Connectivity, protocols, security and IoT needs: a compass to find a path

In Internet of Things, one size does not fit all: to answer multiple needs and use case constraints – data volume, latency, battery consumption, interoperability, security and much more – technologies have been developed in consequence. The transport of data between devices and platforms involves a wide techno-diversity: radio connectivities, protocols and security mechanisms.

Compiling work done by Orange experts in different European countries, this article provides explanations and factual metrics on technologies provided or used by Orange and, based on them, a few arguments to help to find a path in the forest of choices.

Considering some big trees in the techno-diversity

Transporting a message means at least the use of: a connectivity, a protocol and a potential security mechanism. In this article, we consider some widely spread technologies for each topic:

- Connectivity: 3GPP standardized LPWA adaptations of 4G, **LTE-M** and **NB-IoT**, and also **LoRaWAN®**
- Protocols: **SMS**, a basic but still very popular protocol, **MQTT** which became the most popular IoT protocol in the past few years, **CoAP** and **LightweightM2M (LwM2M)**, more recent standards that aim to reduce protocol overhead and battery consumption
- Security mechanisms: **TLS** and **DTLS** using Pre-Shared Key (**PSK**) client authentication and **X.509** public key certificate standards, and **OSCore**, especially in CoAP/LwM2M context

Device and data management	LwM2M	
Application	HTTP, MQTT, CoAP	
Security	TLS, DTLS, OSCore	LoRaWAN® (main approach)
Transport	TCP, UDP	
	IP	
Connectivity	LTE-M, NB-IoT	

Note that most of the mentioned technologies are still evolving. The facts shown in this article are subject to versions and implementations of different technologies. For the same reasons, performances shown are not be taken as an Orange commitment.

Focus #1: Standards and interoperability

SMS is a de facto standard. MQTT initially de facto standard is from 2014 standardized at OASIS. CoAP standardization work has been done mainly by IETF. LwM2M is a standard from OMA.

Using standard protocols does not guarantee capacity to "plug and play" a device on any IoT platform supporting these protocols. A higher layer protocol is needed, for instance to encode and decode the telemetry data, for device management operations such as remote configuration or firmware update.
These higher layer protocols are very commonly proprietary, requiring specific integration on the device or platform side. This can increase the cost and delay of an IoT project and limits the capacity to switch between IoT platform providers. So it is a key aspect to consider when choosing a protocol stack.

Transmission method	Collect telemetry Data	Remote action	Remote configuration / firmware update
CoAP	Proprietary (but interoperability can be achieved in simple cases**)	Proprietary	Proprietary
LwM2M	Interoperable*	Interoperable*	Interoperable*
MQTT	Proprietary (but interoperability can be achieved in simple cases**)	Proprietary (but interoperability can be achieved in simple cases**)	Proprietary
SMS	Proprietary (but interoperability can be achieved in simple cases**)	Proprietary (but interoperability can be achieved in simple cases**)	Proprietary

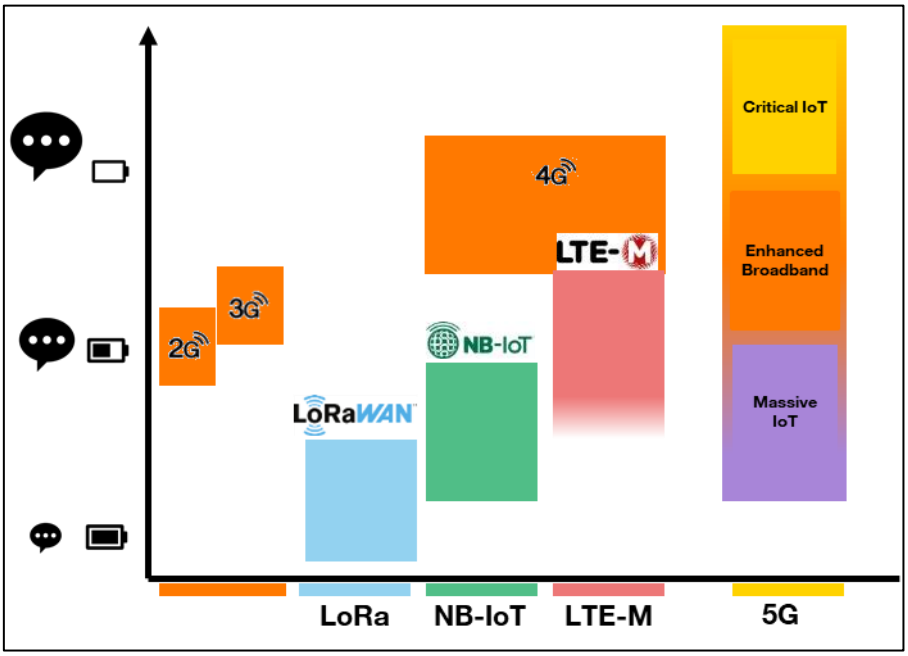
* Proprietary data representation can be used on LwM2M to address some lacks in the standard. The standard is relatively young, we are not sure that the compliance certification system is robust.
** Some IoT platforms propose configurable payload decoding service. This is enough for quick and cost-effective integration of a new device even if payload format is proprietary.

Connectivities have been designed with various capacities

Let us consider some orientations of connectivities. These have been designed to provide a certain range of bandwidth and extended coverage and, linked to that, a level of capacity concerning energy consumption:

- At the low end, LoRaWAN® allows only very short messages in limited number according to restrictions (Duty Cycle) on unlicensed spectrum, and high energy consumption optimization

- NB-IoT, with an uplink limited to 14 Kbytes per second, is also dedicated to low bandwidth communication
- LTE-M covers a wider range of bandwidth and capacities (SMS, IP, voice) and provides nomadism
- Both 4G adaptations provide advanced features, which can be activated on device and network level, like Extended idle mode Discontinuous Reception (eDRX) and Power Save Mode (PSM) dedicated to save energy
- Note that 5G, designed to target multi-services, will cover a wide range of bandwidths and energy consumption capacities, depending on the 5G "application" (i.e., behavior) used: Critical IoT, Enhanced Broadband, Massive IoT



The picture above shows the purpose of different connectivity technologies. The calculations and tests below provide some facts and proofs of their capacities.

Compatibility and performance depend on configuration and radio conditions

Considering each technology in isolation with a single behavior would be pleasant but unfortunately is too simplistic:

1. In order to allow optimization, connectivities and protocols can be configured and/or used in different ways, with consequences on performance and also sometimes on compatibility (for instance the capacity to use TCP-based protocols on NB-IoT).
2. Radio conditions can also impact technologies behavior, causing changes of retransmission and latency. This should lead to the consideration of robustness and resilience of technologies used.

Compatibility and performance explained further need to be considered as network dependent. Tests shown below have been performed on different Orange networks.

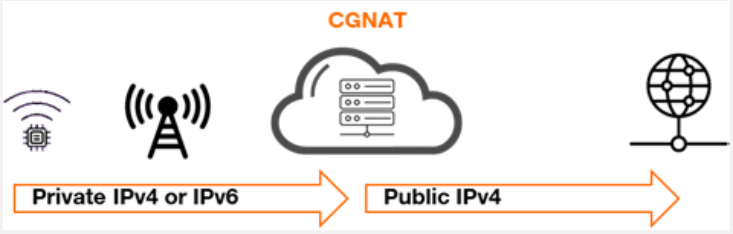
Connectivity	Supported protocols	Supported security
LoRaWAN®	(mainly) non IP	AES 128 (CTR and CMAC modes)
NB-IoT	IP and non IP (*): MQTT, CoAP, LwM2M	TLS/DTLS
LTE-M	SMS (*) IP and non IP (*)(**): MQTT, CoAP, LwM2M Optional voice (**)	TLS/DTLS
2-4G/GSM	SMS IP and non IP: MQTT, CoAP, LwM2M Optional voice	TLS/DTLS

(*) Can be operator implementation dependent.
(**) Still in discussion within ecosystem. To be confirmed.

- LoRaWAN®, in operational deployment, is today mainly non IP. Work to run IP over LoRaWAN® has been done for several years, but with very limited use. Authentication and data encryption is based on AES 128
- NB-IoT: communications done so far on the Orange Belgium network show that:
  - NB-IoT supports MQTT, CoAP and LwM2M
  - Capacity to customize protocol layers timeouts, acknowledges and retransmissions accurately is key to cope with NB-IoT latencies: this can be done with CoAP and LwM2M (which are UDP-based), but not with MQTT (done at TCP level, with often no way to configure)
  - Implementation of TLS/DTLS has drastic impact on energy consumption and bandwidth. In particular, the use of DTLS Resumption reduces significantly the overhead of resuming the secure link, and therefore the overhead linked to security
  - **Conclusion: NB-IoT supports MQTT, CoAP and LwM2M, and DTLS security mechanisms, but optimizations are more easily done with CoAP and LwM2M**
- LTE-M: a lot of work has been done with partners since the launch of LTE-M by Orange in France, especially with **SMS and MQTT(S), confirming that they are well supported**

Focus #2: Impact of Carrier-Grade NAT (CGNAT) on IoT

Since the Internet is running out of public IPv4 addresses, most telco operators use Carrier-Grade NAT (CGNAT) to use a small pool of public addresses for a large number of IoT devices: IoT devices are configured with private network addresses, which are translated by CGNAT during sessions into public IPv4 addresses to reach the public Internet.



In order to free up operator resources, inactive sessions are closed. This timeout is operator and APN dependent (private APN can offer much longer timeout). Timeout management and duration differ also for TCP and UDP:

- TCP offers additional information to maintain the active session
- UDP does not provide such information. So, timeouts are much shorter

CGNAT has two major impacts in the context of IOT:

- Reestablishing sessions is costly in terms of messages exchanged and thus also of battery consumption for the device
- Devices cannot be spontaneously contacted when sessions are closed

For UDP-based protocols, some mechanisms can be put in place to cope with CGNAT timeout:

- Use of periodic keep-alive messages to keep the session open. However, this results in higher battery consumption
- Non-IP wake-up message (e.g., SMS) to ask the device to reopen a session but with complexity (multi-protocols) and consequences on battery consumption

LwM2M offers natively:

- Queue Mode (introduced in v1.0) that allows queuing incoming commands when the session is closed. If resulting latency on downlink is acceptable, it proposes a convenient way to manage at protocol level the CGNAT traversal issue
- TCP binding (introduced in V1.1) allows LwM2M to be sent over CoAP/TCP (instead of default UDP)
- NIDD (introduced in V1.1) promises to improve battery lifetime

Transport efficiency: a first glance before battery consumption

Protocols and security mechanisms define:

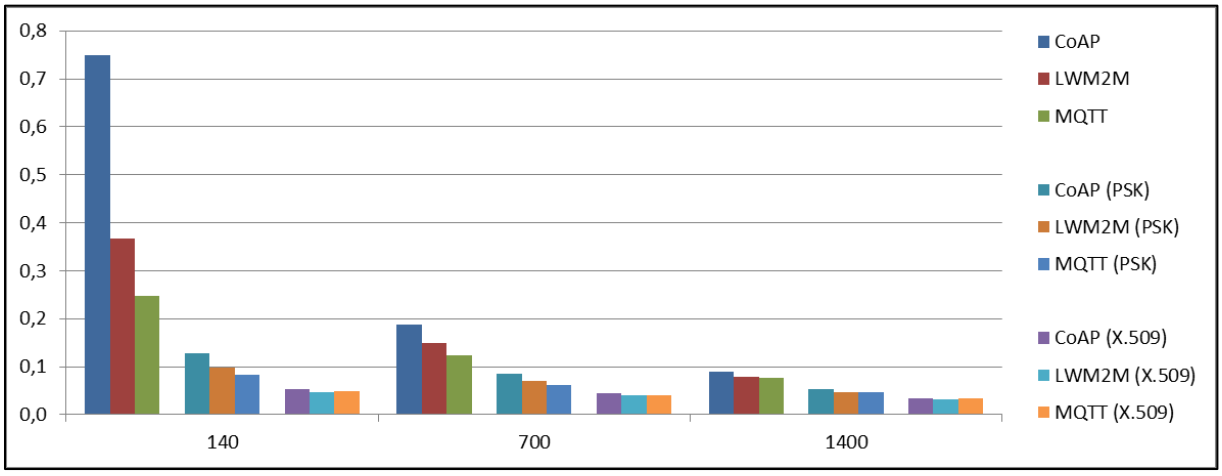
- Ways to format data, with an overhead of bytes and then an efficiency coefficient (total transmitted bytes / payload size)
- The dialog between the sender and the receiver

The energy spent for the transmission, and potential battery consumption at device level, depends then not only on **transport efficiency**, but also on this **dialog**, and its operational realization of each connectivity.

Even if transport efficiency is only a part of the battery consumption subject, the following grid shows that it is highly impacted by protocols (packets and session management) and security (especially handshake) [tests done at Orange in 2020, using CoAP RFC7252, LwM2M 1.1 and DTLS 1.2 – MQTT OASIS v5 and TLS 1.2]:

Transmission method	Payload size	Total transmitted bytes (for security handshake)	Efficiency
CoAP	140	187	0.749
LwM2M	140	435	0.322
MQTT	140	650	0.215
CoAP (PSK)	140	1093 (957)	0.128
LwM2M (PSK)	140	1560 (903)	0.090
MQTTs (PSK)	140	1824 (1029)	0.077
CoAP (X.509)	140	2642 (2494)	0.053
LwM2M (X.509)	140	3097 (2440)	0.045
MQTT (X.509)	140	2984 (2189)	0.047

And for sure, if efficiency differs a lot for small payloads as shown above, **the difference is less significant when security is applied and when the volume of transmitted data (140, 700, 1400 bytes) increases**:



Battery consumption measurements

Then what are the facts? Here are some measurements done at Orange.

SMS on 2G:

- Measurements were done several years ago at Orange to compare SMS and IP/FTP battery consumption. From that time, module efficiency has changed, but the main lesson is still valid: for a message size of 100 bytes, battery consumption is lower (~15%) in SMS than by TCP/FTP

LoRaWAN®:

- Measurement (*) were done with a water meter device, running with different data rates and repetitions, sending one payload of 48 bytes
  - ⇨ Daily consumption varies between 0.002 and 0.25 mAh

	LoRa SF 12 without retransmission	LoRa SF 12 3 transmissions	LoRa SF7 3 transmissions
Daily metering consumption	1,116 mWh		
Daily standby consumption	0,144 mWh		
Daily communication consumption (one 48 bytes payload per day)	0,288 mWh	0,9 mWh	0,0072mWh
3400 mAh battery life	17 years	13 years	20 years

(*) Measurements were done in 2016 at Orange on LoRaWAN®. Since then, Semtech chips have evolved to increase battery life, but these figures are worth considering as an order of magnitude for LoRaWAN®.

LTE-M:
Tests done in 2020 on the Orange France 800 MHz LTE-M network, in good radio conditions (-95 < RSRP < - 90dBm), show the following results to send a GPS location (a few dozen bytes) [MQTT 3.1.1]:

	PSM deactivated	PSM activated
Wake up	0.848 mWh	0.444 mWh
SMS sending (LTE-M based)	0.671 mWh	0.659 mWh
Pre-sleep / Idle PRX Mode	0.068 mWh	0.213 mWh
Total	**1.587 mWh**	**1.316 mWh**

	PSM deactivated	PSM activated
Wake up	0.848 mWh	0.874 mWh
MQTTS transmission	2.274 mWh	2.700 mWh
Pre-sleep / Idle PRX Mode	0.068 mWh	0.068 mWh
Total	**3.190 mWh**	**3.642 mWh**

These results show:
- SMS (resp. MQTTS) brings a battery consumption around 5x (resp. 10x) than LoRa SF12 (2016 tests)
- Consumption for MQTTS is due to transmitted data overhead (MQTT and security)

Remarks:
- Wake up phase may vary with protocol used and can be device / module dependent (here SMS wake up phase is improved with PSM)
- PSM should not impact MQTTS transmission. Difference above may be explained by radio and / or server conditions

NB-IoT:
Tests done in Spain in 2020 on 2G/GPRS and NB-IoT v1 at 800 MHz networks show the following results [CoAP RFC 7252 and MQTT mosquito without TLS]:

	Average Total Consumption (mWh)	CoAP	MQTT
GPRS	Open context + send 16 bytes + close communications	0.1825	0.53
	Open context + send 512 bytes + close communications	0.2112	0.686
NB-IoT	Open context + send 16 bytes + close communications	0.1665	0.49675
	Open context + send 512 bytes + close communications	0.1958	0.61475

These results show:

- COAP over NB-IoT or GPRS brings a battery consumption at the same level of magnitude as LoRaWAN® SF12 (2016 tests)
- NB-IoT brings little advantage in terms of battery consumption for data transmission, compared to 2G/GPRS
- On NB-IoT there is a good correlation (even if not proportional) between transport efficiency and battery consumption: CoAP is more efficient than MQTT, especially for small packets

Conclusions: some help to find a path

Here are some keys features regarding each technology and some recommendations (➔) to help manage them.

Connectivities:

- **LoRaWAN®, LTE-M and NB-IoT**: the tests done confirm the positioning of connectivities on battery consumption: LoRaWAN® first, then NB-IoT and then LTE-M. Factors of gain depend deeply on radio conditions and protocols
  - ➔ While considering additionally maturity and simplicity, LoRaWAN® is a good solution for Low Power IoT.
- **LTE-M and NB-IoT**: a basic use of LTE-M and NB-IoT is possible, but will only bring advantage in term of range. Battery optimization will need advanced feature use (eDRX, PSM) that are complex to handle and that may vary from one network / geography to another: this will require expertise and important tests, in laboratory and on field
  - ➔ LTE-M and NB-IoT complexity should lead to prefer experimented devices partner / providers, who not only master protocols, but also test with experts and in different geographies / conditions.

With protocols:

- **SMS** is a de facto standard. **MQTT**, **CoAP** and **LwM2M** are standards from different organizations. Standardization is not a way to decide between these protocols, and standard (strict) use does not guarantee interoperability
  - ➔ LwM2M provides this interoperability, with standardization in data and device model.
- **SMS**: measurements done on 2G and on LTE-M show that SMS has a good energy efficiency for short payloads. SMS is basic, simple to use, with retries, which bring

robustness when the device is temporarily unreachable, but with no guaranty in terms of transport delay

➔ If potential latency is not an issue, SMS can be preferred for short messages, with an honest answer in terms of energy efficiency, taking benefits of network evolution (2G, 3G, LTE-M, NB-IoT).

- **MQTT**: MQTT is widely used in IOT and well known by developers. For messages bigger than 1 Kbyte, overhead in transport (compared to CoAP and LwM2M) is low, bringing no disadvantage in this situation. The use of TCP is also more resilient in case of NAT traversal, firewall.
  ➔ If simplicity outweighs optimization, MQTT is a good choice of protocol.
- **CoAP and LwM2M**: CoAP and LwM2M offer configuration methods, which allow more resiliency against variability of NB-IoT and LTE-M behavior. On NB-IoT, battery tests show the good correlation – in good radio condition – with transport efficiency, which is better for CoAP and LwM2M
  ➔ CoAP and LwM2M have measured battery advantages on NB-IoT, though DTLS largely reduces transport efficiency. In addition, these protocols can be preferred to MQTT for resilience, while knowing that resilience will be accessed only by configuration expertise.
- **LwM2M**: LwM2M natively provides an optional mechanism to wait for reachability of devices (Queuing Mode)
  ➔ If simplicity is targeted, and if increased latency on uplink can be accepted, the use of lwM2M Queuing Mode is a good option.

And security:

- **Security mechanisms should be well chosen** in relation to the targeted security level (X.509 > PSK > no security), battery consumption efficiency (no security > PSK > X.509) and implementation complexity (X.509 (PKI management), PSK (exchange of keys)
- **Handshake** is a major part, in terms of dialog and data exchange, of PSK or X.509 execution
  ➔ Depending on security sensibility, this can be reduced:
    - No systematic handshake: keeping the security session open for several exchanges (or a “long” period of time)
    - Shorter handshake: just resuming the session, with new session keys while keeping master keys already shared (here also for a certain period of exchange)
- **OSCore** is a way to provide end-to-end security, while not giving keys to intermediate infrastructure elements. It can be used with TLS or DTLS mechanisms or alone, thus simplifying exchanges and saving battery. But OSCore is a new mechanism, which still needs to be handled as such

For more information, contact one of our local sales representatives or visit orange-business.com.