Salami, Dariush; Streibel, Olga; Rhenius, Marcus; Sigg, Stephan

A FAIR Extension for the MQTT Protocol

Published in:
Proceedings of 16th International Conference on Mobility, Sensing and Networking (MSN 2020)

DOI:
10.1109/MSN50589.2020.00019

Published: 01/04/2021

Document Version
Peer-reviewed accepted author manuscript, also known as Final accepted manuscript or Post-print

Please cite the original version:
Salami, D., Streibel, O., Rhenius, M., & Sigg, S. (2021). A FAIR Extension for the MQTT Protocol. In *Proceedings of 16th International Conference on Mobility, Sensing and Networking (MSN 2020)* (pp. 10-16). Article 9394261 IEEE. https://doi.org/10.1109/MSN50589.2020.00019

A FAIR Extension for the MQTT Protocol

Dariush Salami
Department of Communications and Networking
Aalto University, Finland
dariush.salami@aalto.fi

Olga Streibel
Disruptive Technologies
Chemical-Pharmaceutical Development
Bayer AG
olga.streibel@bayer.com

Marcus Rhenius
Information Technology
Product Supply, Pharmaceuticals
Bayer AG
marcus.rhenius@bayer.com

Stephan Sigg
Department of Communications and Networking
Aalto University, Finland
stephan.sigg@aalto.fi

***Abstract*— We address the realization of the Findability, Accessibility, Interoperability, and Reusability (FAIR) data principles in an Internet of Things (IoT) application through a data transfer protocol. In particular, we propose an architecture for the Message Queuing Telemetry Transport (MQTT) protocol that validates, normalizes, and filters the incoming messages based on the FAIR principles to improve the interoperability and reusability of data. We show that our approach can significantly increase the degree of FAIRness of the system by evaluating the architecture using existing maturity indicators in the literature. The proposed architecture successfully passes 18 maturity indicators out of 22. We also evaluate the performance of the system in 4 different settings in terms the latency and dropped messages in a simulation environment. We demonstrate that the architecture not only improves the degree of FAIRness of the system but also reduces the dropped messages rate.**

***Index Terms*— FAIR data principles, mqtt protocol, iot, data transfer protocol**

I. Introduction

With the advance of wireless communication technologies like the fifth generation of cellular networks (5G), a great variety of Internet of Things (IoT) devices have become ubiquitous all round the world. These devices are generating a vast amount of data which can be integrated and used for different purposes. Activity recognition [25], respiration detection [17], fall detection [16] and emotion detection [7] are a few examples of applications using sensor data to provide end users with different services.

Although the significant progress of machine learning and deep learning approaches has opened new doors in sensor-based applications, the data-hungry nature of these techniques is still a challenge in the literature. Transfer learning [29], zero-shot learning [26], and meta learning [22] are a few directions trying to tackle the need for a large amount of data in deep learning tasks from a model-based perspective.

From a data-based perspective, the FAIR (Findable, Accessible, Interoperable, Reusable) data principles [27] can be utilized to increase the data interoperability and re-usability, which can lead to an increase in the availability of sensor-based data for various tasks. Apart from an application point of view, adhering to the FAIR data principles can also increase the reproducibility and replicability of experiments in many domains [11], [13], [23]. However, due to the level of abstraction in these principles, realizing them in a real-world data landscape is complicated, especially when it comes to the IoT world.

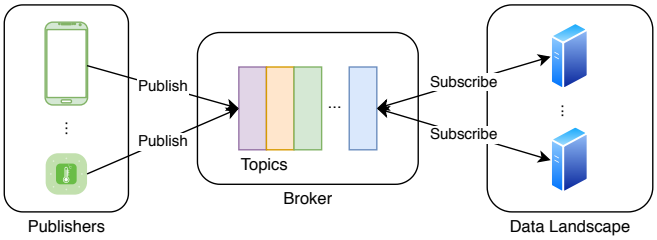


Fig. 1: An IoT system schematic and the role of the MQTT protocol

In IoT ecosystems, the incoming data is heterogeneous by the nature. Different devices and vendors can have their standards including data formats, licenses and even metadata attached to each piece of data that makes them highly challenging to integrate and re-use. The FAIR data principles can play a crucial role in integrating data in a way that they can be unified and normalized to be easily accessible for either human or machines. Finally, we can build different applications including deep-learning based, on top of those data.

As it is shown in Fig. 1, the first step in a FAIR IoT ecosystem is a protocol that facilitates this process. All the data that are coming from different IoT devices should go to the data landscape through a standard protocol. If this protocol has a set of FAIR-realizing features that improve the four building blocks of the principles, the rest of the landscape will be able to use the incoming data in an interoperable and reusable way.

Message Queuing Telemetry Transport (MQTT) [1], Advanced Message Queuing Protocol (AMQP), Constrained Application Protocol (CoAP) [18], Extensible Messaging

and Presence Protocol (XMPP), Data Distribution Service (DDS), and Lightweight Machine to Machine (LwM2M) protocol are common choices for data transfer protocols in the IoT world. Although they have a few features in common, the use cases are different. For example, XMPP is not a good choice for power constrained devices because of the overhead of processing XML messages.

We find in our analysis that among these protocols, MQTT is most suitable and flexible for IoT based applications [10]. Consequently, we propose an extension for the MQTT protocol to ease the implementation of FAIR principles in an IoT ecosystem.

We show that by implementing the proposed architecture in a MQTT-based IoT environment, we can realize the FAIR data principles with a reasonable degree. Our contributions are as follows:

1) We propose an architecture for the MQTT protocol to realize the FAIR data principles in an IoT data landscape.
2) We evaluate the proposed system using existing FAIR metrics in the literature
3) We evaluate the performance of the proposed system in a simulation-based environment

II. Related Work

In this section, we look at previous work on the FAIR data principles, especially from an implementation perspective and also a brief literature review of IoT data transfer protocols.

A. FAIR Data Principles

The FAIR Data Principles have been introduced in [6] to foster interoperability and re-use of data. In particular, they feature 15 facets corresponding to being **F**indable, **A**ccessible, **I**nteroperable, and **R**eusable (F-A-I-R). This is attempt to produce FAIR data is also mirrored by an demand of the European Commission to research institutions and individuals, to produce open data[1]. The FAIR principles intend to give 'a minimal set of community-agreed guiding principles and practices' [24] so that machines and humans and humans shall be able to find (F), access (A), interoperate (I) and re-use (R) data and metadata without effort.

Dunning et al. [5] have analyzed 37 repositories and reported that some of them have already implemented a few aspects of the FAIR data principles, but a considerable number of them are not realized in most of the repositories. They concluded that the implementation of these principles is desirable though, and that the implementation of basic policies (including the use of https and a clear license) is about the main hurdle of implementing the principles.

Since the security aspects of the principles are of high importance for a wide range of applications, Brewster et al. [3] proposed an ontology-based access control for FAIR data principles. This work takes into account the data, metadata, and additional metadata related to the users of the system to provide applications with secure and controlled access to object data in the system.

Rodríguez Iglesias et al. [14] proposed an exemplary methodology to migrate from a traditional data paradigm to the FAIR data principles-enabled repository for Pathogen-Host interaction dataset. They showed that a careful design decision could ensure FAIR data principles.

B. IoT Data Transfer Protocols

As it was mentioned in section I, there are a great variety of data exchange protocols in the IoT world. Numerous studies compare those protocols from different perspectives. For instance, T. Yokotani et al. presented a comprehensive comparison between Hyper Text Transfer Protocol (HTTP) and MQTT [28]. They have shown that MQTT is superior to HTTP in terms of required server resources, payload size, and overhead.

In another study [21], it has been shown that, while CoAP is well suited to communicating with systems designed for HTTP, MQTT is well-designed for sensor-based communication scenarios.

Another group of researchers has compared AMQP to MQTT in terms of packet transmission time in different scenarios [20]. They have shown that MQTT requires much less resources than AMQP because of its simplicity and because it is highly efficient for extremely low powered devices compared to AMQP.

Although the original MQTT protocol is based on the Transmission Control Protocol (TCP) transfer layer, Stanford et al. [19] proposed a User Datagram Protocol (UDP) based version of MQTT named MQTT-SN for sensor networks. For IoT applications, MQTT-SN is the preferred protocol, since the orignal TCP-based MQTT protocol suffers from an increased latency, as it guarantees reliability through an acknowledgment mechanism. Consequently, UDP based IoT data transfer protocols like CoAP or MQTT-SN feature a smaller latency compared to TCP-based MQTT. In applications for which low latency communication is required, MQTT-SN should be utilized.

MQTT does not provide any native authorization mechanism. However, in most of the IoT use-cases and applications, an authorization mechanism is vital. Niruntasukrat et al. [12] proposed an authorization mechanism based on OAuth, which is an open authorization standard. This mechanism can also be used to ensure the authentication and authorization in the FAIR data principles.

User privacy is another critical aspect when it comes to the IoT world. Markovic et al. [9] proposed a provenance stream processing technique through C-SPARQL to enhance the transparency of MQTT brokers. They showed that semantic solutions are feasible so that by processing the provenance records describing the actual behavior of a broker during message forwarding, we can distinguish user privacy violations.

[1]H2020 Guidelines on FAIR Data Management: http://ec.europa.eu/research/participants/data/ref/h2020/grants_manual/hi/oa_pilot/h2020-hi-oa-data-mgt$_e n.pdf$

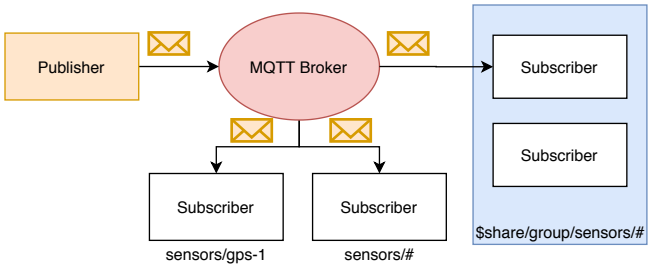


Fig. 2: Shared subscription in MQTT 5.0

III. Analysis of the MQTT Protocol

MQTT is a protocol for device to device communication that operates based on the publish/subscribe communication model and originally uses TCP as its transport layer protocol. As it is shown in Fig. 1, a publisher can publish a message in a queue with a particular topic name. This procedure makes it feasible for the subscribes to only listen to the topics that they need. Moreover, this communication model makes MQTT scalable by decoupling different parts of the system.

A. Specifications

In this section, we introduce some of the useful features of the MQTT protocol that we use as the building blocks of the proposed architecture.

1) Quality of Service (QoS): The MQTT protocol has three different levels of Quality of Service (QoS) for each message and each subscription.

- QoS 0 (at most once): the sender (publisher or broker) sends the message and 'forgets' about it. The message will be delivered to the recipient (broker or subscriber) at most once.
- QoS 1 (at least once): there is a light-weight acknowledgment mechanism between the sender and the receiver that guarantees the message delivery at least once.
- QoS 2 (exactly once): this is the most strict QoS level in MQTT that has a full delivery mechanism ensuring the message delivery exactly once. This mechanism makes the approach the reliable but at the same time increasres latency compared the other two QoS levels.

2) Hierarchical Topic Names: Moreover, MQTT supports hierarchical topic names that are separated using a slash sign. This feature allows subscribers to listen to topics with different granularity levels. Moreover, subscribers can use + and # signs to subscribe to a group of topics with single or multiple levels.

3) Shared Subscriptions: Shared subscription is a concept introduced in MQTT 5.0. As shown in Fig. 2, two or more subscribers can share a subscription in the way that the broker will deliver the messages to only one subscriber in the group based on a predefined policy. This feature is an ideal feature for load balancing use cases. We use the shared subscription concept in the proposed model.

	Aspects	Protocol Correspondence
Findability	F1	messages with UUID
	F2	metadata annotation
	F3	metadata annotation
	F4	✗
Accessibility	A1	protocol itself
	A1.1	open, free, and standard protocol
	A1.2	authentication and authorization
	A2	✗
Interoperability	I1	different data formats
	I2	✗
	I3	reference granuality
Reusability	R1	✗
	R1.1	data licensing
	R1.2	referencing
	R1.3	different data formats

TABLE I: A summary of the related aspects of the FAIR data principles to a protocol

B. MQTT from a FAIR Standpoint

We intend to use the MQTT protocol to provide features that play an essential role in an ecosystem which complies with FAIR data principles. In table I, we have identified the related concepts of a protocol to the aspects of the FAIR data principles [27]. A protocol with the following features can foster the FAIR implementation in an IoT ecosystem:

- A free, open, and standard protocol
- Authorization and authentication
- Supporting different data formats
- Annotating each message with a UUID
- Annotating each data with metadata
- Referring to a source message in another message
- Defining references with different granuality level
- Annotating a message with a license
- Unifying the format of the incoming data and metadata

MQTT is a free, open, and standardized protocol. However, it does not have advanced authentication and authorization mechanisms. Fortunately, a large number of papers address this issue in MQTT [12], [8], [2], [4] (see section II-B).

Moreover, MQTT puts no requirement on the data format that is going to be transferred in the system. Any data format like XML, JSON, plain-text, and binary can be transferred using this protocol. Nevertheless, for the rest of the mentioned requirements, there is no native or third-party mechanism in MQTT.

IV. A FAIR Principles Extension for the MQTT Protocol

In this section, we propose an extension for the MQTT protocol that addresses the requirements mentioned in section III-B. To ensure these requirements that are not tackled in the literature yet, there should be a middleware that not only redirects the data with non-fixable problems to another pipeline for further processing, but also applies a normalizations to align them with the FAIR data principles.

We propose a component named FAIR agent, which is a publisher and a subscriber at the same time in the MQTT

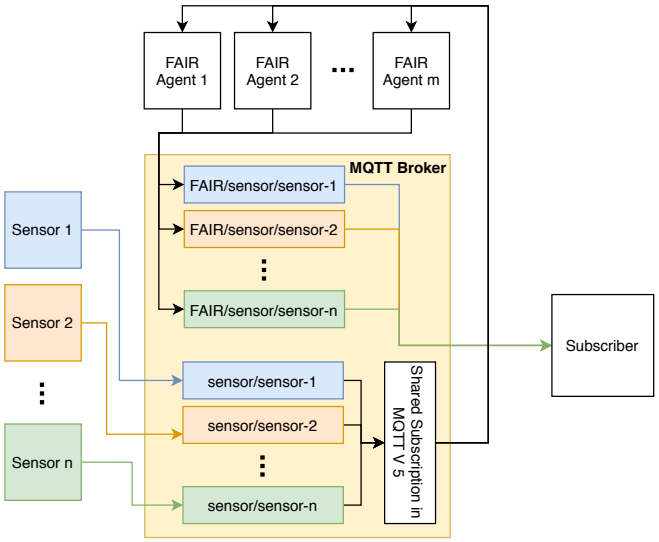


Fig. 3: The architecture of the proposed system

world. This agent is responsible for realizing the maturity indicators in [15]:

- Normalizing the unique identifiers of the incoming messages
- Unifying and normalizing the structure the incoming messages
- Filtering the incoming message if:
  - It is metadata without a valid source data identifier
  - It is data that is referring to a not valid source data identifier
  - It is data that is referring to a valid source data identifier without a valid relationship type
  - It is data without a valid license

The whole architecture is general enough to be applied to an arbitrary domain with a specific set of configurations.

Fig. 3 shows the architecture of the proposed system. n sensors are publishing their data to n different topics. Using the shared subscription, we connect those topics to m FAIR agents. Having unified and filtered the incoming messages based on the configuration, the agents publish the messages in the corresponding topic with a $FAIR$ prefix. Finally, the subscriber(s) listen(s) to the topics with $FAIR$ prefix.

The mechanism of the shared subscriptions and the FAIR agents are demonstrated through a flowchart diagram in Fig. 4. As it is shown in the figure, we use the random distribution policy of the shared subscription concept since we want to distribute the load equally between the FAIR agents. The heart of the proposed system is operation and decision making boxes inside a FAIR agent which were described before. Given the dynamic nature of the FAIR maturity indicators in [15], we keep the proposed system general enough to include new mechanisms in these two operation and decision making mechanisms.

Using the authorization and authentication mechanisms that we mentioned before, we make sure that the sensors are not able to publish to the topics with the $FAIR$ prefix

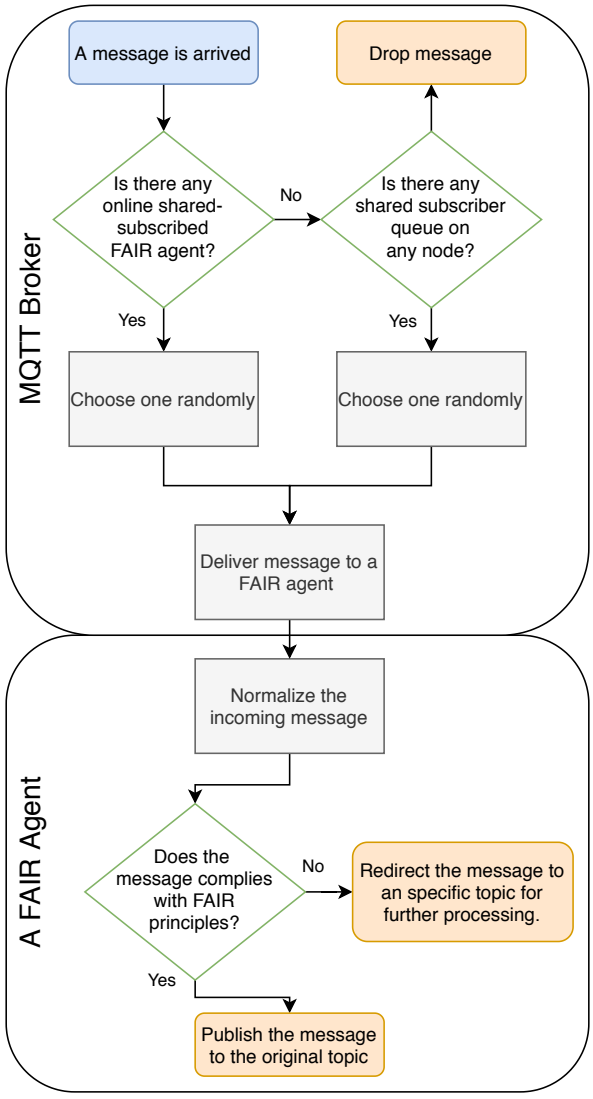


Fig. 4: The flowchart of the proposed system

directly, and the listeners are not able to listen to the topics without the $FAIR$ prefix. By doing so, the data that goes through the data landscape has a specific level of FAIRness.

A. FAIRness evaluation of the proposed architecture

Since realizing the FAIR principles in few concrete and measurable metrics is challenging, evaluating a system from that perspective is not easy. Recently, a group of researchers has proposed an open and community-based framework for evaluating FAIRness of repositories named FAIRSharing [15]. Currently, there are 24 metrics in the framework that can show the degree of the FAIRness of a data repository. In this section we evaluate the proposed system using those metrics in two scenarios. First, we imagine there is no FAIR agent in the system, so the data directly go to the landscape. Second, we add a FAIR agent to the system to validate, standardize, and filter the incoming data. As it is shown in Fig. 5, when there is no FAIR agent in the system, we only pass 4 tests out of 22 tests. But when we add a FAIR agent

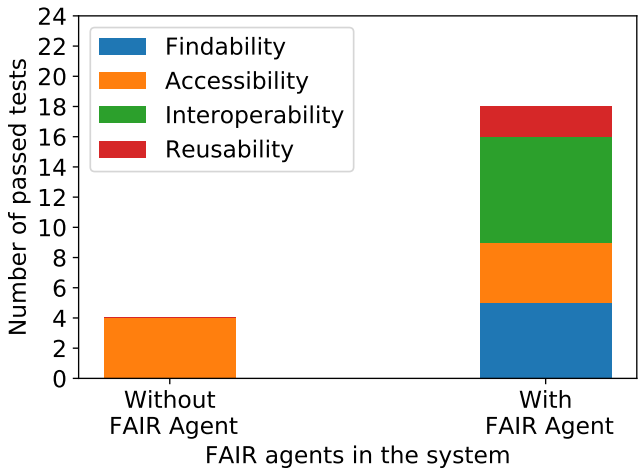


Fig. 5: Evaluation of the proposed architecture through FAIRSharing framework with 22 metrics

	Metric	Status
F	Identifier Persistence	✗
	Data Identifier Persistence	✗
	Searchable in major search engine	✗
	Unique Identifier	✓
	Structured Metadata	✓
	Grounded Metadata	✓
	Data Identifier Explicitly In Metadata	✓
	Metadata Identifier Explicitly In Metadata	✓
A	Open free protocol for data retrieval	○
	Open free protocol for metadata retrieval	○
	Data authentication and authorization	○
	Metadata authentication and authorization	○
	Metadata Persistence	✗
I	Metadata KRL (weak)	✓
	Metadata KRL (strong)	✓
	Data KRL (weak)	✓
	Data KRL (strong)	✓
	Metadata uses FAIR vocabularies (weak)	✓
	Metadata uses FAIR vocabularies (strong)	✓
	Metadata with qualified outward references	✓
R	Metadata Includes License (weak)	✓
	Metadata Includes License (strong)	✓

TABLE II: 22 evaluation metrics in the FAIRsharing framework and their status with respect to the proposed architecture (✓: addressed, ✗: not addressed, ○: not related)

to the system we manage to pass 18 tests all across the four categories.

All of the 22 metrics and their status in the proposed system is shown in table II. The ones shown with ✗ are not addressed because they are related to the infrastructure and not related to the protocol. The metrics whose status is shown with ○ are satisfied because of the protocol that we used for serving the data (HTTP). They are not directly related to the MQTT protocol.

B. Performance Analysis of the Proposed System

Although we use the shared subscription concept for load balancing between FAIR agents and to prevent the system from being a single point of failure, there is an overhead because of the FAIR agents. In this section, we analyze the effect of the agents on latency, and the drop rate when the sensors publish messages with QoS 0.

Setting	# Sn	# A	# Sb	QoS	# Msg	Speed	Err
I	8	0-8	1	0	100,000	-	20%
II	8	0-8	1	0	100,000	1000	20%
III	8	0-8	1	1	100,000	1000	20%
IV	8	0-8	1	2	100,000	1000	20%

TABLE III: Different settings for performance evaluation of the proposed system: #Sn: number of sensors (publishers), #A: number of agents, #Sb: number of subscribers, S QoS: sensor QoS level, #Msg: number of messages to send per each sensor, Speed: number of messages that each sensor sends per second, Err: error rate for the data which means that 20% of the data is inconsistent with the FAIR principles, so they should be dropped by the agents

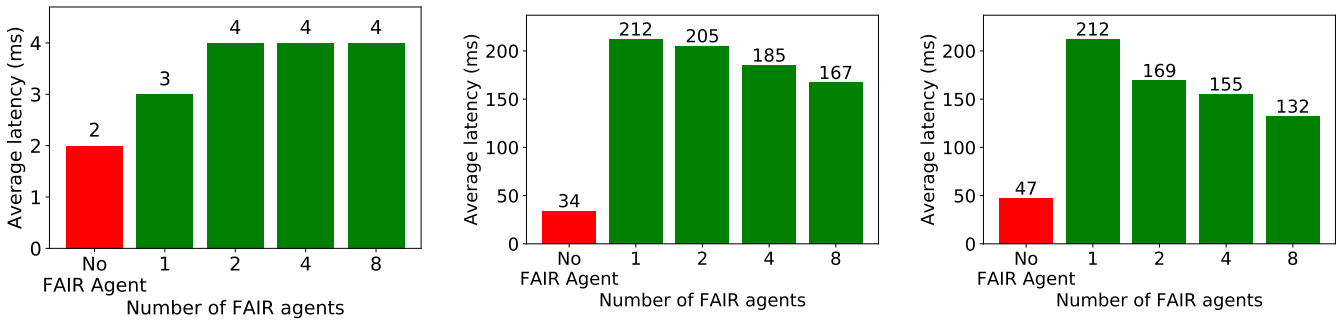
To evaluate the performance of the system, we develop a simulation environment. We use the HiveMQ implementation of the MQTT protocol that supports MQTT 5.0. In the environment, we have a server with 24GBs of RAM and Intel(R) Core(TM) i7-6700HQ CPU, and two clients with 16GBs of RAM and Intel(R) Core(TM) i7-7820HQ CPU. They are connected to each other on a Wireless Local Area Network (WLAN).

We install the MQTT broker on the server and started each sensor as a process on one of the clients and each FAIR agent, and subscriber as a process on the second client. We evaluate the performance of the proposed architecture in different scenarios. These settings are shown in table III. To test the proposed architecture, we create 100,000 messages 30% of which are metadata, and the rest are data. As it is shown in table III, 20% of these data are not fixable so they should be filtered by the FAIR agents. Based on the configuration file for the FAIR agents, the administrator of the system can decide whether to drop these messages or to redirect them to another topic for further processing.

To evaluate the overhead of the proposed FAIR agents, we compare the system with a system without FAIR agents. All generated data will be sent directly to the subscriber(s) when there is no FAIR agent in the system, even though they violate the FAIR principles.

As shown in Fig. 6, when there is no agent in the system, the latency is low. However, all the data, no matter whether it complies with the FAIR principles or not, would go to the data landscape. As it is shown in Fig. 6, when we introduce a single FAIR agent to the system the latency increases. In settings III and IV shown in Fig. 6b and 6c respectively, by adding more agents the latency decreases because the agents are working simultaneously and are able to handle more messages per second in comparison to the system with a single agent.

But in Fig. 6a that is for setting II, as we add more agents to the system, the latency increases. The reason is that the time delay for dispatching the messages through the shared subscription module is more than the gain itself. In other words, since the communication between the system's different components is fast and there is no acknowledg-



(a) Latency analysis for setting II

(b) Latency analysis for setting III

(c) Latency analysis for setting IV

Fig. 6: Latency analysis for different QoS levels: here the latency is the time needed for transferring a message from sensor to a subscriber

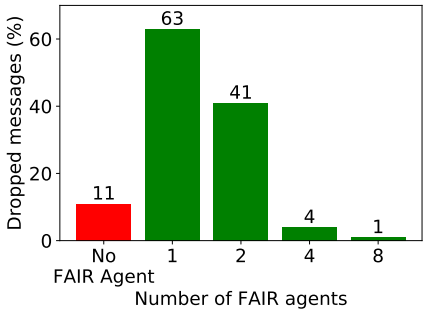


Fig. 7: Percent of the dropped messages in setting I

ment mechanism because of the QoS level, which is 0, the bottleneck of the system is the shared subscription module. As a result, as we increase the number of FAIR agents, we put more load on this module, which increases the latency. Although it is not useful in terms of latency, we can avoid a single point of failure system by adding more agents.

Moreover, we analyze the dropped messages with a different number of agents. In setting II, we have no dropped messages. To analyze this, we remove the delay between two consecutive messages sent by each sensor. In setting I, sensors try to send their messages as soon as possible without any delay. The QoS level in setting I is QoS 0, so a few messages would be dropped since there is no queue in this setting, and the receivers (agents and subscriber(s)) will not be able to handle all the messages immediately.

As it is shown in Fig. 7, when there is no FAIR agent, 11% of the messages is dropped. As we introduce a single FAIR agent to the system, the drop rate goes up to 63%. The reason is that the agent should deserialize the message, add a UUID, and validate it, which is time-consuming. Consequently, the agent will not be able to keep up with the pace of senders, and lots of the messages will be dropped. As we increase the number of FAIR agents not only we address the problem of being single point of failure, but also we decrease the rate of the dropped messages to 1% which is one-tenth of the system without a FAIR agent. In other words, as we increase the number of FAIR agents, we decrease the time required to process the messages since the FAIR agents are working simultaneously using the shared subscription concept. As a result, the number of lost messages drops significantly.

All in all, we analyzed the performance of the proposed system in terms of latency and the drop rate. Although, by introducing the FAIR agent to the system, the latency increases, these FAIR agents can play a crucial role in stabilizing the systems that are trying to realize and implement the FAIR data principles. Furthermore, we can even decrease the drop rate by adding FAIR agents.

V. Conclusion

We identified the related aspects of the FAIR principles to an IoT data transfer protocol. Then, we proposed an extension for the MQTT protocol that ease the implementation of the FAIR principles in an IoT ecosystem. We showed that by using the proposed architecture, we can increase the FAIRness degree of the system significantly and pass 18 FAIRSharing maturity indicator tests out of 22.

We also evaluated the performance of the proposed architecture in terms of the latency and the message drop rate in a simulation environment. We showed that although by introducing FAIR agents to the system, the latency, and the message drop rate increases, we can use the shared subscription concept to decrease that latency as well as decreasing the message drop rate. Moreover, we showed that we can make the architecture more reliable by introducing more FAIR agents to the system.

VI. Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 813999.

References

[1] Ken Borgendale Andrew Banks, Ed Briggs and Rahul Gupta. Message queue telemetry transport protocol (mqtt). Technical report, 2019.

[2] Ranbir Singh Bali, Fehmi Jaafar, and Pavol Zavarasky. Lightweight authentication for mqtt to improve the security of iot communication. In *Proceedings of the 3rd International Conference on Cryptography, Security and Privacy*, pages 6–12, 2019.

[3] Christopher Brewster, Barry Nouwt, Stephan Raaijmakers, and Jack Verhoosel. Ontology-based access control for fair data. *Data Intelligence*, 2(1-2):66–77, 2020.

[4] Marco Calabretta, Riccardo Pecori, Massimo Vecchio, and Luca Veltri. Mqtt-auth: A token-based solution to endow mqtt with authentication and authorization capabilities. 2018.

[5] Alastair Dunning, Madeleine De Smaele, and Jasmin Böhmer. Are the fair data principles fair? *International Journal of digital curation*, 12(2):177–195, 2017.

[6] Force11. Guiding principles for findable, accessible, interoperable and re-usable data publishing version b1. 0. 2014.

[7] Eiman Kanjo, Eman MG Younis, and Chee Siang Ang. Deep learning analysis of mobile physiological, environmental and location sensor data for emotion detection. *Information Fusion*, 49:46–56, 2019.

[8] Reto E Koenig, Lukas Laederach, and Cédric von Allmen. How to authenticate mqtt sessions without channel-and broker security. *arXiv preprint arXiv:1904.00389*, 2019.

[9] Milan Markovic and Peter Edwards. Enhancing transparency of mqtt brokers for iot applications through provenance streams. In *Proceedings of the 6th International Workshop on Middleware and Applications for the Internet of Things*, M4IoT '19, page 17–20, New York, NY, USA, 2019. Association for Computing Machinery.

[10] N. Naik. Choice of effective messaging protocols for iot systems: Mqtt, coap, amqp and http. In *2017 IEEE International Systems Engineering Symposium (ISSE)*, pages 1–7, October 2017.

[11] Engineering National Academies of Sciences, Medicine, et al. *Reproducibility and replicability in science*. National Academies Press, 2019.

[12] Aimaschana Niruntasukrat, Chavee Issariyapat, Panita Pongpaibool, Koonlachat Meesublak, Pramrudee Aiumsupucgul, and Anun Panya. Authorization mechanism for mqtt-based internet of things. In *2016 IEEE International Conference on Communications Workshops (ICC)*, pages 290–295. IEEE, 2016.

[13] Irene V Pasquetto, Bernadette M Randles, and Christine L Borgman. On the reuse of scientific data. *Data Science Journal*, 16:8, 2017.

[14] Alejandro Rodríguez-Iglesias, Alejandro Rodríguez-González, Alistair G. Irvine, Ane Sesma, Martin Urban, Kim E. Hammond-Kosack, and Mark D. Wilkinson. Publishing fair data: An exemplar methodology utilizing phi-base. *Frontiers in Plant Science*, 7:641, 2016.

[15] Susanna-Assunta Sansone, Peter McQuilton, Philippe Rocca-Serra, Alejandra Gonzalez-Beltran, Massimiliano Izzo, Allyson L Lister, and Milo Thurston. Fairsharing as a community approach to standards, repositories and policies. *Nature biotechnology*, 37(4):358–367, 2019.

[16] Guto Leoni Santos, Patricia Takako Endo, Kayo Henrique de Carvalho Monteiro, Elisson da Silva Rocha, Ivanovitch Silva, and Theo Lynn. Accelerometer-based human fall detection using convolutional neural networks. *Sensors*, 19(7):1644, 2019.

[17] Wonju Seo, Namho Kim, Sehyeon Kim, Chanhee Lee, and Sung-Min Park. Deep ecg-respiration network (deeper net) for recognizing mental stress. *Sensors*, 19(13):3021, 2019.

[18] Zach Shelby, Klaus Hartke, and Carsten Bormann. The constrained application protocol (coap). Technical report, 2014.

[19] Andy Stanford-Clark and Hong Linh Truong. Mqtt for sensor networks (mqtt-sn) protocol specification. *International business machines (IBM) Corporation version*, 1:2, 2013.

[20] Nguyen Quoc Uy and Vu Hoai Nam. A comparison of amqp and mqtt protocols for internet of things. In *2019 6th NAFOSTED Conference on Information and Computer Science (NICS)*, pages 292–297. IEEE, 2019.

[21] Henri W van der Westhuizen and Gerhard P Hancke. Comparison between coap and mqtt-server to business system level. In *2018 Wireless Advanced (WiAd)*, pages 1–5. IEEE, 2018.

[22] Joaquin Vanschoren. Meta-learning. In *Automated Machine Learning*, pages 35–61. Springer, Cham, 2019.

[23] Nicole A Vasilevsky, Jessica Minnier, Melissa A Haendel, and Robin E Champieux. Reproducible and reusable research: are journal data sharing policies meeting the mark? *PeerJ*, 5:e3208, 2017.

[24] Charles Vesteghem, Rasmus Froberg Brøndum, Mads Sønderkær, Mia Sommer, Alexander Schmitz, Julie Støve Bødker, Karen Dybkær, Tarec Christoffer El-Galaly, and Martin Bøgsted. Implementing the fair data principles in precision oncology: review of supporting initiatives. *Briefings in Bioinformatics*, 21(3):936–945, 2020.

[25] Jindong Wang, Yiqiang Chen, Shuji Hao, Xiaohui Peng, and Lisha Hu. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, 119:3–11, 2019.

[26] Wei Wang, Vincent W Zheng, Han Yu, and Chunyan Miao. A survey of zero-shot learning: Settings, methods, and applications. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 10(2):1–37, 2019.

[27] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. The fair guiding principles for scientific data management and stewardship. *Scientific data*, 3, 2016.

[28] Tetsuya Yokotani and Yuya Sasaki. Comparison with http and mqtt on required network resources for iot. In *2016 international conference on control, electronics, renewable energy and communications (ICCEREC)*, pages 1–6. IEEE, 2016.

[29] Fuzhen Zhuang, Zhiyuan Qi, Keyu Duan, Dongbo Xi, Yongchun Zhu, Hengshu Zhu, Hui Xiong, and Qing He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 2020.