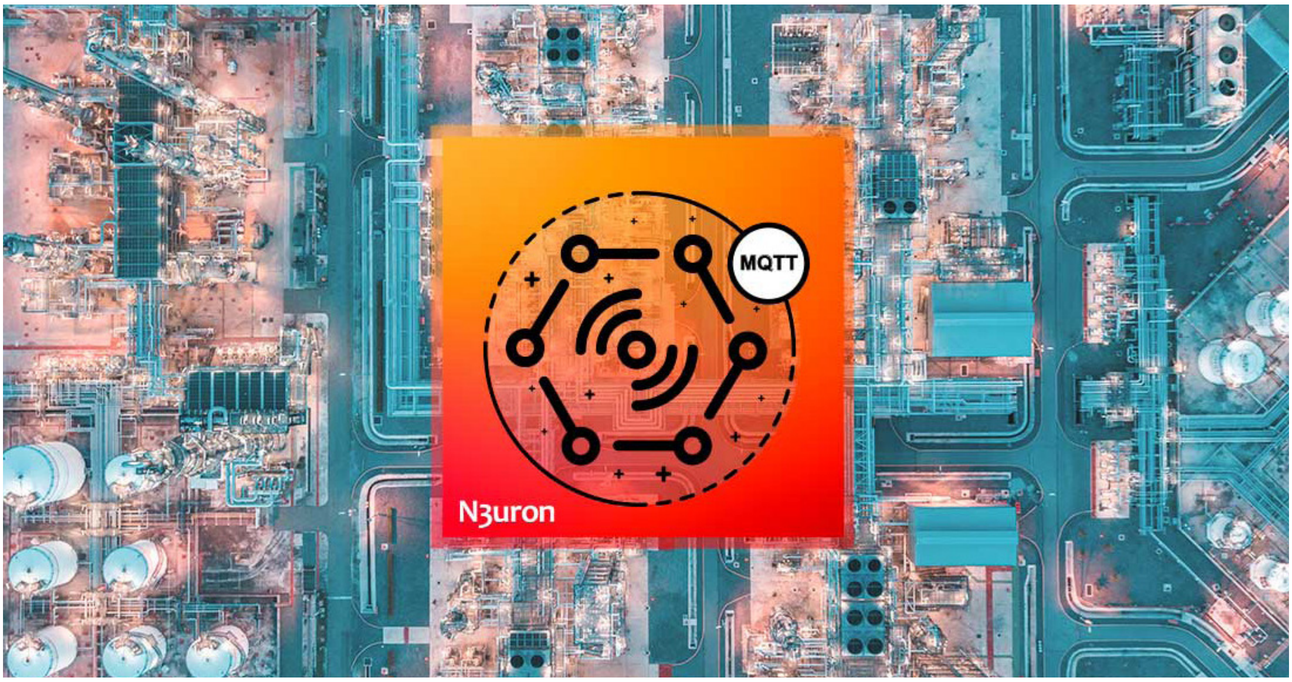
Author: José Granero and Pier Giorgio Parrini

MQTT: The universal messaging protocol for Cloud providers and IIoT systems



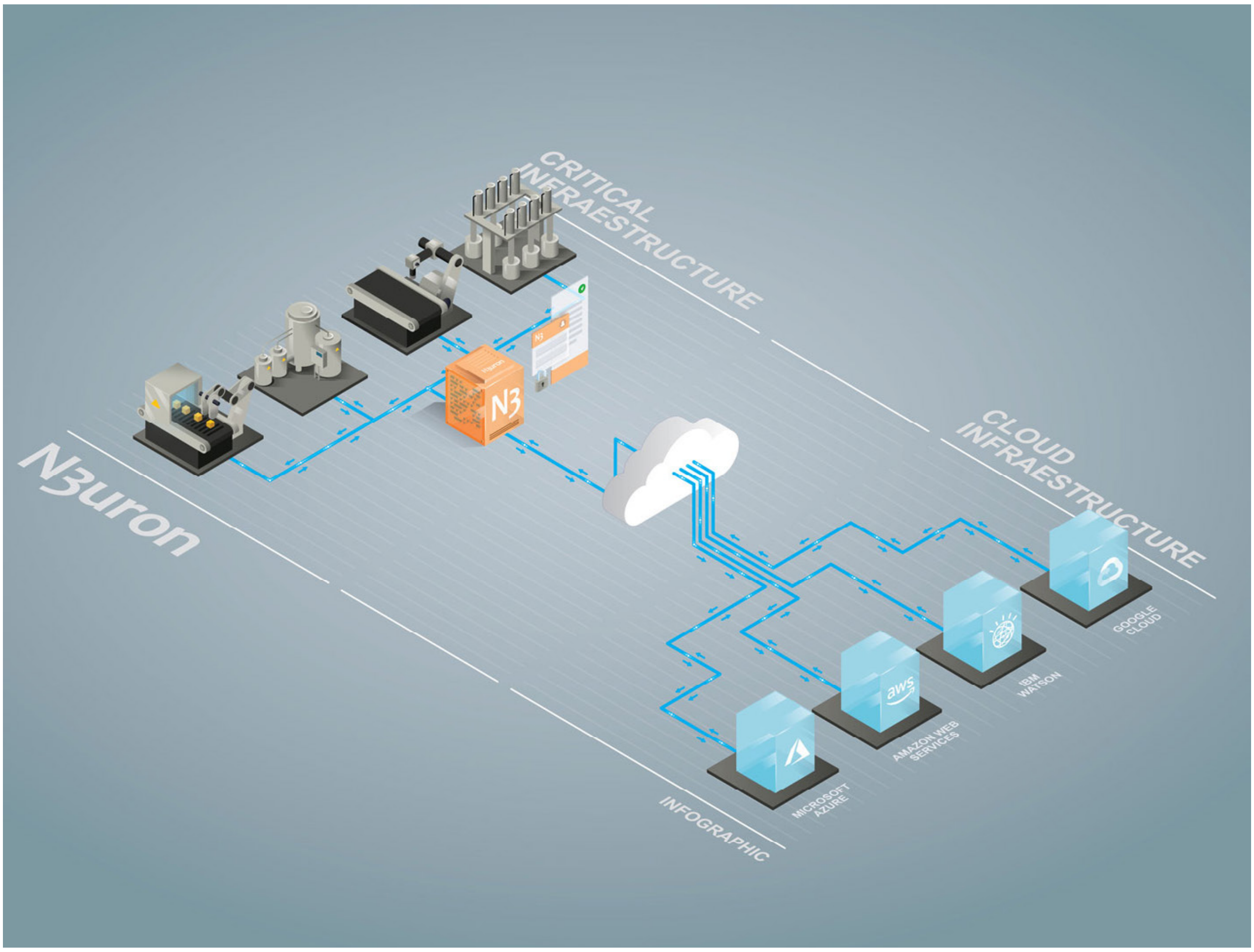
What is MQTT, and why is it ideal for IIoT?

If you work in the **Industrial IoT (IIoT) sector**, you probably have already heard about **Message Queuing Telemetry Transport (MQTT)**. Today it is largely used as a primary communication protocol for IIoT solutions, as well as for data exchange with the leading cloud service providers such as Microsoft Azure, AWS, Google Cloud or IBM Watson and even with social networks like Facebook.

In an ever-increasing automation scenario, where numerous data sources such as sensors, control devices and robots interact with manufacturing systems, and the inevitable digitalization process of companies demands vast volumes of data in order to improve their competitiveness, it is in the industrial space where the true value of the IoT can be found. The IIoT unlocks what were once isolated data silos by using open technologies to seamlessly share these data with any enterprise-level, cloud or third-party system in a secure, flexible and scalable way. One of the most important things to define in any IIoT solution is the communication protocol as it will be the foundation of the system. Most extended IIoT protocols are either publish-and-subscribe typeor poll/response . **MQTT falls into the first category, where clients connect and publish data to a topic on an intermediary broker**; on the other hand, clients can also subscribe to a topic in order to retrieve the data.

Publish/subscribe protocols decouple devices and allow applications to connect to middleware, thus being able to connect any application that requires data from any device without overloading the network. Instead of applications periodically checking devices for any changes, applications can connect to the broker and subscribe to the relevant data.

Regardless of whether industrial systems exchange data with analytics and visualization applications, such as the Supervisory Control and Data Acquisition (SCADA) system, or with a cloud platform, **all data can be transmitted efficiently and securely by using the MQTT protocol**. Built on top of TCP/IP, MQTT is an event-based, lightweight publish/subscribe messaging protocol designed for low bandwidth, high latency and unreliable networks. These outstanding features, together with the fact that it is open-source, have made it emerge as the de facto standard for IIoT and the best choice for transferring huge volumes of data to cloud services as well as integrating devices with third-party applications.



Representation of additional measures in an industrial environment using N3uron

What are the advantages of MQTT?

MQTT was invented by Arlen Nipper and Andy Stanford-Clark in 1999 to create a near real-time protocol with agnostic payloads, connecting data-producing devices such as programmable logic controllers (PLCs), remote terminal units (RTUs) and flow computers on oil and gas pipelines over satellite connection to multiple data consumers to include the SCADA control system and back-end enterprise resource planning (ERP) applications. The key requirements of the protocol were that it be:

- Very easy and simple to implement
- Lightweight and bandwidth efficient
- Data agnostic
- With continuous session awareness
- Quality of service data delivery

These goals are still the central core of MQTT and the reason why it has been so widely adopted for operations and bridging the gap between OT and IT technologies. **MQTT has three main features that make it the ideal IIoT protocol:**

- Low bandwidth
- Transport Layer Security (TLS)
- Stateful awareness

Limited bandwidth is a serious challenge to IIoT, especially for remote sites where unreliable communications are used. MQTT is an event-driven protocol, devices connect directly to the server (broker) and applications, also connected to the server, get updates whenever there are changes to the data, thus preventing sending and receiving repetitive data.

By using TLS, **MQTT establishes a secure, private connection** that requires the exchange and approval of security certificates where data is encrypted.

Last but not least, MQTT provides a really useful feature called “stateful awareness”. Besides transmitting process data from devices, MQTT also sends the device status data about the condition of the device or network connection, making it very easy for the operator to determine if networks are operational or devices are unavailable.

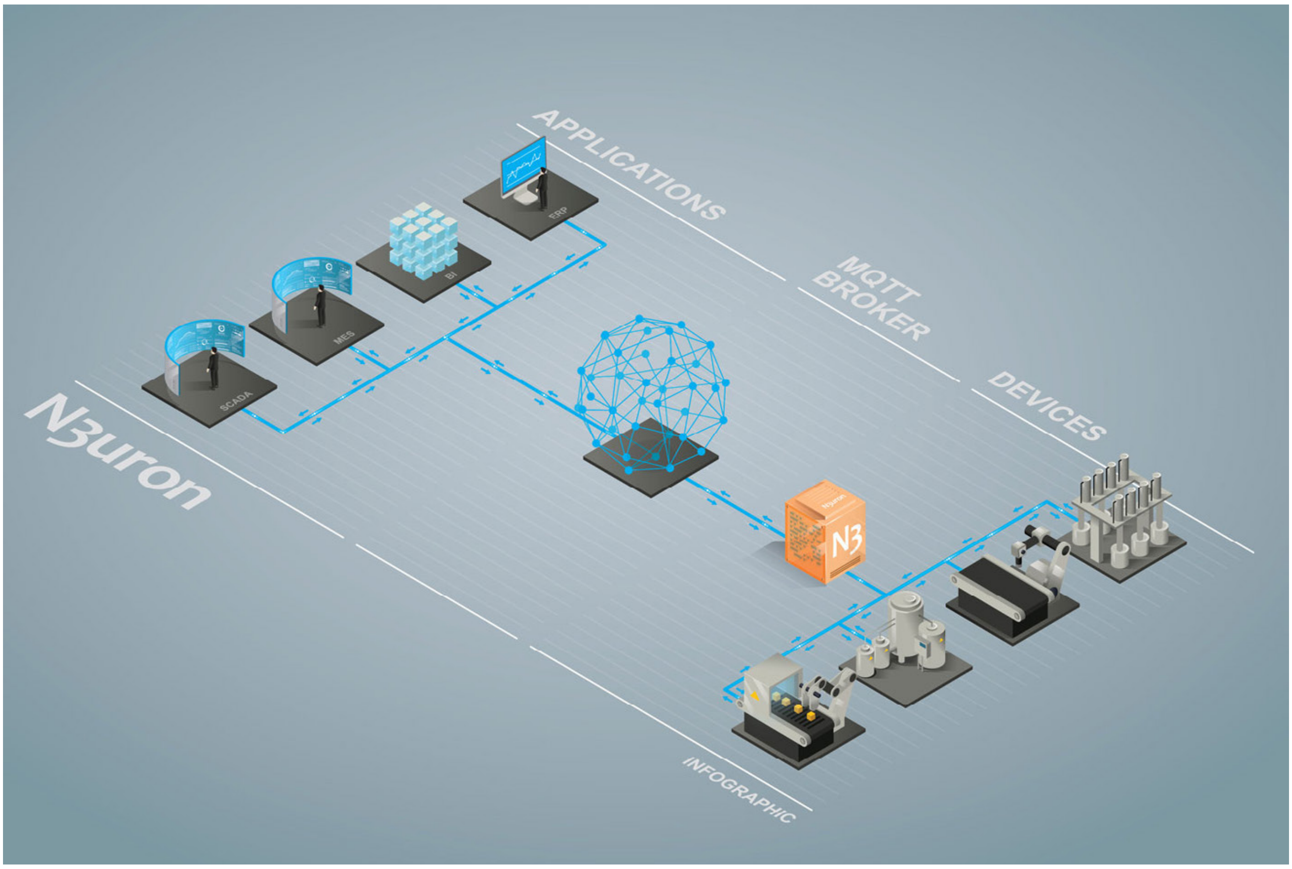
How Does MQTT Work?

MQTT is a messaging protocol based on the publish/subscribe messaging model where publishers send messages; subscribers receive the messages they are interested in, named topics, with brokers passing messages from publishers to subscribers. Both publishers and subscribers are MQTT clients communicating with an MQTT broker.

The MQTT broker is software running on a computer and acts as a post office, using subject lines in place of addresses, more commonly named topics, to which you have to subscribe if you want to receive a copy of that message. Multiple clients can receive the message from a single broker and vice versa. As it is a bidirectional communication protocol, clients can either produce or receive data.

MQTT uses reporting by exception, that is to say, data is only sent when there are changes, it is totally data-agnostic and it is possible in any encoding, encrypted data and virtually every data in binary. It also provides quality of service(Qos)levels to determine how each message is delivered.

The efficiency of the **lightweight MQTT protocol** make it possible to significantly increase the volume of data being monitored, controlled and analyzed.



Distributed architecture made up of multiple devices and industrial applications connecting to an MQTT broker

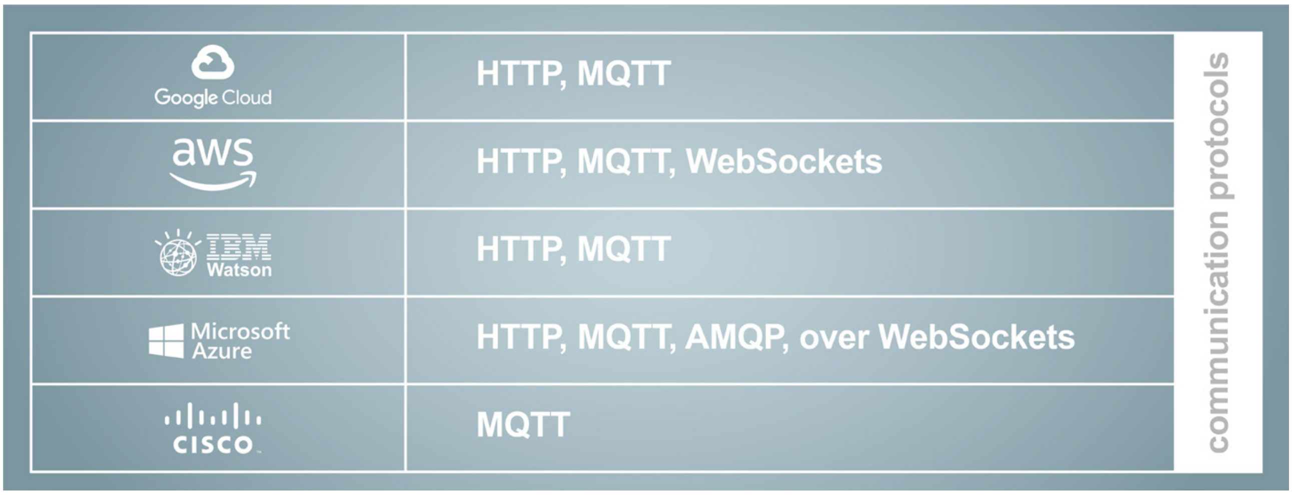
What are the advantages of using MQTT with a Cloud's provider?

As mentioned above, **using MQTT enables to decouple applications from devices**, connecting devices to infrastructure and not to applications, making data available to multiple consumers, not only for operations but any new application that may need it, such as predictive maintenance, advanced analytics and machine learning cloud-based services.

The leading cloud providers enable OT data to be collected via different protocols, but they all have one common feature. Can guess what that is? That's it, MQTT. The main reasons why MQTT has been so widely adopted for cloud data ingestion are:

- It provides true scalability
- Less overall maintenance
- It is easy to configure and deploy (plug and play)
- It provides standardization
- Secure, remote originated connections
- Low power consumption, ideal for battery-powered IoT devices
- Single source of truth
- Seamless migration between providers

So, if you are planning to deploy an IIoT architecture, play it safe and don't hesitate to choose MQTT as your cornerstone.



	communication protocols
Google Cloud	HTTP, MQTT
aws	HTTP, MQTT, WebSockets
IBM Watson	HTTP, MQTT
Microsoft Azure	HTTP, MQTT, AMQP, over WebSockets
cisco	MQTT

Table representing the leading Cloud providers supporting the MQTT communication protocol

How to Get Started with MQTT

If you're ready to try using MQTT, download the N3uron free trial version and read our MQTT manual on how to implement and use N3uron's MQTT software module on our communication platform.
Download the MQTT Manual