



Data Lake – Design, Projeto e Integração 2.0

Data Lake - Design, Projeto e Integração Versão 2.0

O Que é MQTT (Message Queuing Telemetry Transport)?



Para os dispositivos de Internet das Coisas (IoT), a conexão com a Internet é um requisito. A conexão com a Internet permite que os dispositivos trabalhem entre si e com serviços de backend. O protocolo de rede da Internet é o TCP/IP. Desenvolvido com base na pilha TCP/IP, o MQTT (Message Queue Telemetry Transport) tornou-se o padrão para comunicações de IoT.

MQTT é um protocolo muito simples e leve, baseado em publicação e subscrição de eventos. O MQTT foi projetado para conexões com ambientes remotos onde um "pequeno tamanho de código" é necessário ou a largura de banda da rede é limitada, ideal para sistemas embarcados em dispositivos de Internet das Coisas (IoT).

MQTT (em português, Transporte de Telemetria do Serviço de Enfileiramento de Mensagens) é um protocolo de mensagens com base na publicação/assinatura do padrão ISO (ISO / IEC PRF 20922) e funciona sobre o protocolo TCP/IP. O padrão de mensagens de publicação/assinatura requer um intermediário de mensagens chamado broker (veremos mais sobre isso no próximo capítulo).

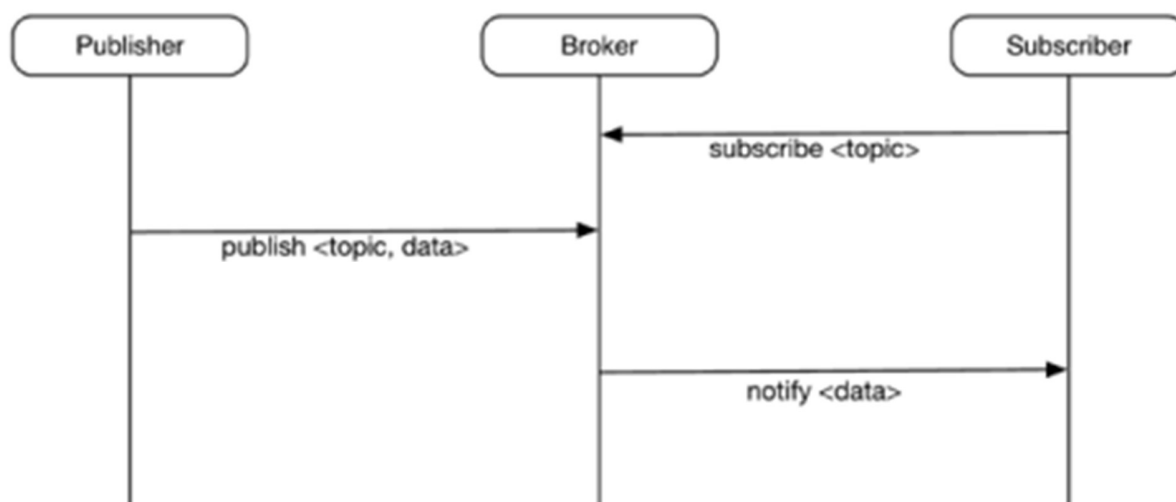
O protocolo Message Queue Telemetry Transport (MQTT) está presente no dia a dia da Internet das Coisas (IoT) e o seu principal uso é fazer as máquinas conversarem, também conhecido como Machine-to-Machine (M2M).

Andy Stanford-Clark, da IBM, e Arlen Nipper, da Cirrus Link, criaram a primeira versão do protocolo em 1999.

O protocolo MQTT é baseado no TCP/IP e ambos, cliente e broker, necessitam da pilha TCP/IP para o seu funcionamento. O MQTT está na mesma camada OSI que o HTTP, porém a maior diferença entre os dois protocolos é o tamanho do payload. No HTTP, o payload é maior, o que inviabiliza o seu uso em conexões de baixa qualidade, como GSM por exemplo.

O MQTT utiliza o paradigma publish/subscribe (pub/sub) para a troca de mensagens. O paradigma pub/sub implementa um middleware chamado de broker. O broker é responsável por receber, enfileirar e disparar as mensagens recebidas dos publishers para os subscribers. O publisher é responsável por se conectar ao broker e publicar mensagens. Já o subscriber é responsável por se

conectar ao broker e receber as mensagens que ele tiver interesse. Na imagem abaixo possível observar a arquitetura do paradigma pub/sub.



O paradigma pub/sub utiliza o conceito de tópicos para processar as mensagens, em que cada mensagem é enviada para um determinado tópico. Diferente de outros protocolos de mensagem, o publisher não envia a mensagem diretamente ao subscriber, mas sim ao broker. O publisher envia a mensagem para o broker em um determinado tópico. O broker é responsável por receber a mensagem do publisher e fazer uma pré-filtragem das mensagens e enviá-las para os subscribers que estiverem registrados em um determinado tópico.

Existem vários brokers MQTT disponíveis, uns pagos outros gratuitos. Uma lista dos principais brokers pode ser conferida aqui:

<https://github.com/mqtt/mqtt.github.io/wiki/servers>

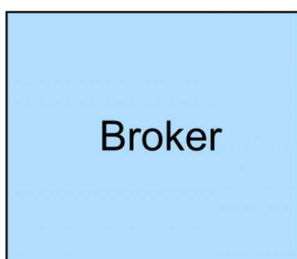
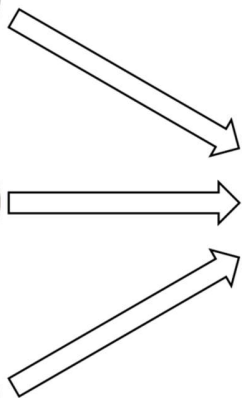
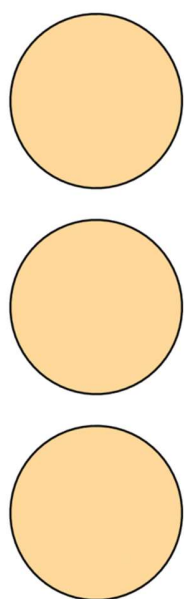
O RabbitMQ (<http://www.rabbitmq.com/>) é um broker AMQP poderoso que através de um plugin permite adicionar suporte a versão 3.x do MQTT. Mas nós usaremos o HiveMQ.

O HiveMQ é um broker MQTT que foi desenvolvido desde o início com a máxima escalabilidade e segurança pronta para a empresa em mente. Ele vem com suporte a soquete da web nativo (web socket) e um plug-in de software livre SDK para estender sua funcionalidade ou integrá-lo a outros componentes. Um servidor de teste público também está disponível. Aqui tem uma lista de brokers públicos:

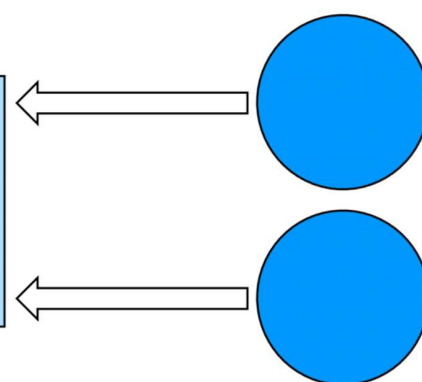
https://github.com/mqtt/mqtt.github.io/wiki/public_brokers

Neste Mini-Projeto, usaremos o Apache NiFi para consumir os eventos do broker (em nosso caso simulação de dados de sensores IoT) como um cliente e armazenar os dados no HDFS.

Sensores
(publish)



Clientes diversos
(subscriber)



No link abaixo você encontra um excelente material da IBM sobre o MQTT **em português**. Leia este material antes de prosseguir para as próximas aulas:



<https://www.ibm.com/developerworks/br/library/iot-mqtt-why-good-for-iot/index.html>

Outras Referências (recomendamos a leitura do material nos links abaixo):

Você sabe o que é MQTT? Descubra aqui!

<http://engprocess.com.br/mqtt/>

MQTT - Protocolos para IoT

<https://www.embarcados.com.br/mqtt-protocolos-para-iot/>

IoT #05 — Dando uma breve análise no protocolo MQTT

https://medium.com/@bruno_live/iot-05-dando-uma-breve-an%C3%A1lise-no-protocolo-mqtt-e404e977fbb6