

# Datenübertragung

## Inhaltsverzeichnis

1	Kommunikation	2
2	MQTT - Message Queuing Telemetry Transport Protocol	3
3	Cloud, Edge und Fog	4
	Referenzen	4

Drahtlose Kommunikationstechnologien basieren heutzutage auf IEEE 802.15.4 sowie IEEE 802.15.4e Übertragungsprotokoll ([Celebi et al., 2020](#)). Das Protokoll beschreibt den die untersten beiden Schichten des OSI Modells (Abb. 1), die Bitübertragung und den MAC-Layer für Wireless Personal Area Networks (WPAN).

Einige der relevanten Technologien für die Datenübertragung sind, Wi-Fi, Bluetooth, ZigBee, LoRa, RFID, SigFox und Mobilfunk. Die Technologien unterscheiden sich in der Reichweite, der Datenrate, der Energieeffizienz und der Anzahl der Geräte, die angeschlossen werden können. Die Wahl der Technologie hängt von der Anwendung ab. Für die Übertragung von Sensordaten über kurze Distanzen eignet sich Bluetooth, ZigBee oder WiFi. Für die Übertragung von Sensordaten über lange Distanzen eignet sich LoRa oder Mobilfunk. Für die Übertragung von Sensordaten über kurze Distanzen mit hoher Datenrate eignet sich Wi-Fi. Für die Übertragung von Sensordaten über kurze Distanzen mit geringer Datenrate eignet sich RFID.

Je nach Anwendungsfall (Use Case) sind die Werte für die Latenzzeit (Latency), Verlässlichkeit (Reliability), Größe der Datenmenge (Data size), Reichweite (Range) für die Wahl der Kommunikationstechnologie relevant.

ISO-OSI 7 Schichtenmodell	Protokolle und Dienste (Auswahl)
Layer 7: Anwendung / Application	<b>MQTT</b> , HTTP, HTTPS, FTPS, FTP, SMTP, XMPP, LDAP, NCP...
Layer 6: Darstellung / Presentation	
Layer 5: Sitzung / Session	TLS, SSL
Layer 4: Transport	<b>TCP</b> , UDP, SCTP, SPX,
Layer 3: Vermittlung / Network	<b>IP</b> , ICMP, IGMP, Ipsec, IPX
Layer 2: Sicherung / Data Link	
Layer 1: Physikalische Schicht / Physical	Ethernet, Token Ring, Wireless LAN

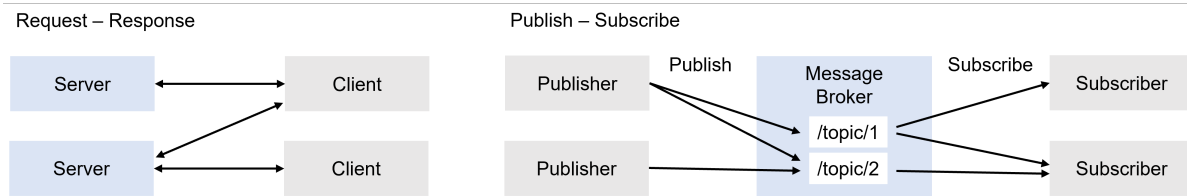
**Abb. 1:** Das ISO OSI Schichtenmodell mit einer Auswahl von Protokollen für die jeweilige Schicht. Das MQTT Protokoll ist in der Schichten der Darstellungs- und Anwendungsschicht angesiedelt und nutzt TCP/IP für die Datenübertragung über die IP Adressen der IoT Geräte.

**ISO OSI Schichtenmodell** Das ISO OSI Schichtenmodell (Open Systems Interconnection Model) ist ein konzeptuelles Referenzmodell welches die Funktionsweise von Netzwerksystemen beschreibt mit dem Ziel Kommunikation über die unterschiedlichen Technologien hinweg zu beschreiben von der Übertragung von einzelnen Bits über den Datentransport bis hin zur Anwendung beispielsweise HTTPS oder FTP (ISO, 1994). Das ISO/OSI Schichtenmodell besteht aus sieben hierarchisch aufgebauten Schichten und jeder einzelnen Schicht ist eine Aufgabe zugeordnet, wobei das Übertragungsmedium nicht definiert ist. Die Schichten eins bis vier sind transportorientierte Schichten und die Schichten fünf bis sieben sind anwendungsorientierte Schichten.

**Long Range Wide Area Network (LoRaWAN) oder LoRa** LoRaWan, Long Range Wide Area Network ermöglicht ein energieeffizientes übertragen von kleinen Datenmengen über grosse Distanzen (bis 10km).

## 1 Kommunikation

Zwei Kommunikationsparadigmen kommen in IoT Systemen zum Einsatz, *Publish-Subscribe* und *Request-Response*. Je nach Szenario und Anwendungsfall kommt der eine oder andere Ansatz zum Einsatz. In der *Publish-Subscribe* Kommunikation sind zwei Entitäten involviert, der Publisher, der die Daten publiziert und der Subscriber, der die Daten konsumiert, eine *one-to-many Kommunikation*. Dies ist gerade in IoT Systemen von Vorteil, wo mehrere Geräte die Daten von einem Gerät konsumieren können ohne, dass zu jedem einzelnen Gerät eine Verbindung aufgebaut werden muss. Eine one-to-one Kommunikation ist hingegen eine *Request-Response* Kommunikation, in der Daten zwischen zwei Entitäten ausgetauscht werden, wobei hier der Empfänger (Adresse) der Nachricht oder der Daten bekannt sein muss. In der *Request-Response* Kommunikation ist der Server zentraler Bestandteil der Kommunikation, wohingegen in der *Publish-Subscribe* Kommunikation der Broker zentraler Bestandteil der Kommunikation ist (Hirmer, 2023). Bei *Publish-Subscribe* muss der Broker bekannt sein wohingegen die Identität der Publisher und Subscriber nicht erforderlich ist.



**Abb. 2:** Kommunikationsparadigmen in der IoT Request-Response versus Publish-Subscribe

Die zentrale Komponente beim Publish-Subscribe Modells ist der Message Broker, der für das Empfangen, Zwischenspeichern und die Vermittlung der Nachrichten zu den Subscriber verantwortlich ist. Die Nachrichten oder Datenpakete werden Topics (Themen) zugeordnet die hierarchisch strukturiert sind. So können Subscriber gewisse Topics oder Untertopics subscriben, wie beispielsweise den Temperaturdaten einer Wetterstation mit *stationA/temperature*.

Message Broker können *Quality of Service* parameter definieren, die die Zuverlässigkeit der Nachrichtenübertragung definieren, ob beispielsweise die Nachricht genau oder mindestens einmal zugestellt werden soll. Dies geht jedoch zu Lasten der Performance, was gerade bei Echtzeitkommunikation relevant ist. Einer der wesentlichen Vorteile des Publish-Subscribe Modells ist, dass Publisher und Subscriber nicht gleichzeitig online sein müssen, da der Broker die Nachrichten zwischenspeichert und diese bei der nächsten Verbindung zustellt. Dies ist ermöglicht eine asynchrone Kommunikation in Echtzeit, eine wichtige Anforderung von IoT. Dies ist vorallem bei batteriebetriebenen Geräten sinnvoll, die energieeffizient arbeiten und folglich nicht kontinuierlich online sind. Eines der verbreitesten Protokolle, die *Publish-Subscribe* umsetzen ist MQTT. In **GeoMQTT** kann eine Nachricht mit einem Zeitstempel oder Interval und einer Geometrie zusätzlich dem Topic name hinzugefügt werden. Dies ermöglicht dem Subscriber zeitliche oder räumliche Filter zusätzlich zu den thematischen *Topic* Filter zu nutzen Herlé et al. (2019). Ein umfangreicherer offener Standard ist AMQP, seit 2010, der auch *request-response* Kommunikation ermöglicht<sup>1</sup>.

Eine weitere wichtige Anforderung an IoT ist die asynchrone Kommunikation in Echtzeit. Synchrone Kommunikation in Echtzeit bedingt, dass die Uhren synchron sind und beide zur gleichen Zeit kommunizieren, was in vielen Anwendungsfällen der IoT nicht wünschenswert ist.

## 2 MQTT - Message Queuing Telemetry Transport Protocol

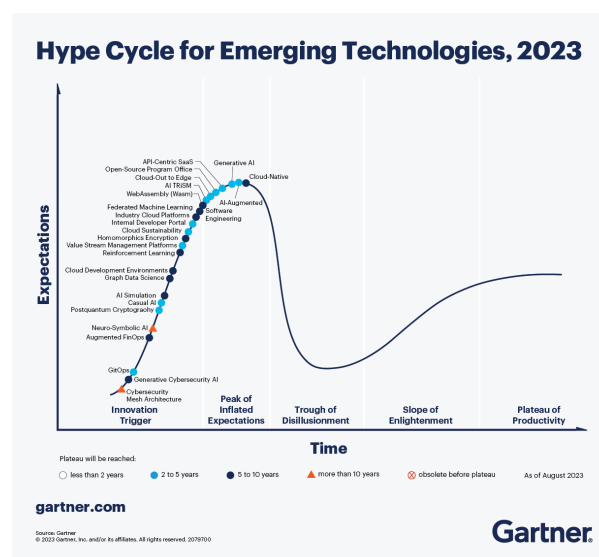
Das MQTT Protokoll mit dem *Publisher-Subscriber* Ansatz ermöglicht asynchrone Kommunikation von Events in Echtzeit in dem zwischen Subscriber und Publisher ein *Broker-Server* in der Kommunikation dazwischen steht, der die Nachrichten zwischenspeichert. MQTT ist ein Protokoll für die Kommunikation zwischen Geräten und Servern und wurde ursprünglich für eine schlanke Datenübertragung über Satellitenkommunikation entwickelt. MQTT ist ein offenes Protokoll, das seit 1999 entwickelt wird, auf TCP/IP basiert und ab der Version 3.1 geöffnet wurde.

<sup>1</sup> Diese Protokolle sind nach dem Entwurfsmuster (Design Pattern) Beobachter (Observer) implementiert

Der Publisher sendet *publish* eine Nachricht zu einem Topic (beispielsweise *gebäude1/labor1/temperature*) mit einem bestimmten *Quality of Service* (at most once, at least once, exactly once) Parameter. Der Broker speichert die Nachricht und sendet diese an alle Subscriber, die dieses Topic abonnieren *subscribe*. **Mosquitto** ist ein quelloffener Message Broker, der die MQTT implementiert. Mosquitto ist schlank und eignet sich für den Einsatz auf allen Geräten, von stromsparenden Einplatinencomputern bis hin zu kompletten Servern. Eclipse Paho ist eine quelloffene Implementierung von MQTT und bietet Bibliotheken in verschiedenen Programmiersprachen wie Python, C++ oder Java an.

### 3 Cloud, Edge und Fog

Mit zunehmender Rechenleistung auf IoT Geräten können diese vermehrt selbst Daten prozessieren, was eine Verlagerung der Analyse ermöglicht. Die Begriffe Cloud, Edge und Fog bezeichnen im wesentlichen wo in Infrastrukturen die Datenprozessierung durchgeführt wird. In **Cloud** Infrastrukturen erfolgt die Prozessierung zentralisiert in der Cloud und grossen Recheninfrastrukturen, wohingegen **Edge**-Computing eine Infrastruktur bezeichnet, in der die lokalen Geräte selbst einen Teil der Daten dezentral und möglichst lokal verarbeiten. Fog - ein von Cisco eingeführter Begriff - bezeichnet Cloud Computing im lokalen Netzwerk, beispielsweise könnte ein Transportunternehmen die Verwaltung der Datenprozessierung in der eigenen Cloud durchführen.



**Abb. 3:** Im Gartner Hype Cycle for Emerging Technologies 2023 erreicht der “Hype” *Cloud-Out to Edge* den Peak of Inflated Expectations (Gartner, 2023). Das Hype Cycle Thema “Cloud-Out to Edge” wurde im aktualisierten Hype Cycle (2024) als Hype Thema entfernt.

### Referenzen

Celebi, H. B., Pitarokoilis, A., & Skoglund, M. (2020). Wireless Communication for the Industrial IoT. In I. Butun (Hrsg.), *Industrial IoT : Challenges, Design Principles, Applications, and*

*Security* (S. 57–94). Springer International Publishing. [https://doi.org/10.1007/978-3-030-42500-5\\_2](https://doi.org/10.1007/978-3-030-42500-5_2)

Gartner. (2023). 4 Exciting New Trends in the Gartner Emerging Technologies Hype Cycle. In *Gartner*. <https://www.gartner.com/en/articles/what-s-new-in-the-2023-gartner-hype-cycle-for-emerging-technologies>.

Gartner. (2024). Gartner 2024 Hype Cycle for Emerging Technologies Highlights Developer Productivity, Total Experience, AI and Security. In *Gartner*. <https://www.gartner.com/en/newsroom/press-releases/2024-08-21-gartner-2024-hype-cycle-for-emerging-technologies-highlights-developer-productivity-total-experience-ai-and-security>.

Herlé, S., Bill, R., & Blankenbach, J. M. (2019). *A GeoEvent-driven architecture based on GeoMQTT for the Geospatial IoT* (phdthesis RWTH-2019-10695). Dissertation, Rheinisch-Westfälische Technische Hochschule Aachen, 2019.

Hirmer, P. (2023). Foundations. In P. Hirmer (Hrsg.), *Model-Based Approaches to the Internet of Things* (S. 7–15). Springer International Publishing. [https://doi.org/10.1007/978-3-031-18884-8\\_2](https://doi.org/10.1007/978-3-031-18884-8_2)

ISO. (1994). ISO/IEC 7498-1:1994. In *ISO*. <https://www.iso.org/standard/20269.html>.

Laska, M., Herle, S., Klamma, R., & Blankenbach, J. (2018). A Scalable Architecture for Real-Time Stream Processing of Spatiotemporal IoT Stream Data—Performance Analysis on the Example of Map Matching. *ISPRS International Journal of Geo-Information*, 7(7), 238. <https://doi.org/10.3390/ijgi7070238>