How Do You Select Which IoT Protocol to Use?

CHRISTIAN LEGARE | DIRECTOR OF IOT OS PLATFORM | FEB 2018
## IoT Systems – Consumer vs Industrial

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Industrial IoT</th>
<th>Human IoT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Market Opportunity</td>
<td>Brownfield (known environment)</td>
<td>Greenfield (unchartered domain)</td>
</tr>
<tr>
<td>Product Lifecycle</td>
<td>Until dead or obsolete</td>
<td>Whims of style and/or budget</td>
</tr>
<tr>
<td>Solution Integration</td>
<td>Heterogeneous APIs</td>
<td>Vertically integrated</td>
</tr>
<tr>
<td>Security</td>
<td>Access</td>
<td>Identity &amp; privacy</td>
</tr>
<tr>
<td>Interaction</td>
<td>Autonomous</td>
<td>Reactive</td>
</tr>
<tr>
<td>Availability</td>
<td>0.99999 to 0.99999 (4–5 ‘9’s)</td>
<td>0.99 to 0.999 (2–3 ‘9’s)</td>
</tr>
<tr>
<td>Access to Internet</td>
<td>Intermittent to independent</td>
<td>Persistent to interrupted</td>
</tr>
<tr>
<td>Response to Failure</td>
<td>Resilient, fail-in-place</td>
<td>Retry, replace</td>
</tr>
<tr>
<td>Network Topology</td>
<td>Federations of peer-to-peer</td>
<td>Constellations of peripherals</td>
</tr>
<tr>
<td>Physical Connectivity</td>
<td>Legacy &amp; purpose-built</td>
<td>Evolving broadband &amp; wireless</td>
</tr>
</tbody>
</table>

Source: Patrick Morehead, Forbes, “Who Wins In The Industrial Internet Of Things (IIoT)?”, October 29 2013
IoT Building Blocks

IoT = Identification + Sensing + Communication + Computation + Services + Semantics

Application Protocol | DDS | CoAP | AMQP | MQTT | MQTT-SN | XMPP | HTTP/REST
--- | --- | --- | --- | --- | --- | --- | ---
Service Discovery | | | mDNS | | DNS-SD |
Routing Protocol | | | | | | | RPL
Network Layer | | | 6loWPAN | | | | IPv4 / IPv6
Link Layer | | | | | | IEEE 802.15.4 |
Physical / Device Layer | LTE-A | EPCglobal | IEEE 802.15.4 | | Z-Wave |
Influential Protocols | | | | | | | IEEE 1888.3, IPSec, IEEE 1905.1

Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications

Ala Al-Fuqaha, Senior Member, IEEE
Mohsen Guizani, Fellow, IEEE
Mehdi Mohammadi, Student Member, IEEE
Mohammed Aledhari, Student Member, IEEE
Moussa Ayyash, Senior Member, IEEE
IOT DEVELOPER SURVEY RESULTS
April 2017

What messaging protocols do you use for your IoT solution?

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Usage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP</td>
<td>60.1%</td>
</tr>
<tr>
<td>MQTT</td>
<td>54.7%</td>
</tr>
<tr>
<td>CoAP</td>
<td>26.7%</td>
</tr>
<tr>
<td>In-house / proprietary</td>
<td>18.4%</td>
</tr>
<tr>
<td>HTTP/2</td>
<td>16.8%</td>
</tr>
<tr>
<td>AMQP</td>
<td>15.0%</td>
</tr>
<tr>
<td>XMPP</td>
<td>10.3%</td>
</tr>
<tr>
<td>Other</td>
<td>7.1%</td>
</tr>
<tr>
<td>Don’t know</td>
<td>7.1%</td>
</tr>
<tr>
<td>Proprietary vendor protocol (specify below)</td>
<td>4.9%</td>
</tr>
<tr>
<td>DDS</td>
<td>4.0%</td>
</tr>
<tr>
<td>None</td>
<td>3.6%</td>
</tr>
</tbody>
</table>
IP Usage

Web Services

- IP family of protocols that can be used to provide services to a device
- Examples: SMS text, e-mail, file sharing, streaming audio, speech to text, social media ...

IoT Services

- The availability of back-end services based on IP protocols
- Differentiating “IoT devices” from “connected devices”:
- Examples: Storage, multiple devices/applications data usage, system analytics and potential for efficiency gain...
# Internet Protocol Types

<table>
<thead>
<tr>
<th>Request/Response</th>
<th>Publish/Subscribe</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP</td>
<td>MQTT</td>
</tr>
<tr>
<td>WebSocket</td>
<td>CoAP</td>
</tr>
<tr>
<td>CoAP</td>
<td>XMPP</td>
</tr>
<tr>
<td>Web Services</td>
<td>IoT Services</td>
</tr>
<tr>
<td>IoT Services</td>
<td></td>
</tr>
</tbody>
</table>
Request/Response
HTTP
HTTP Request

Client opens connection and sends a request message to HTTP server

Server returns a response

Client closes the connection

HTTP is ‘stateless’
HTTP and HTTPS

- Typical HTTPS
  - showing the messages, not the number of packets
- Areas in blue are optional
  - bidirectional SSL/TLS
- Connection is initiated by a client
- Client always has to poll the server, server cannot initiate connection: not efficient for an embedded device
- High overhead: Open/Send/Close for every message
Request/Response

Websocket
WebSockets are a bi-directional, full-duplex, persistent connections from a client to a server.

Once a WebSocket connection is established the connection stays open until the client or server decides to close this connection.
With this open connection, the client or server can send a message at any given time to the other.

This makes web programming entirely event driven, not (just) user initiated.

It is stateful.
Publish/Subscribe

CoAP
CoAP Architecture

RFC 7252

Proxy/Gateway

Constrained device

The Internet

Constrained Environments

REST

CoAP Architecture

HTTP

CoAP
As of July 2016, the IETF draft defining publish/subscribe and message queuing functionality for CoAP that extends the capabilities for supporting nodes with long breaks in connectivity and/or uptime is in its 5th iteration.

https://datatracker.ietf.org/doc/draft-koster-core-coap-pubsub/
Publish/Subscribe

MQTT
MQTT

- MQTT has a client/server model, where every device is a client and connects to a server, known as a broker, over TCP.
- MQTT is message oriented. Every message is a discrete chunk of data, opaque to the broker.
- Every message is published to an address, known as a topic. Clients may subscribe to multiple topics.
- Every client subscribed to a topic receives every message published to the topic.
MQTT Quality of Service (QoS)

There are 3 QoS levels in MQTT:

- **QoS 0** at most once
  - Does not survive network failure
  - Never duplicated

- **QoS 1** at least once
  - Survives network failure
  - Can be duplicated

- **QoS 2** exactly once
  - Survives network failure
  - Never duplicated

The Quality of Service (QoS) level is an agreement between sender and receiver of a message regarding the guarantees of delivering a message.
MQTT-SN

- MQTT-SN uses UDP instead of TCP
- Client connects to Gateway via MQTT-SN
- Gateway can be integrated into broker/server or can connect via MQTT
  - The main function of the Gateway is to translate between MQTT and MQTT-SN
  - Forwader encapsulates MQTT-SN packets and forwards unchanged to the gateway
DDS is Decentralized

- Is an Object Management Group (OMG) standard
- Introduced in 2004
- Uses a Publish/Subscribe architecture
- Uses network resources efficiently
- Can be deployed without servers or brokers
- Commercial and Open Source versions

**Fast**
- 100,000’s update/sec

**Scalable**
- Load independent # apps

**Managed with QoS**

**Reliable**
- No single point of failure
Web Versus Dedicated IoT

Web

- Hundreds / thousands of bytes
- XML
- HTTP
- TLS
- TCP
- IPv6
  - Inefficient content encoding
  - Huge overhead, difficult parsing
  - Requires full Internet devices

Internet of Things

- Tens of bytes
- Web Objects
- CoAP
- DTLS
- UDP
- 6LoWPAN
  - Efficient objects
  - Efficient Web
  - Optimized IP access
### Messaging Standards Usage

What messaging protocols do you use for your IoT solution?

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>HTTP</td>
<td>60.1%</td>
</tr>
<tr>
<td>MQTT</td>
<td>54.7%</td>
</tr>
<tr>
<td>CoAP</td>
<td>26.7%</td>
</tr>
<tr>
<td>In-house / proprietary</td>
<td>18.4%</td>
</tr>
<tr>
<td>HTTP/2</td>
<td>16.8%</td>
</tr>
<tr>
<td>AMQP</td>
<td>15.0%</td>
</tr>
<tr>
<td>XMPP</td>
<td>10.3%</td>
</tr>
<tr>
<td>Other</td>
<td>7.1%</td>
</tr>
<tr>
<td>Don’t know</td>
<td>7.1%</td>
</tr>
<tr>
<td>Proprietary vendor protocol (specify below)</td>
<td>4.9%</td>
</tr>
<tr>
<td>DDS</td>
<td>4.0%</td>
</tr>
<tr>
<td>None</td>
<td>3.6%</td>
</tr>
</tbody>
</table>
So, which protocol do I use?
There are 100’s if not 1000’s of different use cases. The needs of your use case will determine the protocol you need to use.

Define your system requirements very precisely, for example:
- What does ‘real time’ mean in your use case?
- What is a ‘thing’ in your system?
- What hardware resources are available to your ‘things’?

Choose the protocol that satisfies your system requirements:
- How you have defined your system will be critical to your choice of protocol(s)
Thank you.