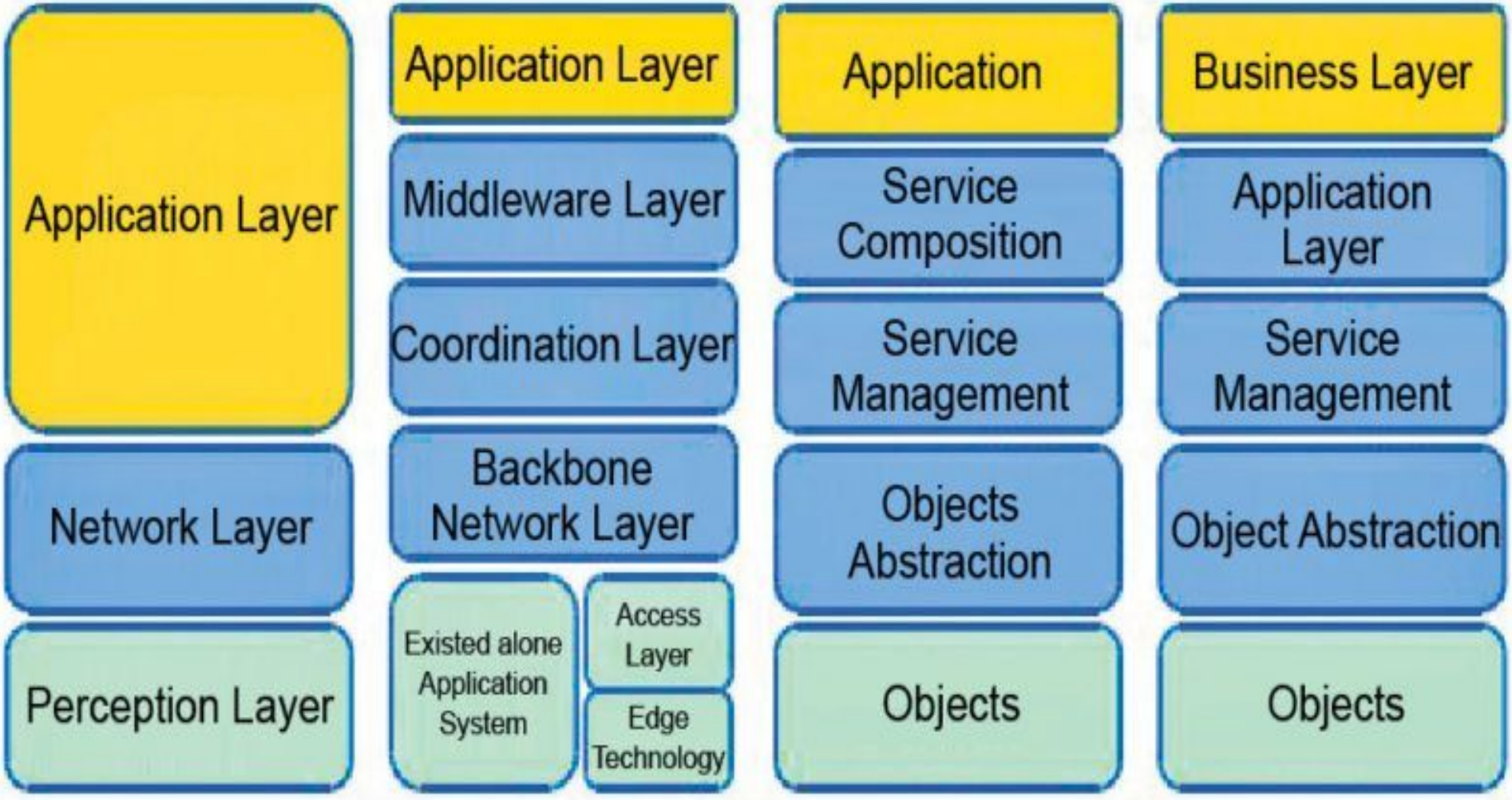
What connects the things…?

IOT Protocols


Application Layer
Network Layer
Perception Layer
Application Layer
Middleware Layer
Coordination Layer
Backbone Network Layer
Existed alone Application System
Access Layer
Edge Technology
Application
Service Composition
Service Management
Objects Abstraction
Objects
Business Layer
Application Layer
Service Management
Object Abstraction
Objects

IOT Architecture….

As the IoT is capable of connecting billions of heterogeneous objects via the Internet, there is an emerging requirement for a dynamic layered architecture.

- **Objects layer**
- **Object abstraction layer**
- **Service management layer**
- **Application layer**
- **Business layer**

IOT Protocols….

IEEE (Institute of Electrical and Electronics Engineers) and ETSI (European Telecommunications Standards Institute) have defined some of the most important IoT protocols, as listed below;

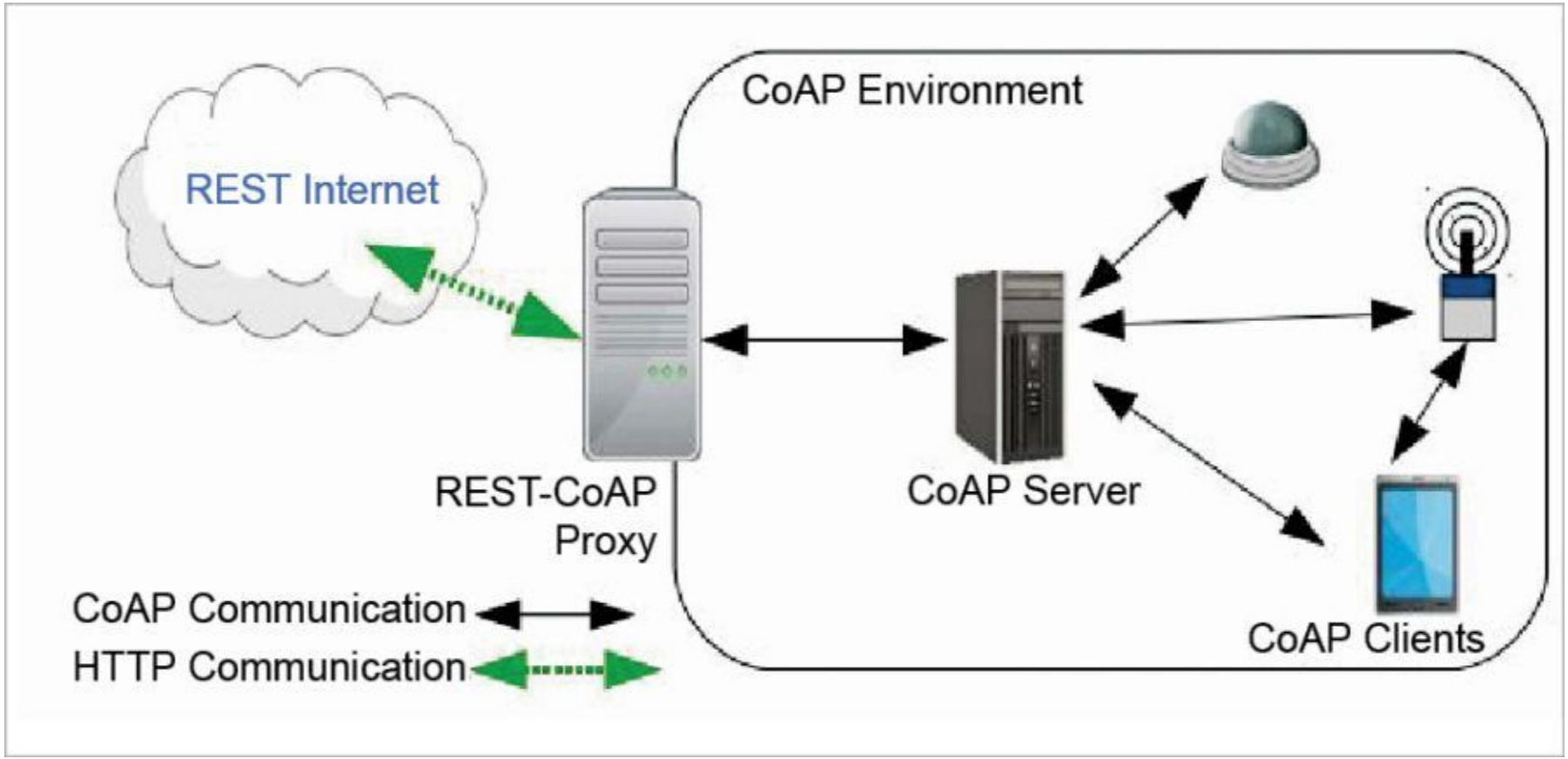
Table 1 protocols in different layers

Application layer	HTTP, CoAP, EBHTTP, LTP, SNMP, IPfix, DNS, NTP, SSH, DLMS, COSEM, DNP, MODBUS
Network/Communication layer	IPv6/IPv4, RPL, TCP/UDP, uIP, SLIP, 6LoWPAN,
PHY/MAC layer	IEEE 802.11 Series, 802.15 Series, 802.3, 802.16, WirelessHART, Z-WAVE, UWB, IrDA, PLC, LonWorks, KNX

Constrained Application Protocol (CoAP)

This was created by the IETF Constrained RESTful Environments (CoRE) working group. CoAP is an Internet application protocol for constrained devices. It is designed to be used between devices on the same constrained network, between devices and general nodes on the Internet, and between devices on different constrained networks—both joined on the Internet. This protocol is especially designed for IoT systems based on HTTP protocols.

CoAP makes use of the UDP protocol for lightweight implementation. It also makes use of RESTful architecture, which is very similar to the HTTP protocol. It is used within mobiles and social network based applications and eliminates ambiguity by using the HTTP *get, post, put* and *delete* methods. Apart from communicating IoT data, CoAP has been developed along with DTLS for the secure exchange of messages. It uses DTLS for the secure transfer of data in the transport layer.

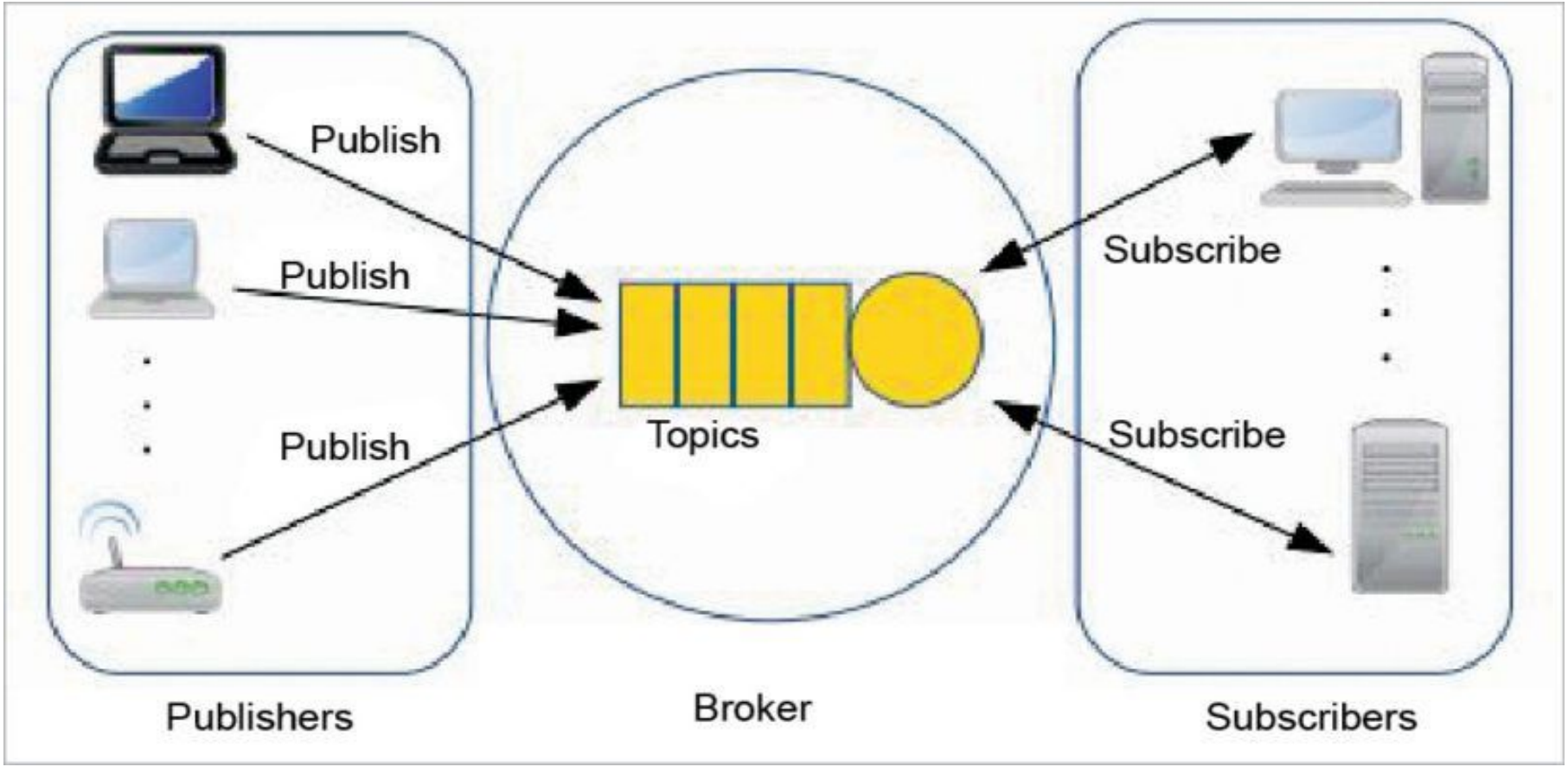

CoAP Environment
REST Internet
REST-CoAP
Proxy
CoAP Server
CoAP Clients
CoAP Communication
HTTP Communication

MQTT Protocol

MQTT (Message Queue Telemetry Transport), a messaging protocol, was developed by Andy Stanford-Clark of IBM and Arlen Nipper of Arcom in 1999. It is mostly used for remote monitoring in IoT. Its primary task is to acquire data from many devices and transport it to the IT infrastructure. MQTT connects devices and networks with applications and middleware. A hub-and-spoke architecture is natural for MQTT. All the devices connect to data concentrator servers like IBM’s new MessageSight appliance. MQTT protocols work on top of TCP to provide simple and reliable streams of data.

MQTT Protocol consists of three main components: **subscriber, publisher and broker**. The publisher generates the data and transmits the information to subscribers through the broker. The broker ensures security by cross-checking the authorisation of publishers and subscribers.

MQTT Protocol is the preferred option for IoT based devices, and is able to provide efficient information-routing functions to small, cheap, low-memory and power-consuming devices in vulnerable and low bandwidth based netw


Publish
Publish
Publish
Topics
Subscribe
Subscribe
Publishers
Broker
Subscribers