetId (always 0 for qos 0)

topicName

qos

retainFlag

payload

dupFlag

Example

4314

"topic/1"

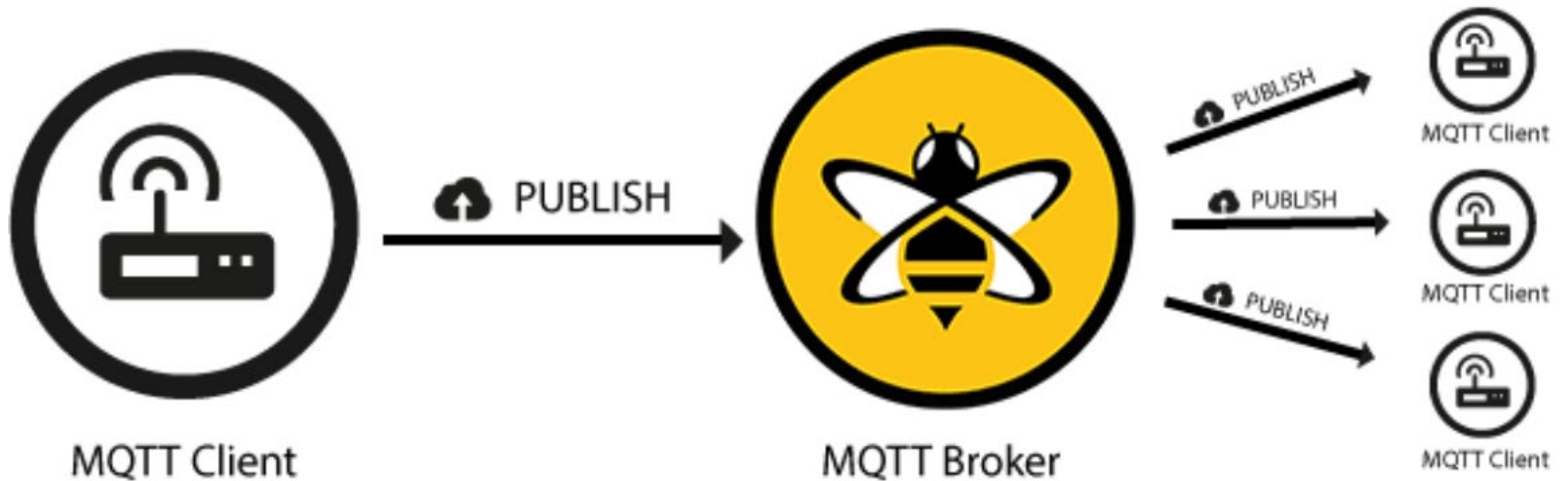
1

false

"temperature:32.5"

false

MQTT PUBLISH-SUBSCRIBE



MQTT SUBSCRIBE

MQTT-Packet:

SUBSCRIBE



contains:

packetId

qos1 } (list of topic + qos)
topic1

qos2 }
topic2

...

Example

4312

1

"topic/1"

0

"topic/2"

...

MQTT SUBSCRIBE

MQTT-Packet:

SUBACK



contains:

packetId

returnCode 1 (one returnCode for each
returnCode 2 topic from SUBSCRIBE,
in the same order)

...

Example

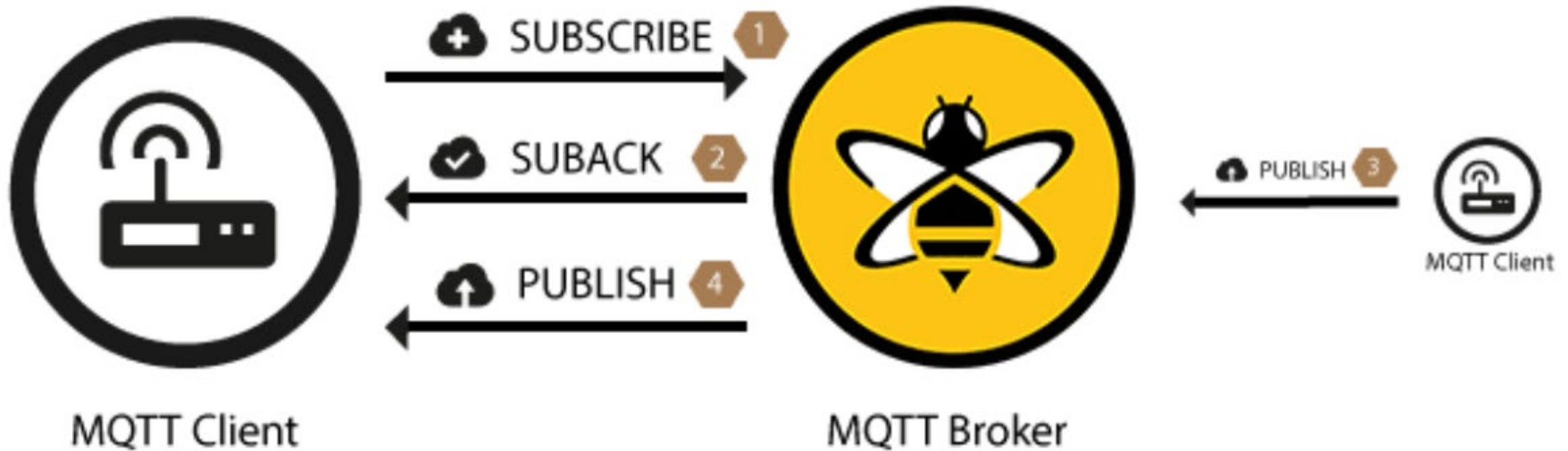
4313

2

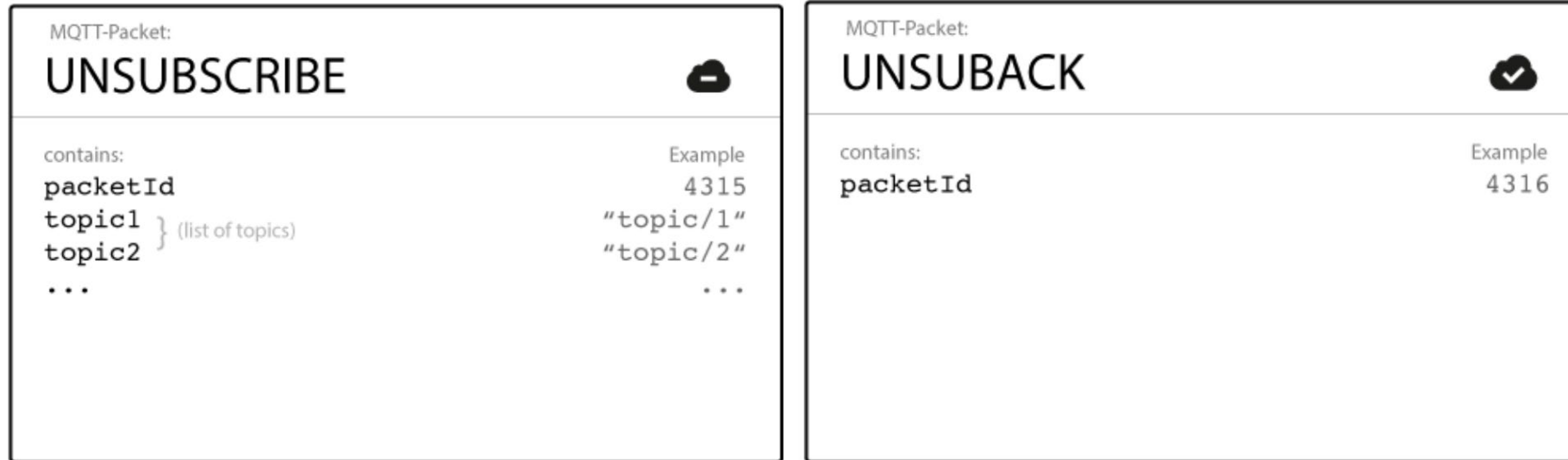
0

...

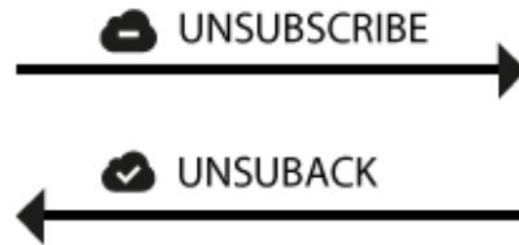
MQTT SUBSCRIBE...



MQTT Unsub



MQTT Client



MQTT Broker

MQTT OUTROS PONTOS..

TOPIC IDENTIFICATION

WILDCARDS

- SINGLE-LEVEL: +
- MULTILEVEL: #

TOPICS BEBINING WITH "\$"

MQTT BROKER IMPLEMENTAÇÃO

BROKERS MQTT



Use you Own Locally Installed Server



Use a Cloud Based Sever or Virtual Server



Use a Shared Server Application

Mosca	Mosca is Node.js based and so requires node.js to be installed. See GitHub It can also be installed as a node in node-red. It is not very feature rich when compared to mosquitto.
emqtt	Written in Erlang is Open Source and described as massively scalable. <i>EMQ</i> Implements both MQTT V3.1 and V3.1.1 protocol specifications, and supports MQTT-SN, CoAP, WebSocket, STOMP . See Github
Python Test Broker	Python test brokers including MQTT v5 https://github.com/eclipse/paho.mqtt.testing/tree/master/interoperability
VerneMQ	Written in Erlang and supporting clustering. Website

Online Cloud Base MQTT Brokers/Servers

Broker Type	Broker Address and Port	Websocket Support	SSL support
Mosquitto	test.mosquitto.org 1883	Yes Encrypted port 8081 Un-encrypted 8080	Yes 8883 With Client certificate 8884
HiveMQ	broker.hivemq.com 1883	Yes 8000	
Mosquitto	iot.eclipse.org	Yes 80 and 443 (SSL)	Yes 8883
mosca	test.mosca.io 1883		

10 Free Public & Private MQTT Brokers(For Testing & Prototyping)

<https://mntolia.com/10-free-public-private-mqtt-brokers-for-testing-prototyping/>

PUBLIC MQTT BROKERS

Name	Broker Address	Regular/TCP Port	SSL/TLS PORT	Web Socket Port	Retain Messages	Persistent Session	Sign Up Required	Link
Eclipse	iot.eclipse.org	1883	N/A	80, 443	YES	YES	NO	http://iot.eclipse.org/sandbox.html
Mosquitto	test.mosquitto.org	1883	8883, 8884	80	YES	YES	NO	http://test.mosquitto.org/
HiveMQ	broker.hivemq.com	1883	N/A	8000	YES	YES	NO	https://www.hivemq.com/try-out/
Flespi	mqtt.flespi.io	1883	8883	80, 443	YES	YES	YES	https://flespi.com/mqtt-broker
Dioty	mqtt.dioty.co	1883	8883	8080, 8880	YES	YES	YES	http://www.dioty.co/
Fluux	mqtt.fluux.io	1883	8883	N/A	N/A	N/A	NO	https://fluux.io/

PRIVATE MQTT BROKERS

Broker Name	URL	Regular/TCP Port	SSL/TLS PORT	Web Socket Port	Retain Messages	Persistent Session	QoS Levels	Free Limits
Azure	https://docs.microsoft.com/en-us/azure/iot-hub/iot-hub-mqtt-support	N/A	8883	443	NO	YES	0, 1	8000 messages/day
AWS	https://docs.aws.amazon.com/iot/latest/developerguide/mqtt.html	N/A	8883	443	NO	YES	0, 1	250,000/month
CloudMQTT	https://www.cloudmqtt.com/plans.html	Custom Port	Custom Port	Custom Port	NOT SURE	YES	0, 1, 2	5 Connections & 10 Kbit/s

5 Best MQTT Clients for Testing on Desktop & Mobile

- <https://mntolia.com/5-best-mqtt-tools-for-desktop-mobile/>

DUVIDAS?

KOFUJI@USP.BR

