



Universidade do Estado do Rio de Janeiro

Centro de Tecnologia e Ciências

Faculdade de Engenharia

Fernando de Oliveira Lima

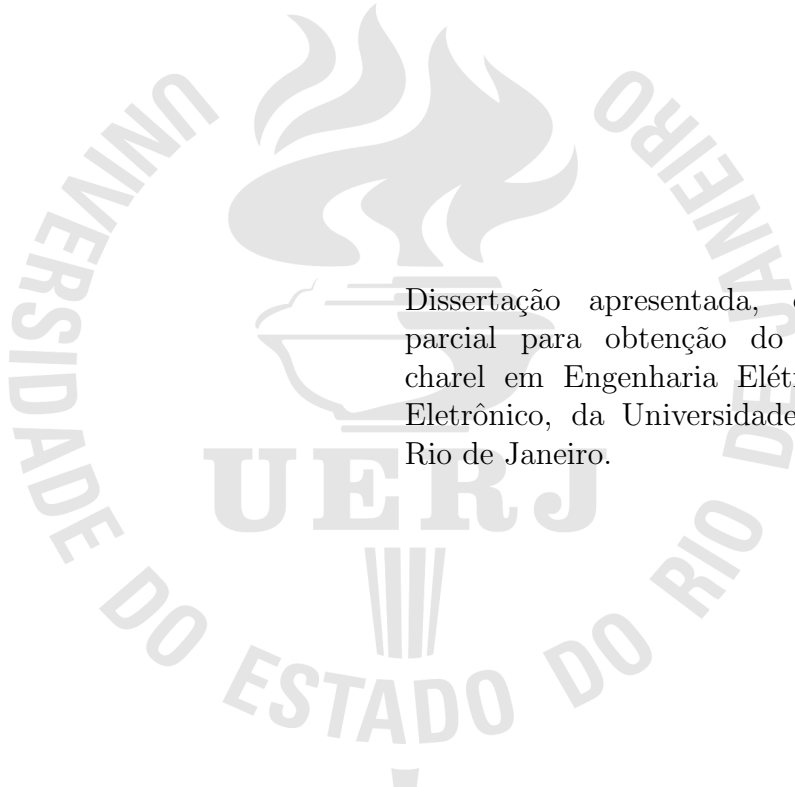
**Sistema Escalável para Aplicações de Internet das Coisas
utilizando MQTT**

Rio de Janeiro

2018

Fernando de Oliveira Lima

Sistema Escalável para Aplicações de Internet das Coisas utilizando MQTT



Dissertação apresentada, como requisito parcial para obtenção do título de Bacharel em Engenharia Elétrica - Sistemas Eletrônico, da Universidade do Estado do Rio de Janeiro.

Orientador: Prof. Michel Tcheou, DSc

Co-Orientador Prof. Lisandro Lovisolo, DSc

Rio de Janeiro

2018

CATALOGAÇÃO NA FONTE

S237

UERJ / REDE SIRIUS / BIBLIOTECA CTC/B

Sobrenome, Nome do Autor

Título do trabalho / Nome completo do autor. – 2012.
105 f.

Orientadores: Nome do orientador1;

Nome do orientador1.

Dissertação(Mestrado) – Universidade do Estado do Rio de
Janeiro, Faculdade de Engenharia.

Texto a ser informado pela biblioteca

CDU 621:528.8

Autorizo, apenas para fins acadêmicos e científicos, a reprodução total ou parcial desta
dissertação, desde que citada a fonte.

Assinatura

Data

Nome do Aluno

Título do Trabalho

Dissertação apresentada, como requisito parcial para obtenção do título de Bacharel em Engenharia Elétrica - Sistemas Eletrônico, da Universidade do Estado do Rio de Janeiro.

Aprovado em: 28 de Agosto 2018

Banca Examinadora:

Prof. Dr. Nome do Professor 1 (Orientador)
Instituto de Matemática e Estatística da UERJ

Prof. Dr. Nome do Professor 2
Faculdade de Engenharia da UERJ

Prof. Dr. Nome do Professor 3
Universidade Federal do Rio de Janeiro - UFRJ - COPPE

Prof. Dr. Nome do Professor 4
Instituto de Geociências da UFF

Prof. Dr. Nome do Professor 5
Universidade Federal do Rio de Janeiro - UFRJ - COPPE

Rio de Janeiro

2018

DEDICATÓRIA

Aqui entra sua dedicatória.

AGRADECIMENTO

Aqui entra seu agradecimento.

É importante sempre lembrar do agradecimento à instituição que financiou sua bolsa, se for o caso...

Agradeço à FAPERJ pela bolsa de Mestrado concedida.

RESUMO

LIMA, Fernando *Sistema Escalável para Aplicações de Internet das Coisas utilizando MQTT*. 105 f. Dissertação (Engenharia Elétrica - Sistemas Eletrônicos) - Faculdade de Engenharia, Universidade do Estado do Rio de Janeiro (UERJ), Rio de Janeiro, 2018.

No meio da revolução dos dados, cresce o interesse em comunicação entre máquinas e compartilhamento de dados telemétricos sobre dispositivos, seja numa fábrica ou em residências. Esta dissertação trata sobre um sistema para aplicações de internet das coisas(IoT) utilizando MQTT a *lingua franca* para publicação de dados telemétricos via TCP/IP. Englobando todos os setores de aquisição dos dados a camada de aplicação em consoles.

Palavras-chave: iot, mqtt, indústria.

ABSTRACT

In the verge of the data revolution, a growing interest in communication between machines and the sharing of telemetric data on devices rises, whether in a factory or in a residence. This dissertation deals with a system for Internet applications of things (IoT) using MQTT the *lingua franca* for publishing telemetric data via TCP / IP. Encompassing all sectors of data acquisition to the application layer in consoles.

Keywords: iot, mqtt, industry .

LISTA DE FIGURAS

Figura 1	As três camadas do IoT, dos sensores ao mundo real.....	12
Figura 2	Diferenças entre OSI e TCP/IP em suas camadas	15
Figura 3	Interface de comunicação. A interface tem seu próprio protocolo que direciona e se comunica a um ou mais protocolos de aplicação	18
Figura 4	O conceito de Data Stream para a abstração do transporte de dados.....	19
Figura 5	O padrão Publish/Subscribe. Retirado de [1]	20
Figura 6	Comunicação entre Publishers e Subscribers por Data Stream	21
Figura 7	Exemplo de gerenciamento de um broker.....	22

LISTA DE TABELAS

Tabela 1	As camadas e e suas funções	15
----------	-----------------------------------	----

SUMÁRIO

	INTRODUÇÃO	10
1	INDÚSTRIA 4.0 E INTERNET DAS COISAS.....	11
1.1	Internet das Coisas.....	11
1.2	As Camadas do IoT.....	12
1.2.1	Aquisição	12
1.2.2	Transmissão	13
1.2.3	Aplicação	14
1.3	Camadas de Rede	14
2	A INTERFACE E SUA LIGAÇÃO COM IOT	16
2.1	Algumas tecnologias em IoT	16
2.1.1	RFID.....	16
2.1.2	Redes NB	16
2.1.3	Low Energy Bluetooth.....	16
2.1.4	Base TCP/IP.....	17
2.2	A Interface.....	17
3	O PROJETO	19
3.1	Camada de Abstração	19
3.2	Publishers e Subscribers	19
3.3	A implementação.....	21
3.3.1	MQTT	21
3.3.1.1	Broker.....	22
	CONCLUSÃO	23
	REFERÊNCIAS.....	24

INTRODUÇÃO

O cenário atual do desenvolvimento tecnológico encontra-se no meio de uma quarta revolução industrial. Nunca se produziu tantos dados e utilizou-se redes como a própria internet para propaga-los. É de se esperar que tanto o cenário acadêmico e o próprio mercado demandem inovações para o compartilhamento desses dados em tempo real ou próximo disso. Fazendo aquecer o mercado que engloba transporte, análise e inteligência de dados.

Essa revolução possui um nome, Indústria 4.0. Ela engloba todas as áreas que lidam com dados, da análise à rede que distribui os dados. E dentre estas áreas complexas, que envolvem quase todos os subgrupos da engenharia elétrica, encontra-se a Internet das Coisas, ou IoT, como iremos nos referenciar nesta dissertação.

A Internet das Coisas é a rede ou sistema que adquire, compartilha e aplica dados de dispositivos previamente equipados para medir e divulga-los. Ela é derivada métodos de comunicação entre máquinas e telemetria. Pode ser dissecada em três camadas de aquisição, comunicação e aplicação destes dados e pode ser implementada utilizando diversos protocolos de comunicação, dependendo da tecnologia disponível. É importante que o sistemas IoT deva ser construído de forma a atender a aplicação de uma forma eficiente, porém tal tarefa não é fácil nem simples.

Este projeto propõe uma interface que fornece comunicação entre as diferentes tecnologias e camadas de rede, de forma que o usuário só se preocupe em implementar e configurar a interface para mapear a melhor opção de ferramentas para a aplicação. O projeto lida com protocolos baseados na pilha TCP/IP, uma unanimidade em redes que se comunicam com a internet. Podendo se estender para outras protocolos de aplicações de escopo fechado. O foco está no protocolo de aplicação MQTT (Message Queuing Telemetry Transport), um protocolo que trabalho em cima do TCP/IP, leve e extremamente utilizado para compartilhamento dados telemétricos, de estado e de pequenas mensagens. Oferecendo uma API para tanto a aquisição assim como o recebimento e armazenamento destes dados.

1 INDÚSTRIA 4.0 E INTERNET DAS COISAS

A revolução dos dados atingiu praticamente todas as áreas de engenharia elétrica, desde a eletrônica, desenvolvendo dispositivos capazes de receber dados telemétricos, processá-los e enviá-los para demais hubs, a servidores de armazenamento de dados, recorrentemente chamados de Data Warehouses. Esse conjunto de mudanças engloba a Indústria 4.0, uma indústria que capta dados de suas máquinas em tempo real em larga escala, analisa, armazena, e utiliza inteligência artificial e estatística, para tomada de decisões estratégicas, contando sempre, é claro, com ajustes humanos.

1.1 Internet das Coisas

”A Internet das Coisas tem o potencial de mudar o mundo. Assim como a Internet fez. Talvez até mais” [2]. Uma tradução livre de Rampim [3] da frase de Kevin Ashton, cofundador do Auto-ID Center, em 1999. Apesar de ser um nome feito somente para chamar atenção, foi a primeira citação da expressão Internet das Coisas, e de lá vingou.

Dentre o meio da Indústria 4.0, encontra-se a internet das coisas ou IoT, responsável por estruturar as aplicações de aquisição, transmissão e armazenamento de dados a serem analisados. Não é uma surpresa que este setor envolva áreas como eletrônica, computação e telecomunicações em um pacote só. De fato suas camadas são mundos diferentes interligados a um propósito : transmitir dados sobre um dispositivo e/ou para um dispositivo em tempo real. Segundo a Cisco IBSG [4], há mais objetos conectados que a própria população mundial, fazendo com que o ano de 2009 ser considerado o ano de nascimento do IoT.

Pode-se definir IoT como a estrutura que comunica dispositivos em rede, permitindo a transmissão de dados sobre estes em tempo real. É a ponte que permite a troca de informações sobre um dispositivo, qual seu status, seu desempenho, suas condições físicas e do ambiente ao seu redor. Mas, para que este ciclo esteja completo é necessário camadas que desempenham tarefas específicas, para que o dado chegue a quem ou a o que está esperando.

1.2 As Camadas do IoT

Semelhante as camadas de rede, as camadas de IoT também exercem funções específicas no transporte de dados, e a camada acima não necessariamente precisa saber como a inferior funciona, somente precisa dos dados que esta camada entrega e executar suas tarefas sobre estes até chegar ao destino especificado, formando uma estrutura de pilha como na figura Figura 1;

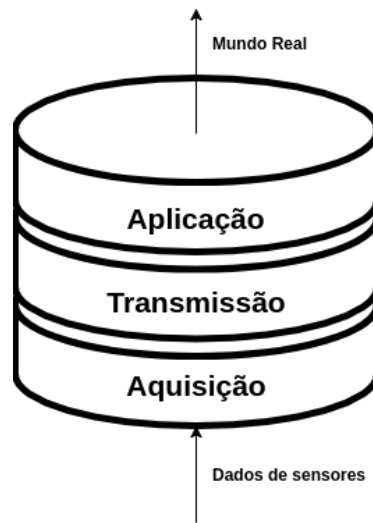


Figura 1 As três camadas do IoT, dos sensores ao mundo real

A primeira camada é a de aquisição de dados, que lida com o mundo físico e amostra estes dados através de sensores e conversores A/D, também realiza o processamento para entregar em um formato adequado para transmissão e entendível do outro lado, dependendo da aplicação. A segunda camada é a camada de transmissão, onde estão, efetivamente, as camadas de rede embutidas. Como o nome já denuncia, ela lida com os aspectos de rede e comunicação para que os dados cheguem aos seus destinos. E por último temos a camada de aplicação, a mais abrangente e que envolve maior poder computacional. Ela recebe os dados e lida com os processos de aplicação destes dados, seja análise, visualização, armazenamento ou a estruturação destes.

1.2.1 Aquisição

A etapa de aquisição está inserida diretamente no contexto de dados físicos, geralmente são hardwares menos complexos, focados em processamento de dados e entrada e

saída com conversão analógico-digital. Se comunicam com sensores ou centrais de controle lógico. São responsáveis por:

- Receber dados de sensores;
- conversão A/D;
- Processamento e calibragem de valores;
- Envio de dados em tempo real;

Para atender essas tarefas, não é necessário grande poder de processamento, micro-controladores ou microprocessadores são capazes de atender tais necessidades se acompanhados de módulos de rede e portas I/O, assim como a implementação do software. Veremos dois exemplos no capítulo de implementação do projeto, que utilizam tanto MCU (Micro-Controller Units) ou Consoles com Sistemas Operacionais leves.

1.2.2 Transmissão

Esta camada é o coração do IoT. A forma de transmissão define quais dispositivos eletrônicos e qual sua especificação técnica necessária para os quesitos de transmissão. Também define como os softwares da camada de aplicação e aquisição devem ser implementados baseado na estrutura da pilha de rede que será usada para transmitir.

Na próxima seção, veremos sobre a camada de rede e suas diversas formas de implementação. É importante que esta camada seja definida da melhor forma a atender sua aplicação, atendendo aspectos:

- quantidade de dados transmitido;
- número de acessos;
- distância entre dispositivos;
- segurança;

1.2.3 Aplicação

A camada de aplicação encabeça a pilha do IoT. É ela que de fato trata os dados e realiza as aplicações deste. Ela disponibiliza os dados para o mundo real, podendo exercer múltiplas funções simultâneas incluindo:

- Armazenamento e Análise;
- Visualização;
- Inteligência e aprendizado;
- Serviços e servidores;
- Gerenciamento e configuração;

Nesta camada estão presentes os endpoints apontados pela camada de aquisição, o destino dos dados. Bem assim como os servidores que gerenciam os clientes (geralmente implementados na camada de aquisição) e serviços e configurações oferecidos pelo sistema em si.

1.3 Camadas de Rede

Como visto anteriormente, a camada de transmissão basicamente define a infraestrutura do sistema. Ela é construído com as camadas de rede como base. Portanto definir as camadas de rede e seus protocolos é definir a camada de transmissão em si.

Redes de computadores são complexas com diferentes aspectos a se preocupar. Dividir em camadas permite modularizar a implementação da rede, de modo que cada camada tenha uma tarefa na estrutura de comunicação dos aspectos físicos ao software. Como a camada de cima não precisa saber sobre os detalhes e especificações da camada de baixo, as mudanças de uma parte do sistema é transparente para o resto do sistema. Existem diversas formas de implementação de camadas, mas todas se baseiam em um modelo de referência, o modelo OSI [5].

Baseado no modelo OSI. Temos o modelo TCP/IP [6], visualizado em Figura 2 e utilizado na internet e base para protocolos de aplicação muito utilizados como HTTP, WebSocket e MQTT. Em ambos cada camada exerce uma tarefa com seu respectivo protocolo, como resumido na tabela Tabela 1.

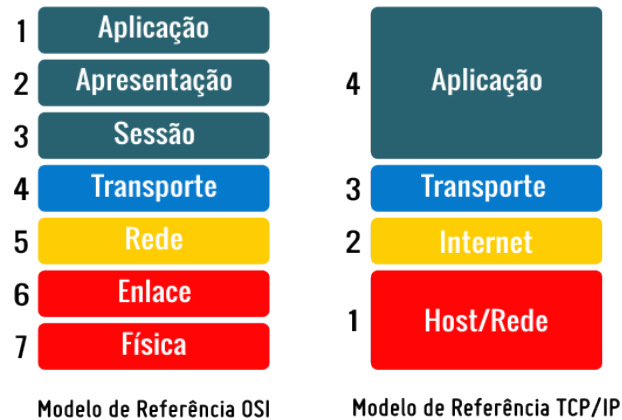


Figura 2 Diferenças entre OSI e TCP/IP em suas camadas

Tabela 1 As camadas e e suas funções

CAMADA	DETALHES
7 - Aplicação	Define instruções específicas da aplicação
6 - Apresentação	Formatação dos dados, conversão dos dados
5 - Sessão	Negociação e conexão com outros nós, analogia
4 - Transporte	Oferece métodos para a entrega de end-to-end
3 - Rede	Roteamento de pacotes em uma ou várias redes
2 - Enlace	Detecção de erros;
1 - Física	Aspectos físicos da transmissão

O foco desta literatura está na camada de aplicação e suas funcionalidades. Apesar de diferentes tecnologias utilizarem diferentes camadas abaixo, serão específicas características de protocolos construídos em cima do TCP/IP, por seu uso na Internet e redes industriais, e que sejam eficientes em trocas de mensagens em tempo real.

2 A INTERFACE E SUA LIGAÇÃO COM IOT

No capítulo 1, foi visto as bases para se implementar um projeto de IoT. A área começou a receber fortes investimentos e atenção por volta de 2009 e desde então foram feitas consideráveis implementações utilizando tecnologias e protocolos diferentes. Neste capítulo serão apresentados algumas dessas variações, para fins de comparação e respaldo para importância e objetivo deste projeto.

2.1 Algumas tecnologias em IoT

Estas são algumas tecnologias que satisfazem as condições apresentadas para um sistema IoT, nem todas utilizam o protocolo TCP/IP, mas todas são capazes de fazer seus dispositivos comunicarem-se em tempo real levando em consideração seus alcances e escalabilidade.

2.1.1 RFID

As primeiras aplicações de IoT foram em laboratórios de aplicações de RFID [3], junto com códigos bidimensionais, para aplicações de identificação de objetos. Uma das soluções mais populares e de baixo custo de IoT utilizando Rádio frequência.

2.1.2 Redes NB

Redes que utilizam bandas restritas visando baixo consumo e distância de transmissão são a nova fronteira, as mensagens de IoT são geralmente curtas, dados telemétricos, status etc, logo estes protocolos mostram-se úteis para este tipo de aplicação. Já se encontram implementadas algumas redes como SigFox [7] e LoRa [8].

2.1.3 Low Energy Bluetooth

As novas gerações de Bluetooth consomem muito menos energia, o que tornaram a tecnologia viável para aplicações IoT. Geralmente, módulos Bluetooth são utilizados como beacons [9]. Pontos espalhados por uma região, no qual podem se comunicar com os módulos de dispositivos mobile ao se aproximar, oferecendo links para conteúdo e exclusividades.

2.1.4 Base TCP/IP

As tecnologias mais comuns de se encontrar em aplicações IoT, os protocolos construídos com base no TCP/IP são vastamente utilizados e possuem uma rede mundialmente distribuída, o que facilita o uso. Pode-se implementar uma gama de protocolos de aplicações, alguns mais eficientes que outros.

O protocolo mais simples seria o HTTP, altamente usado na internet, porém não é eficiente no consumo de energia por abrir uma conexão a cada envio de dados. Para minimizar estas desvantagens, foi desenvolvido o CoAP [10] protocolo nos mesmos moldes do HTTP com o modelo REST, porém mais simples, mais leve, com baixo overhead e utilizado em redes locais.

Mas os mais utilizados em aplicações são sem dúvidas os protocolos que mantêm conexão aberta, em especial Websocket e MQTT, sendo o primeiro mais utilizado para chats e mensagens e o segundo domina o mundo do M2M e Telemetria.

2.2 A Interface

Inicialmente, os conceitos e ideias do projeto eram voltados a desenvolver uma interface no qual um desenvolvedor poderia implementar um sistema IoT de ponta a ponta utilizando APIs que direcionariam para um desses protocolos da seção 2.1, porém as diferenças entre os protocolos e as camadas de base, fazem com que esta solução esteja mais distante. Então o foco voltou-se para tecnologias que tenham base na pilha TCP/IP, por sua vasta implementação nas redes industriais e residências e na Internet.

Neste projeto iremos ver a implementação desta interface para o protocolo MQTT, cuja escolha será justificada adiante. Serão descritas as interfaces para as três camadas, que são de baixo custo, open-source e altamente escaláveis para construir outras aplicações com esta como base.

Para abstrair as camadas e aplicação, a interface deve estar dentro dos requisitos do IoT, bem assim como apresentar uma estrutura que se traduza aos protocolos baseados TCP/IP, para isso, algumas características fundamentais podem ser destacadas como base para um sistema IoT descritos.

- Full-Duplex. Capaz de receber e enviar mensagens ao mesmo tempo;
- Multicast. Capaz de enviar mensagens um ou mais dispositivos simultâneos;

- Envio de mensagens em tempo real;

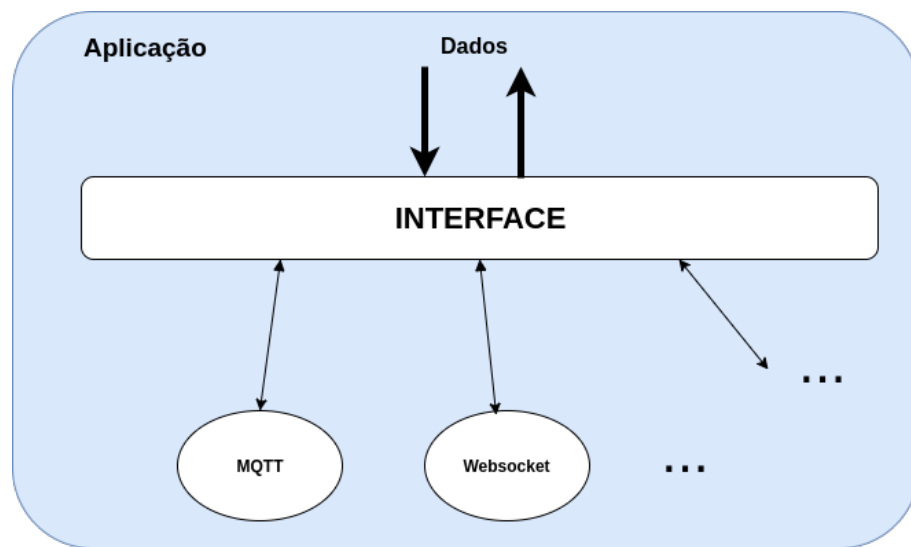


Figura 3 Interface de comunicação. A interface tem seu próprio protocolo que direciona e se comunica a um ou mais protocolos de aplicação

Essas características estão dentro do escopo apresentado no capítulo 1, a estratégia foi direcionada para protocolos que atendessem esse pré-requisito, assim construindo uma camada de abstração da interface por cima dos protocolos que contemplam essas características como mostra a figure Figura 3, o que será mais detalhado no próximo capítulo.

3 O PROJETO

Os dois últimos capítulos descreveram o conceito de Internet das Coisas e as especificações que o projeto deve contemplar, visando sempre ajudar na construção de sistemas IoT que melhor se encaixem na aplicação. Neste capítulo serão descritos as implementações do projeto, apresentado os motivos das escolhas de tecnologias e protocolos especificados. E terminando sobre persistência de dados em aplicações IoT e por quê a escolha de implementação bancos de dados é importante para a aplicação.

3.1 Camada de Abstração

Devido a interação entre dispositivos de aquisição de dados e aplicação e armazenamento de dados, foi necessário uma implementação de um protocolo de comunicação único entre os dispositivos e implementação em cada um destes em suas diferentes linguagens de programação.

O protocolo consiste em uma abstração de um canal de envio de dados chamado Data Stream mostra do em Figura 4, no qual passam dados após realizar um processamento dos dados em uma determinada velocidade podendo conter um limite de pacote de dados. Nas pontas desse canal estão os Publishers e Subscribers, que serão descritos adiante. Este conceito é uma forma de abstrair, unificar e simplificar a forma de transporte de dados, de uma modo que a interface possa ter o controle sobre os aspectos de transmissão. Cada protocolo na camada de aplicação, implementa este conceito de uma certa forma, porém o desenvolvedor não precisará se preocupar com estes detalhes.

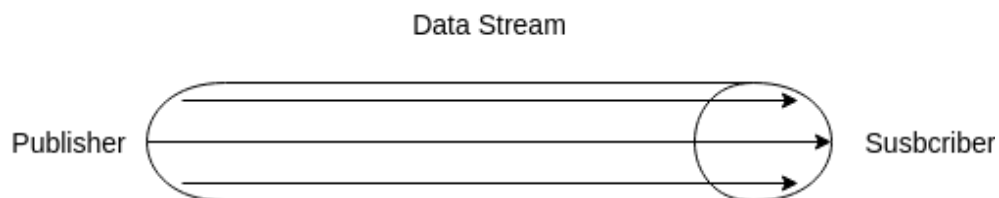


Figura 4 O conceito de Data Stream para a abstração do transporte de dados

3.2 Publishers e Subscribers

Para enviar e receber dados de uma forma a atender os requisitos da seção 2.2, foi utilizado um padrão de comunicação recorrente em aplicações contemporâneas, o padrão

Publish/Subscribe [1].

O padrão Publish/Subscribe permite que as mensagens sejam transmitidas assíncronas e para vários dispositivos simultaneamente. Para transmitir uma mensagem, um client pode simplesmente enviar uma mensagem para o tópico que os envia imediatamente para todos os subscribers. Todos os componentes que se inscreverem no tópico receberão todas as mensagens transmitidas, a menos que uma política de filtragem de mensagens seja definida pelo assinante.

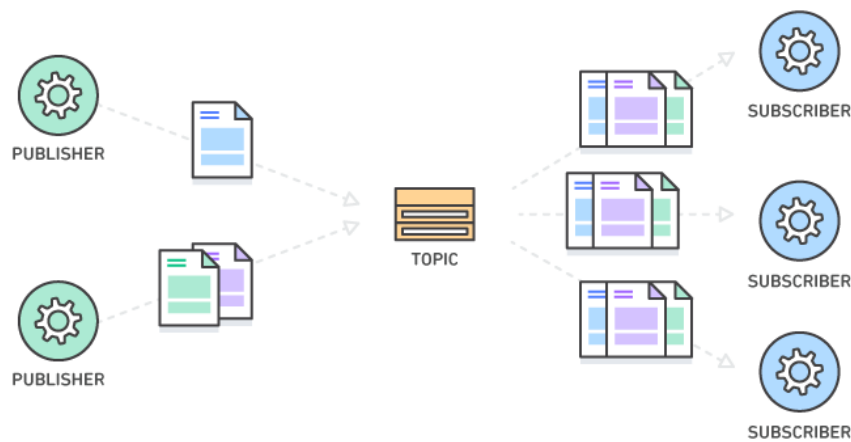


Figura 5 O padrão Publish/Subscribe. Retirado de [1]

Qualquer mensagem publicada em um tópico é imediatamente recebida por todos os subscribers do tópico. As mensagens de podem ser usadas para arquiteturas orientadas a eventos ou para desacoplar aplicativos, aumentando o desempenho, a confiabilidade e a escalabilidade. Com isso, foram criados duas funções possíveis para cada dispositivo dentro deste padrão, os Publishers e os Subscribers, sua comunicação é descrita em Figura 6.

Publishers são dispositivos que criam Data Stream e enviam dados por estes, regulam o processamento dos dados estipulam limites de tamanho de cada pacote de dado e determinam o intervalo de envio de pacotes. O protocolo permite que estes enviem os dados e também permite que outros dispositivos possam passar configurações remotamente para modificar os parâmetros de cada Data Stream, como o intervalo de envio ou outra configuração criada pelo tipo de Data Stream implementado.

Subscribers estão na outra ponta recebendo os dados, são capazes de enviar as configurações do Data Stream para os Publishers a chegada destes dados como um dri-

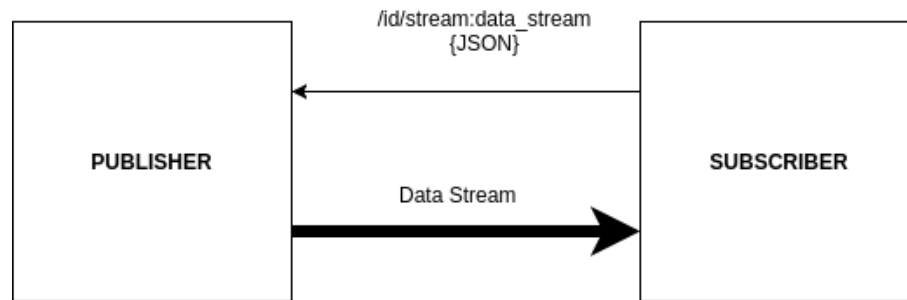


Figura 6 Comunicação entre Publishers e Subscribers por Data Stream

ver para a aplicação. Essas funcionalidades foram implementadas Orientadas a Objeto e são escaláveis para aplicações mais complexas que serão implementadas para o uso dos sistemas em aplicações de sensoramento e visualização dos dados.

3.3 A implementação

Existem protocolos de aplicação que podem ser facilmente mapeados por esse tipo de interface. Alguns já são construídos no modelo Publish/Subscribe, outros em modelos parecidos. Para critérios de comparação, apresentam-se dois protocolos de aplicação, MQTT e WebSockets [11]. O protocolo MQTT, no qual será implementado nesse projeto, foi moldado no padrão citado, enquanto o WebSockets é orientado a eventos. Ambos enviam mensagens em tempo real através de um servidor que controla o fluxo de mensagens da aplicação.

Na questão de desempenho, estes protocolos se mostram mais eficientes dentre os construídos sobre TCP/IP, comparado, por exemplo, com o HTTP, protocolo de rede mais recorrente em redes locais e na internet. Como mostrado em [12] e [13], HTTP, por sua natureza de abrir e fechar conexão a cada requisição de dados e seu cabeçalho, requer mais banda e consome mais energia que protocolos leves e de conexão persistente como os dois acima. O que faz sua escolha remota para a aplicação deste projeto.

3.3.1 MQTT

O protocolo MQTT [14] foi utilizado escolhido por ser leve e ideal para aplicações em tempo real com vários dispositivos simultaneamente. É um protocolo no padrão Publish/Subscribe ideal para definir a função de cada dispositivo seja enviando dados (Publish) ou recebendo estes (Subscribe).

Para gerenciar os clients (responsáveis pela implementação da comunicação MQTT) em cada dispositivo é necessário um servidor chamado Broker. Este foi implementado com o Mosquitto [15], um broker open source e leve capaz de ser instalado localmente e no servidor do laboratório para testes remotos.

3.3.1.1 Broker

O Broker é o servidor do padrão Publish/Subscribe, ele efetivamente executa as ordens de publicação (publish) feita por algum cliente para os tópicos que outros clientes estão inscritos (subscribed), possui todas as listas de tópicos, é orientado a conexão e não persiste informações dos clientes, ou seja, em caso de queda de conexão, estes devem se inscrever novamente nos tópicos.

Nota: A arquitetura Broker não é exclusividade do MQTT, outros protocolos utilizam esse tipo de implementação em servidores.

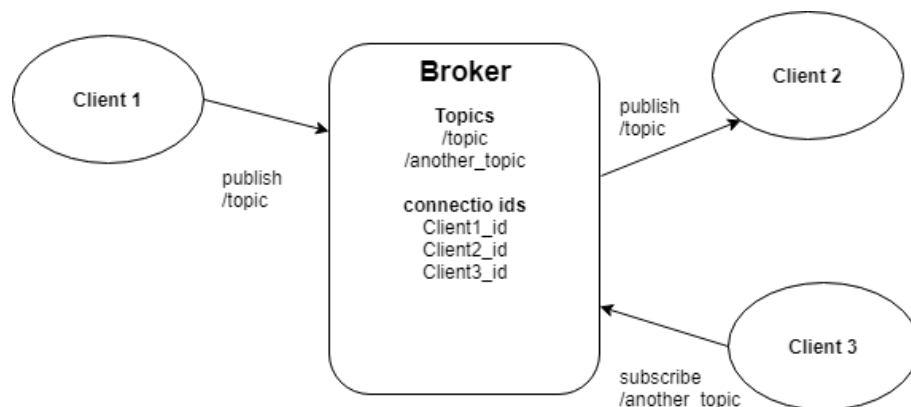


Figura 7 Exemplo de gerenciamento de um broker

Na figura Figura 7, o Broker armazena os tópicos e os ids de conexão dos clientes, 2 estava inscrito para ouvir as mensagens do tópico *topic*, enquanto 3 enviava uma ordem de inscrição em *another_topic*, 1 envia ordem de publicação para *topic*.

CONCLUSÃO

Aqui entra sua conclusão!!

REFERÊNCIAS

- [1] AWS. *Pub/Sub Messaging Asynchronous event notifications*. Disponível em: [<https://aws.amazon.com/pt/pub-sub-messaging/>](https://aws.amazon.com/pt/pub-sub-messaging/). Acesso em: 21/07/2018.
- [2] ASHTON, K. That 'internet of things' thing. v. 22, p. 97–114, 01 2009.
- [3] DIAS, R. R. de F. *Internet das Coisas sem Mistérios*. [S.l.]: Netpress Books, 2016.
- [4] EVANS, D. *The Internet of Things How the Next Evolution of the Internet Is Changing Everything*. [S.l.], 04 2011.
- [5] ZIMMERMANN, H. Innovations in internetworking. In: PARTRIDGE, C. (Ed.). Norwood, MA, USA: Artech House, Inc., 1988. cap. OSI Reference Model—The ISO Model of Architecture for Open Systems Interconnection, p. 2–9. ISBN 0-89006-337-0. Disponível em: <http://dl.acm.org/citation.cfm?id=59309.59310>.
- [6] COMER, D. E. *Internetworking with TCP/IP - Volume I: Principles, Protocols and Architecture*. 3. ed. [S.l.]: Prentice Hall, 1995.
- [7] Sigfox. *Sigfox Technology Overview*. Disponível em: <https://www.sigfox.com/en/sigfox-iot-technology-overview>. Acesso em: 21/07/2018.
- [8] LoRa Alliance, Inc. LoRaWAN™ 1.0.3 Specification. 2017.
- [9] Endeavor Brasil - Time De Conteúdo. *Beacon: o GPS que ajuda sua marca a localizar as melhores oportunidades*. Disponível em: <https://endeavor.org.br/estrategia-e-gestao/beacon/>. Acesso em: 21/07/2018.
- [10] CoAP. *Constrained Application Protocol (CoAP)*. Disponível em: <http://coap.technology/>. Acesso em: 21/07/2018.
- [11] Mozilla Developer Network. *WebSockets*. Disponível em: https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API. Acesso em: 21/07/2018.

- [12] YOKOTANI, T.; SASAKI, Y. Comparison with http and mqtt on required network resources for iot. 01 2017.
- [13] NAIK, N. Choice of effective messaging protocols for iot systems: Mqtt, coap, amqp and http. 10 2017.
- [14] IBM. *MQTT*. Disponível em: <<http://mqtt.org/>>. Acesso em: 21/07/2018.
- [15] Eclipse. *Mosquitto*. Disponível em: <<https://mosquitto.org/>>. Acesso em: 21/07/2018.