Internet of Things

DR. RAIHAN UL ISLAM

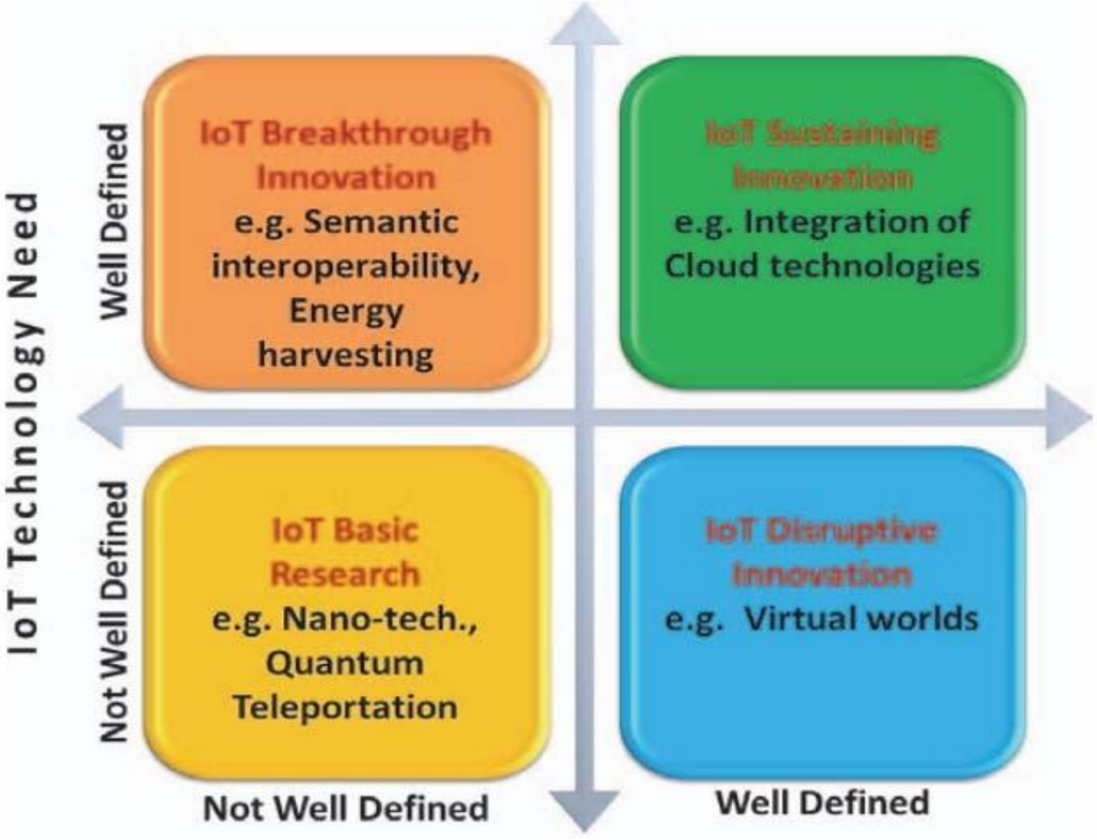
ASSOCIATE PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

ROOM NO# AB3-1003
EMAIL: RAIHAN.ISLAM@EWUBD.EDU

MOBILE: +8801992392611

Convergence of Domains



IoT Technology Need

Well Defined

IoT Breakthrough Innovation
e.g. Semantic interoperability, Energy harvesting

IoT Sustaining Innovation
e.g. Integration of Cloud technologies

Not Well Defined

IoT Basic Research
e.g. Nano-tech., Quantum Teleportation

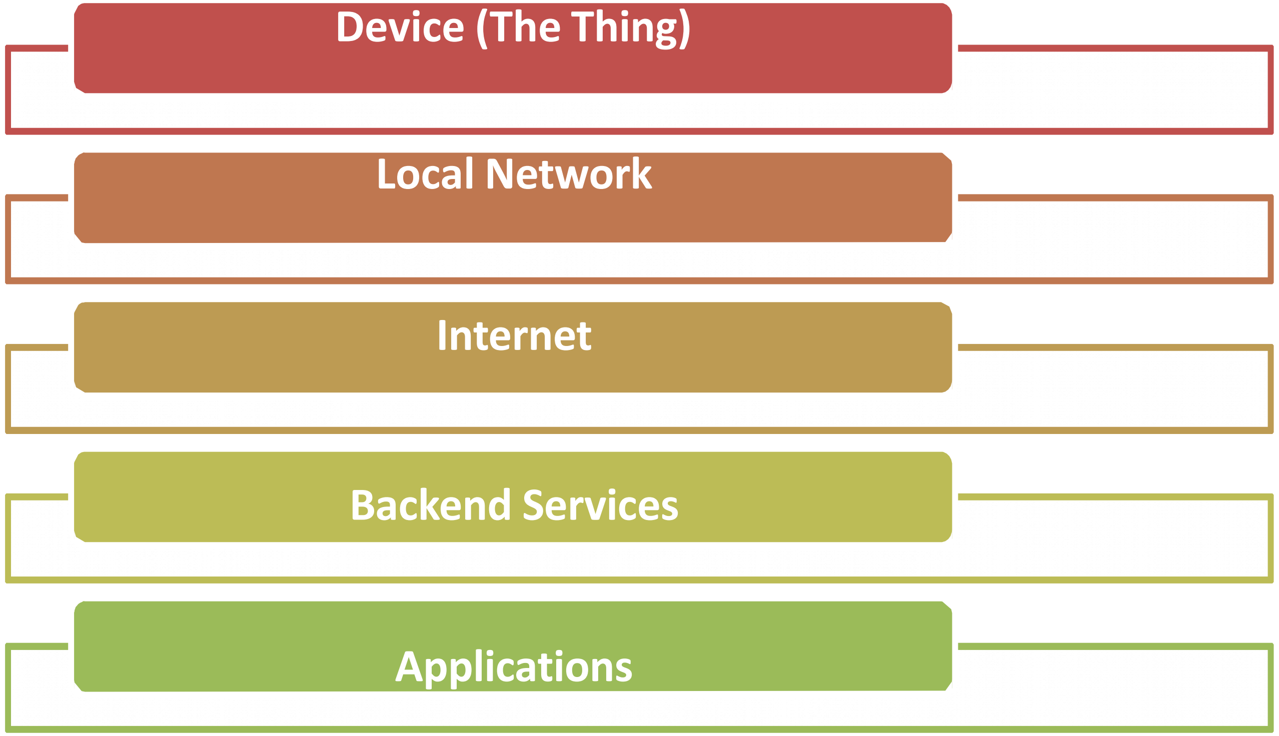
IoT Disruptive Innovation
e.g. Virtual worlds

Not Well Defined

Well Defined

Source: O. Vermesan, P. Friess**, “**Internet of Things – Converging Technologies for Smart Environments and Integrated Ecosystems”, River Publishers, Series in Communications, 2013

IoT Components



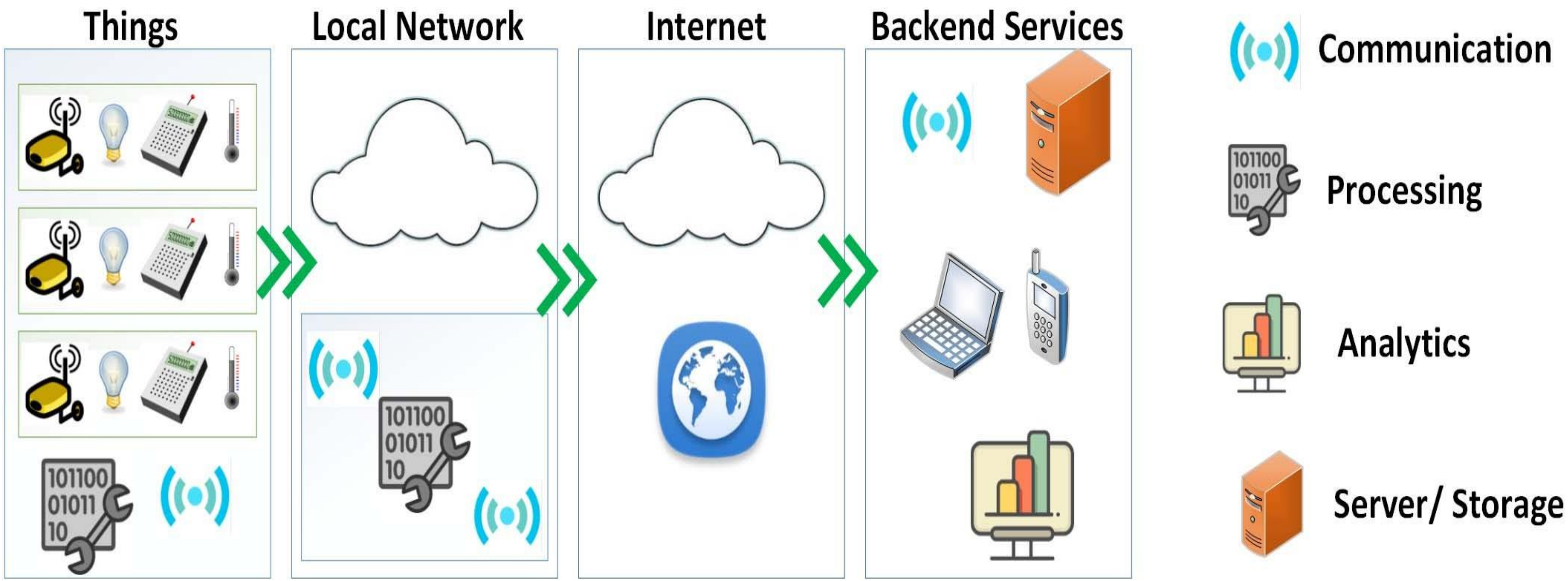
Device (The Thing)

Local Network

Internet

Backend Services

Applications



Things

Local Network

Internet

Backend Services

Communication

Processing

Analytics

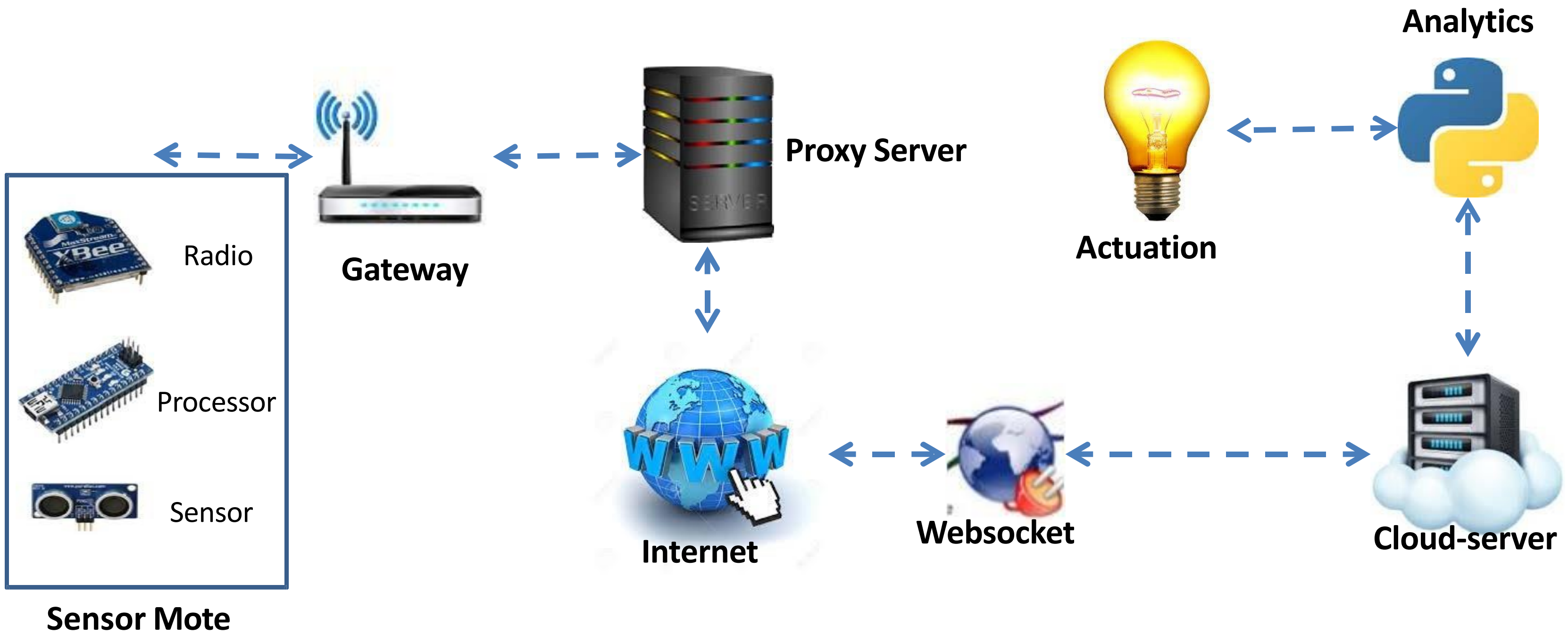
Server/ Storage

Functional Components of IoT

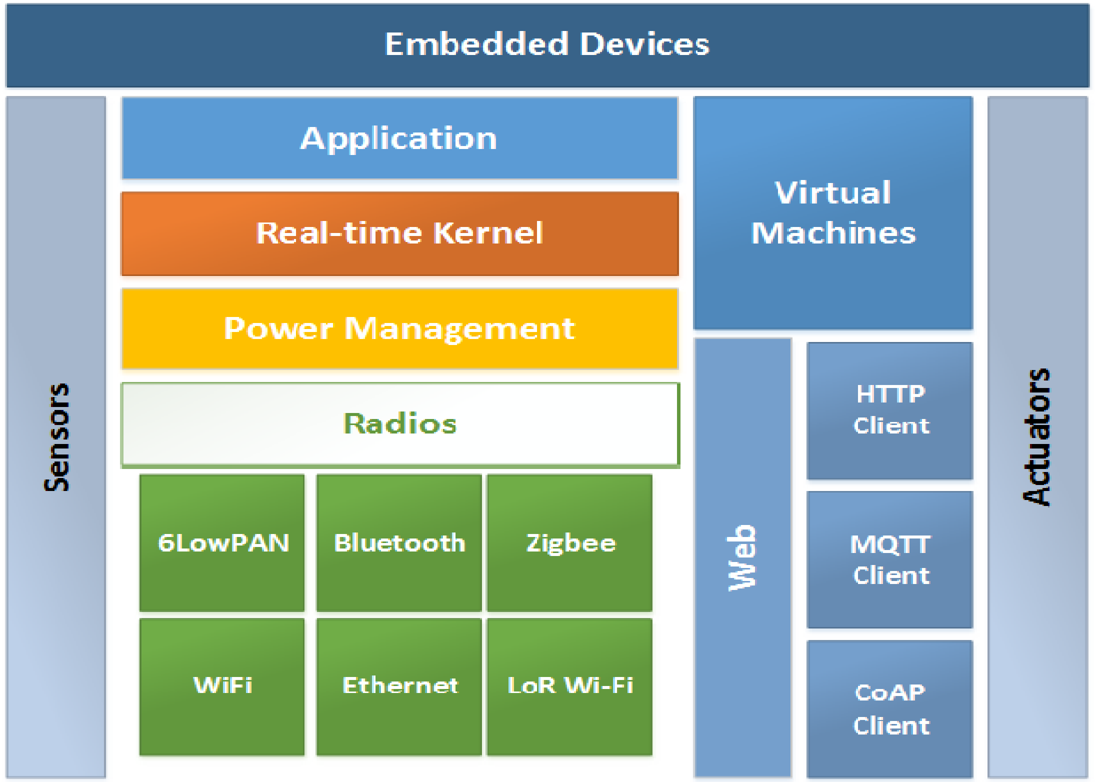
- ✓ Component for interaction and communication with other IoT devices
- ✓ Component for processing and analysis of operations
- ✓ Component for Internet interaction
- ✓ Components for handling Web services of applications
- ✓ Component to integrate application services
- ✓ User interface to access IoT

Source: O Vermesan, P. Friess**, “**Internet of Things – Converging Technologies for Smart Environments and Integrated Ecosystems”, River Publishers, Series in Communications, 2013

An Example IoT Implementation



IoT Interdependencies



Embedded Devices

Sensors

Application

Real-time Kernel

Power Management

Radios

6LowPAN

Bluetooth

Zigbee

WiFi

Ethernet

LoR Wi-Fi

Virtual Machines

Web

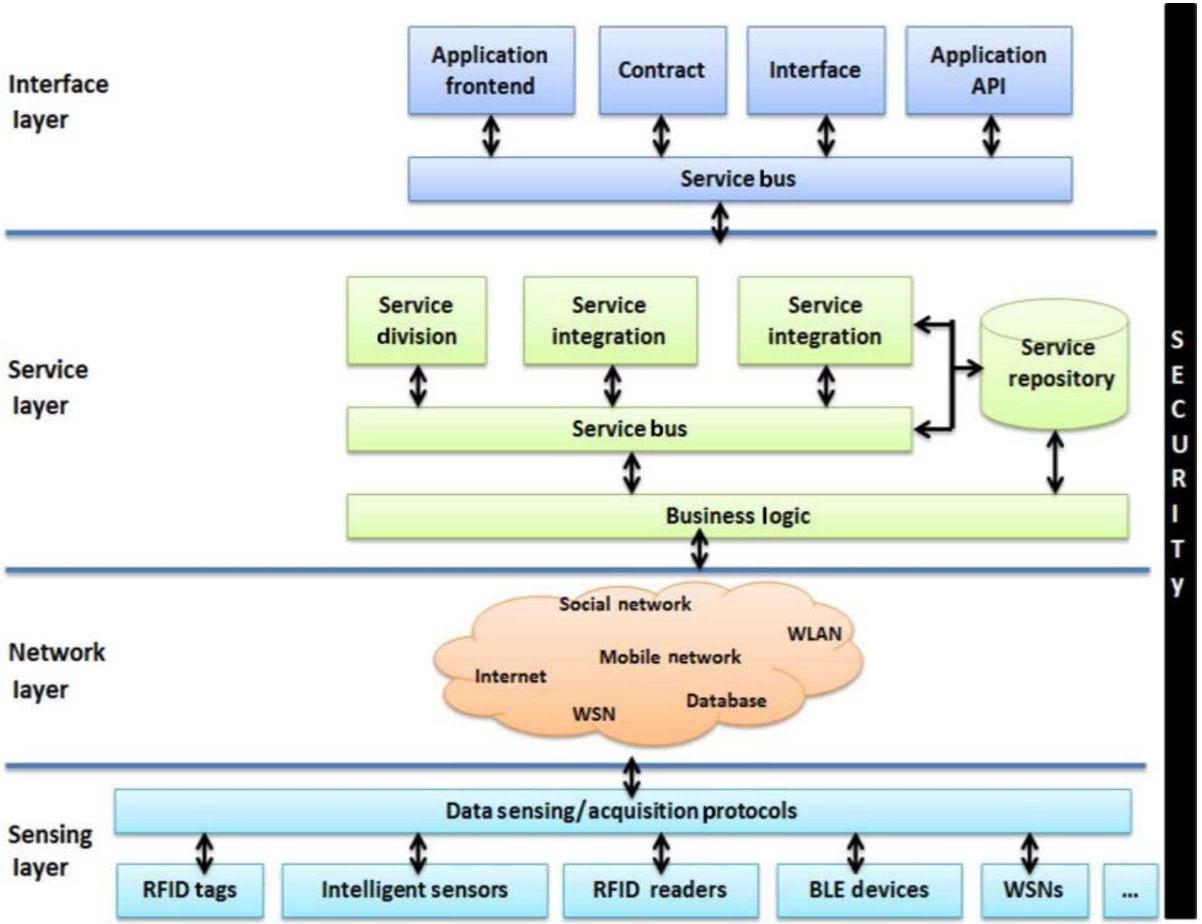
HTTP Client

MQTT Client

CoAP Client

Actuators

IoT Service Oriented Architecture



Source: Li Da Xu, Wu He, and Shancang Li, “Internet of Things in Industries: A Survey “, IEEE Transactions on Industrial Informatics, Vol. 10, No. 4, Nov. 2014.

IoT Categories

✓ Industrial IoT

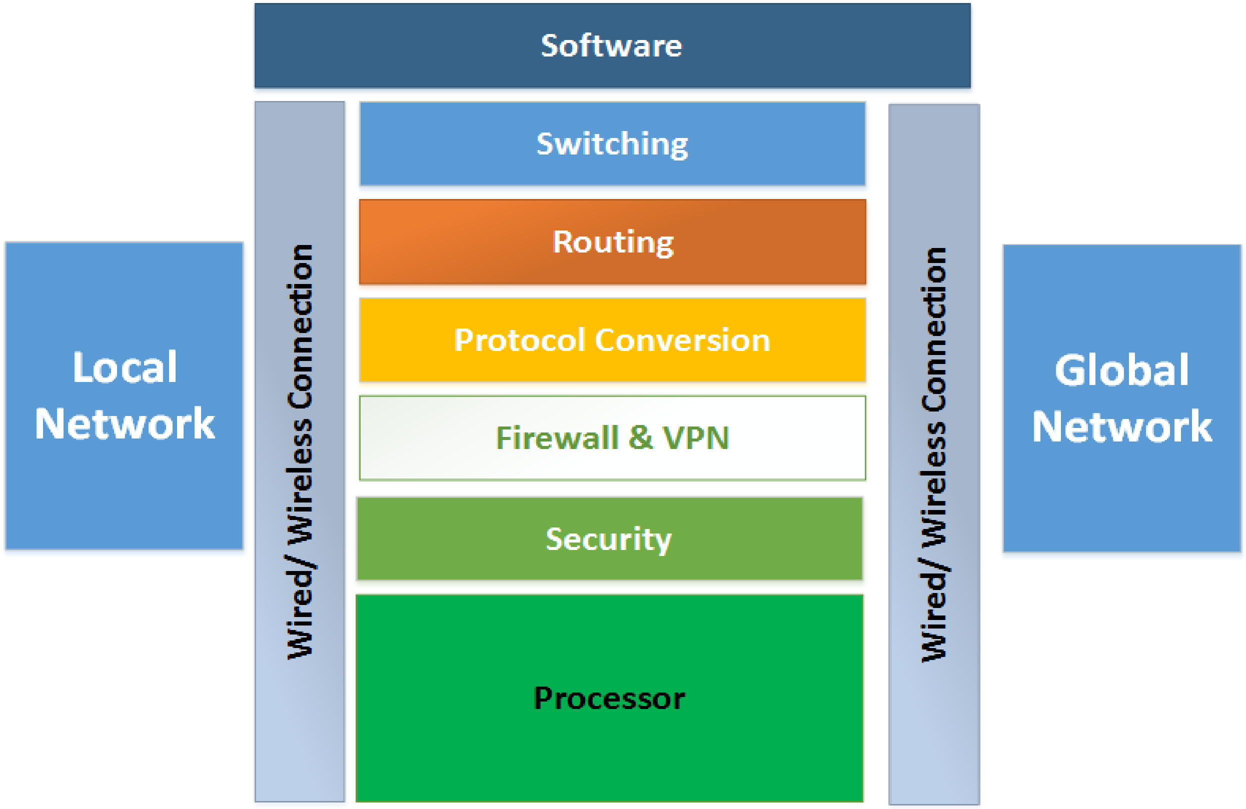
- IoT device connects to an IP network and the global Internet.
- Communication between the nodes done using regular as well as industry specific technologies.

✓ Consumer IoT

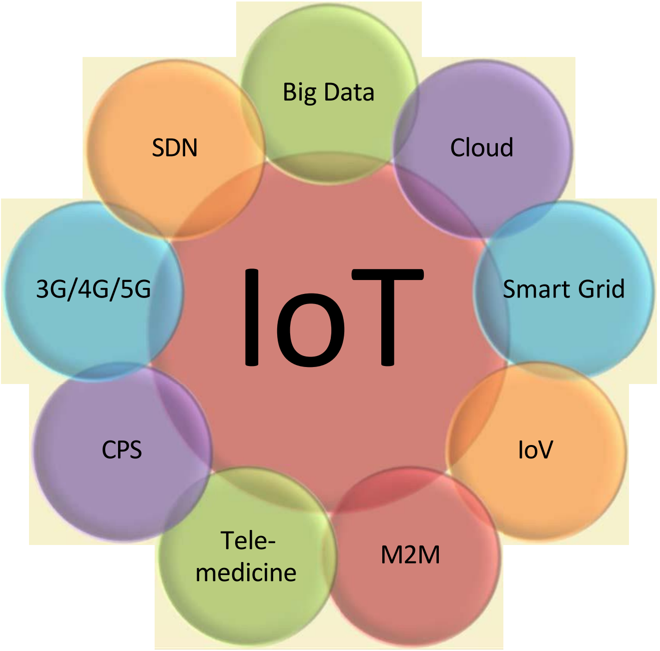
- IoT device communicates within the locally networked devices.
- Local communication is done mainly via Bluetooth, Zigbee or WiFi.
- Generally limited to local communication by a Gateway

IoT Gateways

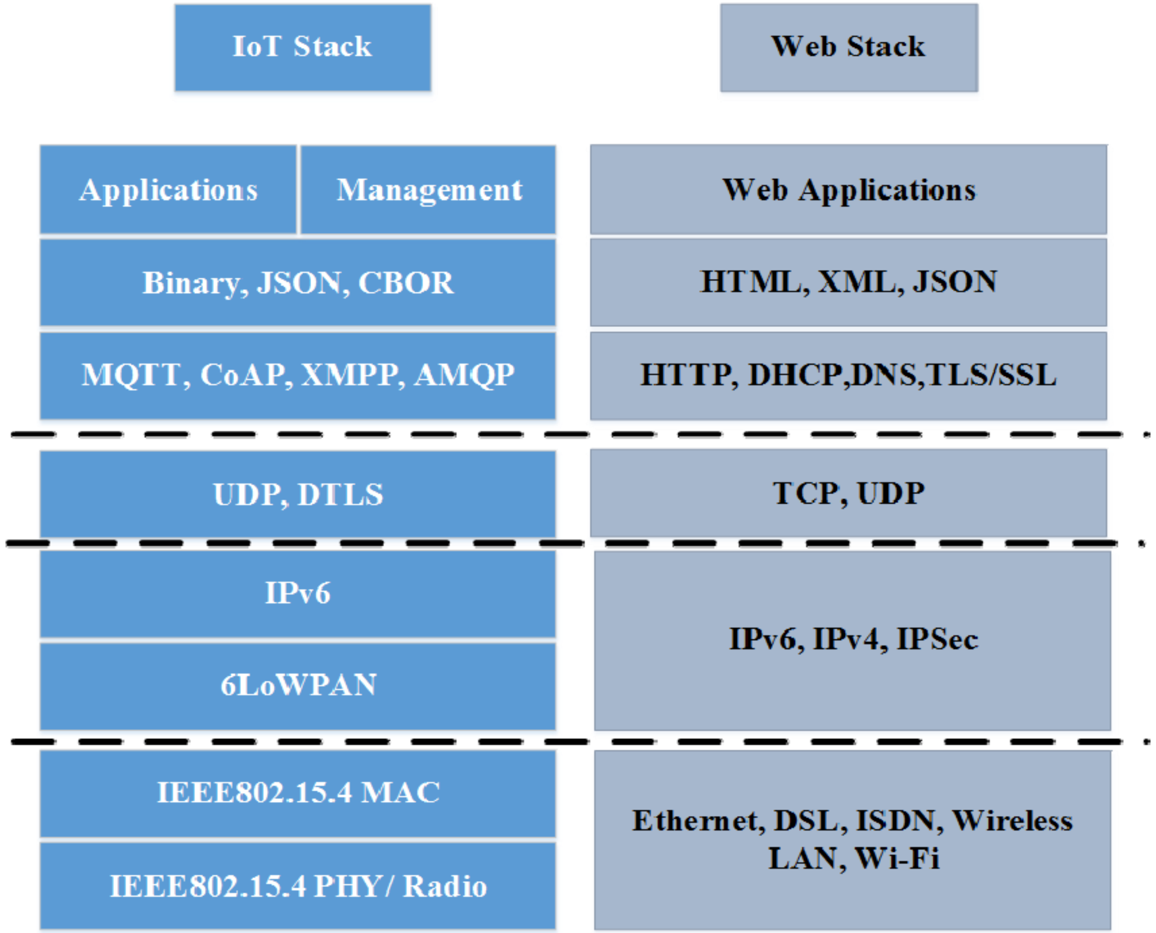
✓



IoT and Associated Technologies

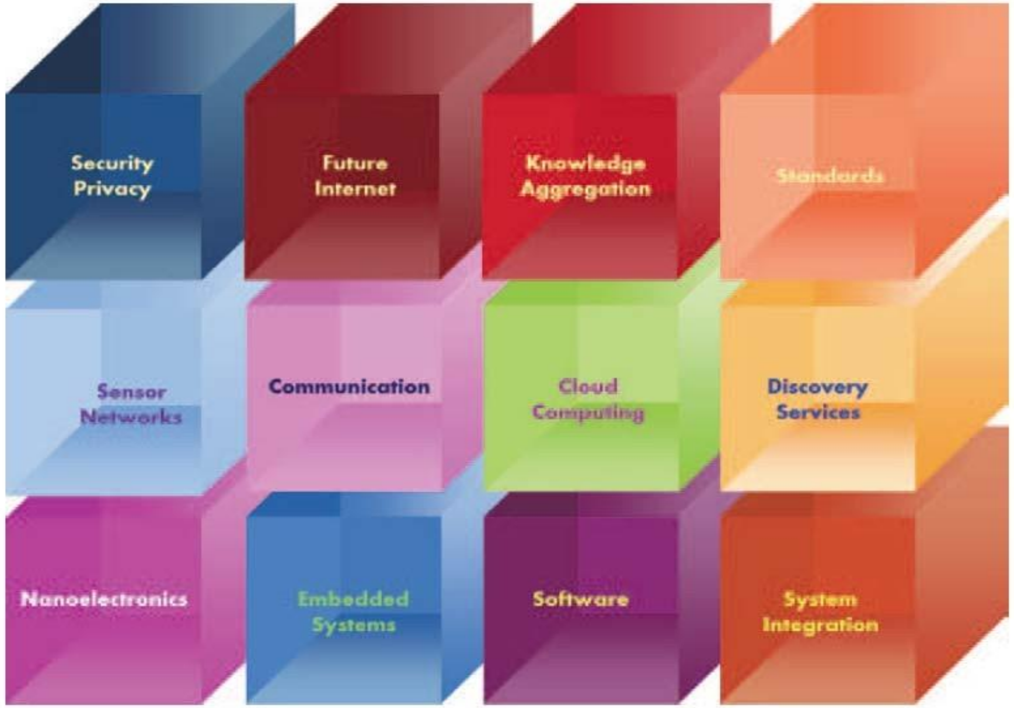


Technical Deviations from Regular Web



IoT Stack		Web Stack
Applications	Management	Web Applications
Binary, JSON, CBOR		HTML, XML, JSON
MQTT, CoAP, XMPP, AMQP		HTTP, DHCP,DNS,TLS/SSL
UDP, DTLS		TCP, UDP
IPv6		IPv6, IPv4, IPSec
6LoWPAN		
IEEE802.15.4 MAC		Ethernet, DSL, ISDN, Wireless LAN, Wi-Fi
IEEE802.15.4 PHY/ Radio		

Key Technologies for IoT



Source: O Vermesan, P. Friess**, “**Internet of Things – Converging Technologies for Smart Environments and Integrated Ecosystems”, River Publishers, Series in Communications, 2013

IoT Challenges

✓ Security

✓ Scalability

✓ Energy efficiency

✓ Bandwidth management

✓ Modeling and Analysis

✓ Interfacing

✓ Interoperability

✓ Data storage

✓ Data Analytics

✓ Complexity management (e.g., SDN)

Considerations

- ✓ Communication between the IoT device(s) and the outside world dictates the network architecture.
- ✓ Choice of communication technology dictates the IoT device hardware requirements and costs.
- ✓ Due to the presence of numerous applications of IoT enabled devices, a single networking paradigm not sufficient to address all the needs of the consumer or the IoT device.

✓

Complexity of Networks

- ✓ Growth of networks
- ✓ Interference among devices
- ✓ Network management
- ✓ Heterogeneity in networks
- ✓ Protocol standardization within networks

Source: O Vermesan, P. Friess**, "**Internet of Things – Converging Technologies for Smart Environments and Integrated Ecosystems", River Publishers, Series in Communications, 2013

Wireless Networks

- Traffic and load management
- Variations in wireless networks – Wireless Body Area Networks and other Personal Area Networks
- Interoperability
- Network management
- Overlay networks

Source: O. Vermesan, P. Friess**, "**Internet of Things – Converging Technologies for Smart Environments and Integrated Ecosystems", River Publishers, Series in Communications, 2013

Scalability

- Flexibility within Internet
- IoT integration
- Large scale deployment
- Real-time connectivity of billions of devices

Introduction

- ✓ RFID is an acronym for “radio-frequency identification”
- ✓ Data digitally encoded in RFID tags, which can be read by a reader.
- ✓ Somewhat similar to barcodes.
- ✓ Data read from tags are stored in a database by the reader.
- ✓ As compared to traditional barcodes and QR codes, RFID tag data can be read outside the line-of-sight.

Source: “How does RFID work?” AB&R (Online)

RFID Features

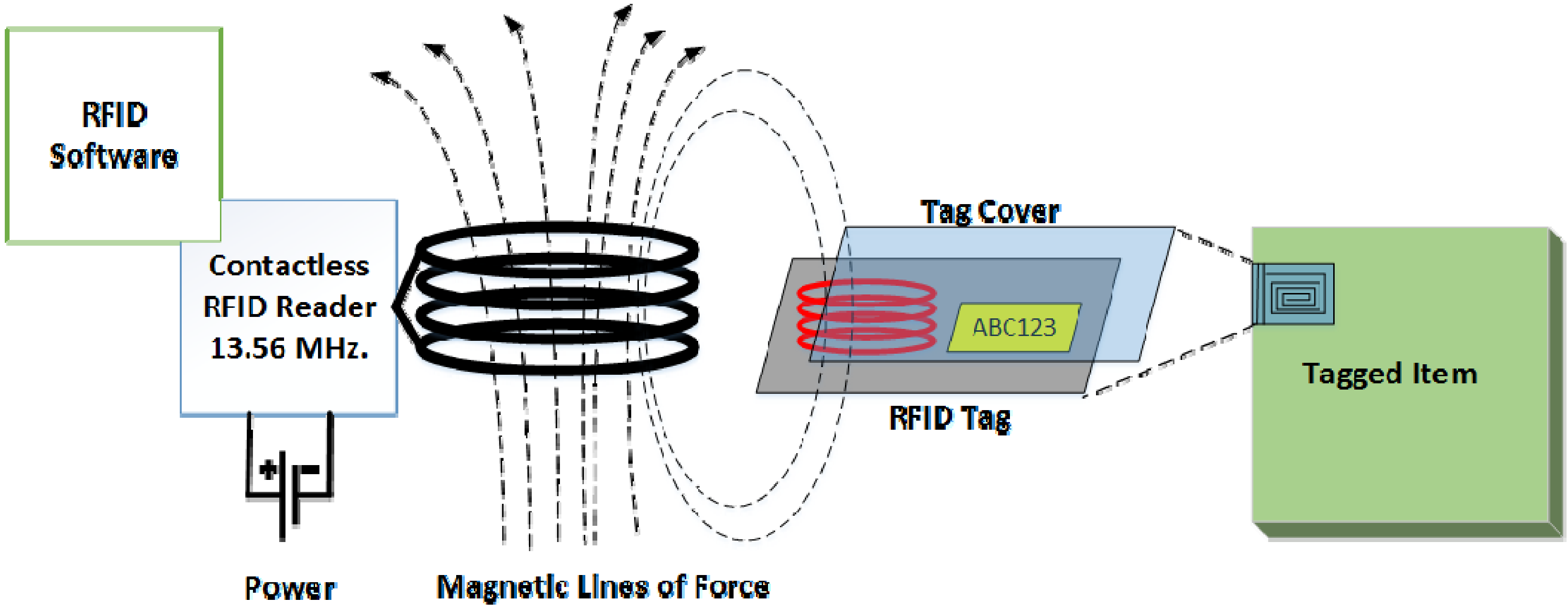
- ✓ RFID tag consists of an integrated circuit and an antenna.
- ✓ The tag is covered by a protective material which also acts as a shield against various environmental effects.
- ✓ Tags may be passive or active.
- ✓ Passive RFID tags are the most widely used.
- ✓ Passive tags have to be powered by a reader inductively before they can transmit information, whereas active tags have their own power supply.

Source: "How does RFID work?" AB&R (Online)

Working Principle

- ✓ Derived from Automatic Identification and Data Capture (AIDC) technology.
- ✓ AIDC performs object identification, object data collection and mapping of the collected data to computer systems with little or no human intervention.
- ✓ AIDC uses wired communication
- ✓ RFID uses radio waves to perform AIDC functions.
- ✓ The main components of an RFID system include an RFID tag or smart label, an RFID reader, and an antenna.

Source: "How does RFID work?" AB&R (Online)


RFID Software
Contactless RFID Reader 13.56 MHz.
Power
Magnetic Lines of Force
Tag Cover
ABC123
RFID Tag
Tagged Item

Applications

- ✓ Inventory management
- ✓ Asset tracking
- ✓ Personnel tracking
- ✓ Controlling access to restricted areas
- ✓ ID badging
- ✓ Supply chain management
- ✓ Counterfeit prevention (e.g. in the pharmaceutical industry)

Source: "How does RFID work?" AB&R (Online)

ZigBee

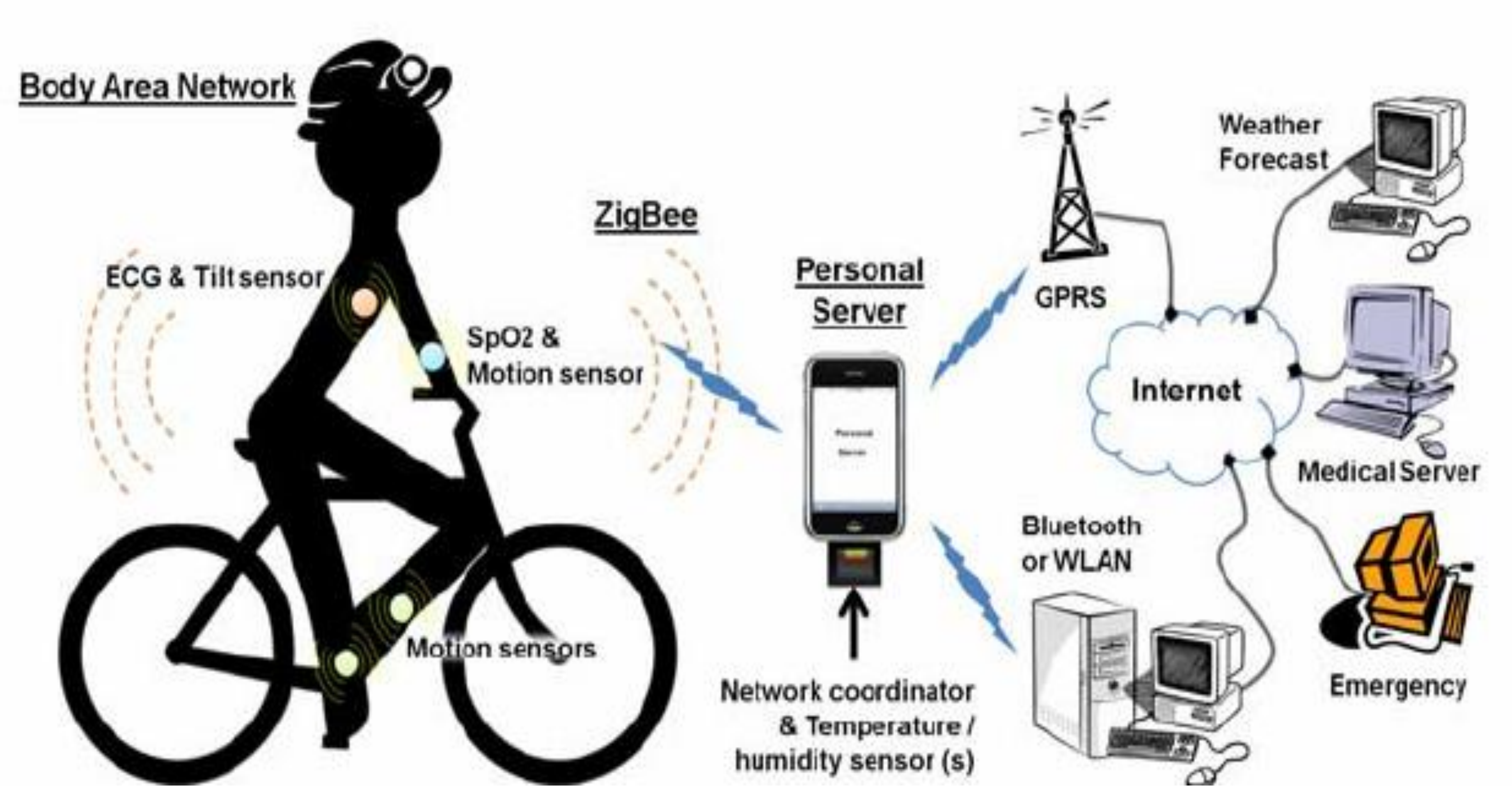
ZigBee communication protocol :

ZigBee is newly developed technology that works on IEEE standard 802.15.4, which can be used in the wireless sensor network (WSN).

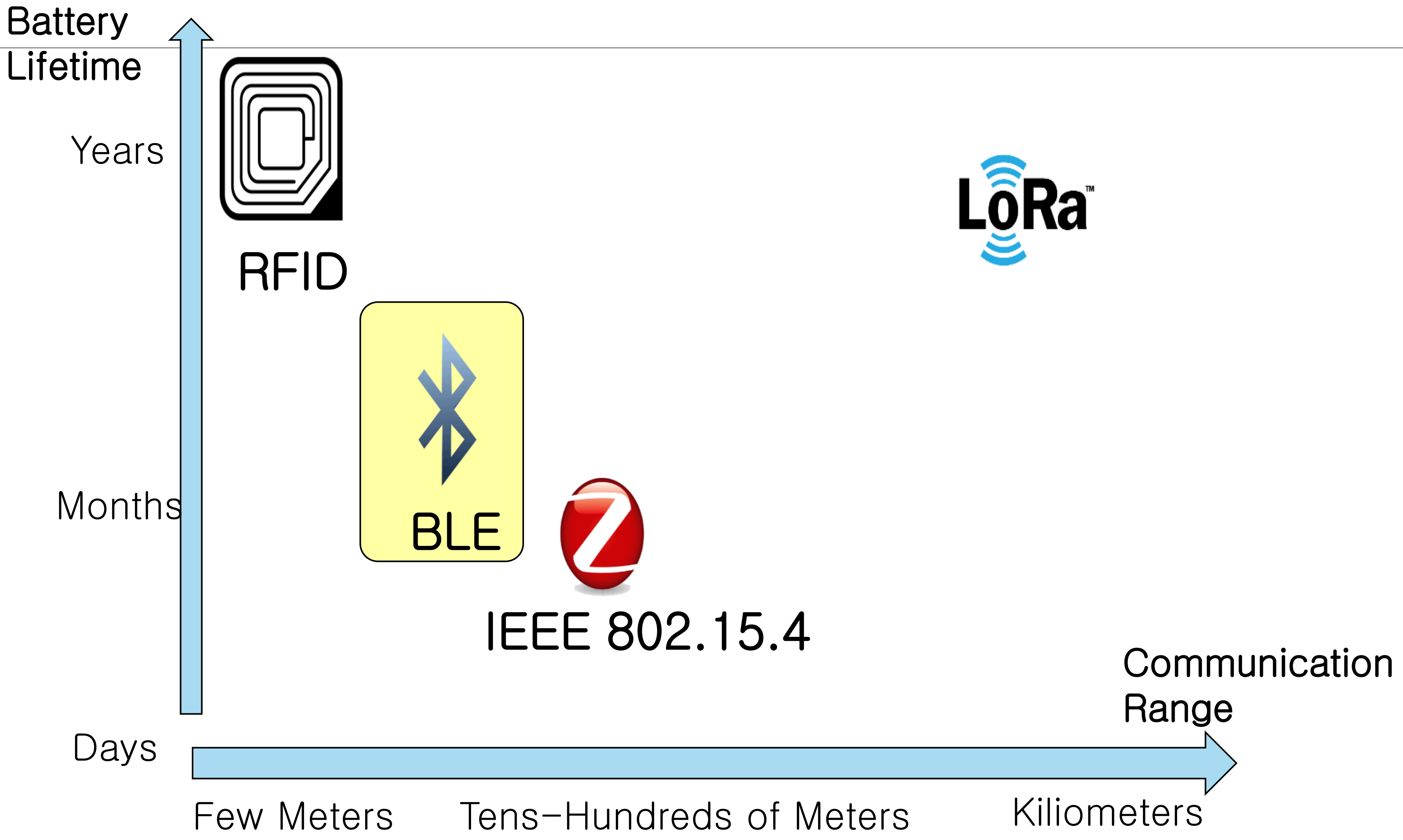
The low data rates, low power consumption, low cost are main features of ZigBee.

The **ZigBee** technology is designed to carry small amounts of data over a short distance while consuming very little power.

It is a short-range communication standard like Bluetooth and Wi-Fi, covering range of 10 to 100 meters.


Body Area Network
ZigBee
ECG & Tilt sensor
SpO2 &
Motion sensor
Motion sensors
Personal
Server
Personal
Server
Network coordinator
& Temperature /
humidity sensor (s)
GPRS
Weather
Forecast
Internet
Medical Server
Bluetooth
or WLAN
Emergency

Bluetooth Low Energy (BLE)



Battery Lifetime

Years

Months

Days

RFID

BLE

IEEE 802.15.4

Communication Range

Few Meters

Tens-Hundreds of Meters

Kiliometers

How much Energy does Traditional Bluetooth Use?

Traditional Bluetooth is *connection oriented.* When a device is connected, a link is maintained, even if there is no data flowing.

Sniff modes allow devices to sleep, reducing power consumption to give months of battery life

Peak transmit current is typically around 25mA

Even though it has been independently shown to be lower power than other radio standards, it is still not low enough power for coin cells and energy harvesting applications

What is Bluetooth Low Energy?

Bluetooth low energy is a NEW, open, short range radio technology

- Blank sheet of paper design
- Different to Bluetooth classic (BR/EDR)
- Optimized for ultra low power
- Enable coin cell battery use cases
  - < 20mA peak current
  - < 5 uA average current

Basic Concepts of Bluetooth 4.0

Everything is optimized for lowest power consumption

- Short packets reduce TX peak current
- Short packets reduce RX time
- Less RF channels to improve discovery and connection time
- Simple state machine
- Single protocol
- Etc.

Bluetooth Low Energy Factsheet

Range:	~ 150 meters open field
Output Power:	~ 10 mW (10dBm)
Max Current:	~ 15 mA
Latency:	3 ms
Topology:	Star
Connections:	> 2 billion
Modulation:	GFSK @ 2.4 GHz
Robustness:	Adaptive Frequency Hopping, 24 bit CRC
Security:	128bit AES CCM
Sleep current:	~ 1µA
Modes:	Broadcast, Connection, Event Data Models, Reads, Writes

Bluetooth Low Energy Factsheet (2)

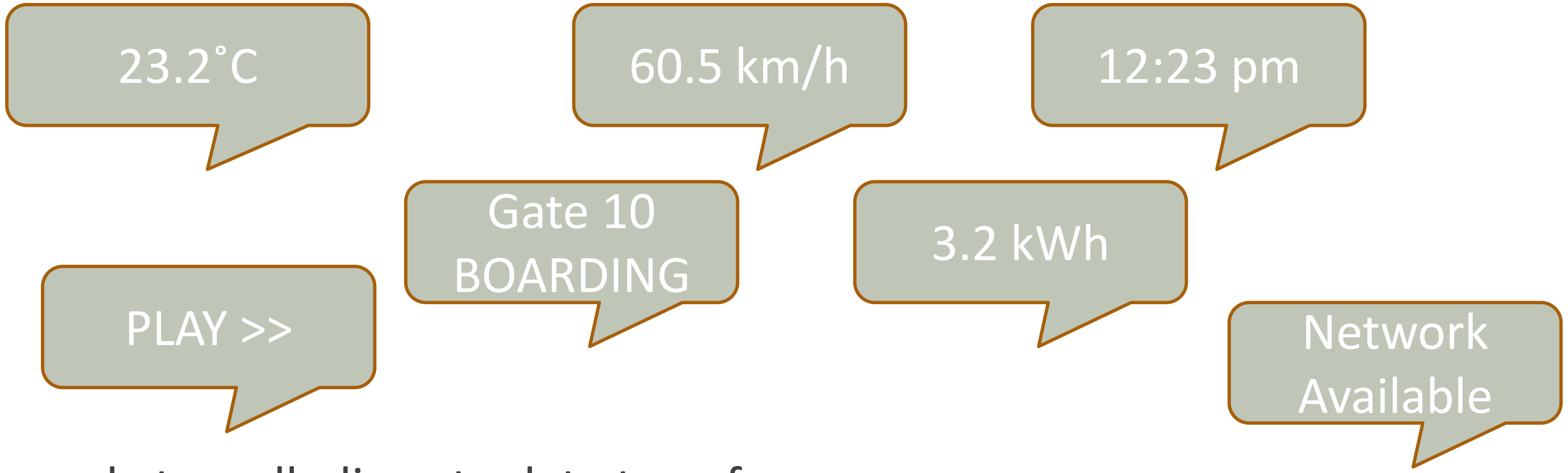
Data Throughput

- For Bluetooth low energy, data throughput is not a meaningful parameter. It does not support streaming.

- It has a data rate of 1Mbps, but is not optimized for file transfer.

- It is designed for sending small chunks of data (exposing state)

Designed for Exposing State



23.2˚C

60.5 km/h

12:23 pm

Gate 10 BOARDING

3.2 kWh

PLAY >>

Network Available

It’s good at small, discrete data transfers.

Data can triggered by local events.

Data can be read at any time by a client.

6LoWPAN

Introduction

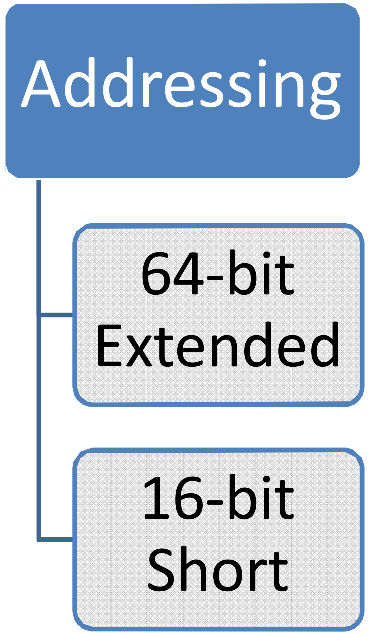
- ✓ Low-power Wireless Personal Area Networks over IPv6.
- ✓ Allows for the smallest devices with limited processing ability to transmit information wirelessly using an Internet protocol.
- ✓ Allows low-power devices to connect to the Internet.
- ✓ Created by the Internet Engineering Task Force (IETF) - RFC 5933 and RFC 4919.

Source: T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik , JP. Vasseur, R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", IETF, Standards Track, Mar. 2012

Features of 6LoWPANs

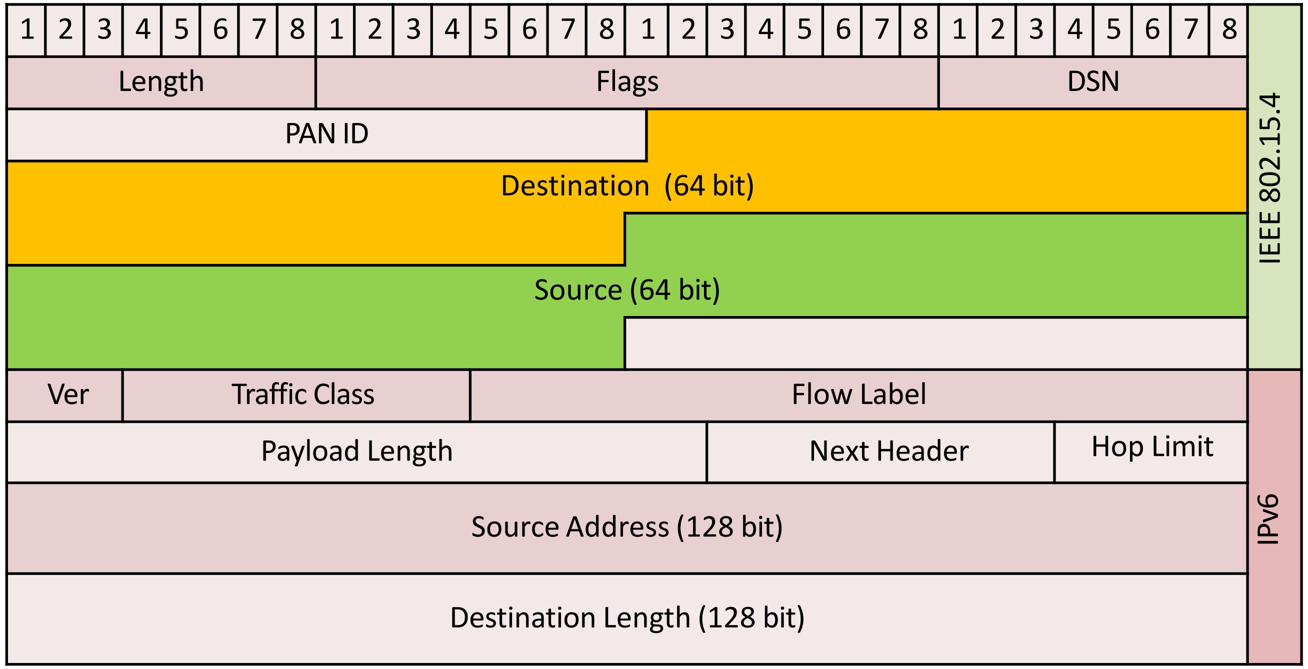
- ✓ Allows IEEE 802.15.4 radios to carry 128-bit addresses of Internet Protocol version 6 (IPv6).
- ✓ Header compression and address translation techniques allow the IEEE 802.15.4 radios to access the Internet.
- ✓ IPv6 packets compressed and reformatted to fit the IEEE 802.15.4 packet format.
- ✓ Uses include IoT, Smart grid, and M2M applications.

Addressing in 6LoWPAN



- 64-bit addresses: globally unique
- 16 bit addresses: PAN specific; assigned by PAN coordinator
- IPv6 multicast not supported by 802.15.4
- IPv6 packets carried as link layer broadcast frames

6LowPAN Packet Format



1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8 1 2 3 4 5 6 7 8

Length

Flags

DSN

PAN ID

Destination (64 bit)

Source (64 bit)

Ver

Traffic Class

Flow Label

Payload Length

Next Header

Hop Limit

Source Address (128 bit)

Destination Length (128 bit)

IEEE 802.15.4

IPv6

Header Type: Dispatch Header

1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
0 1		Dispatch						Type Specific Header																							

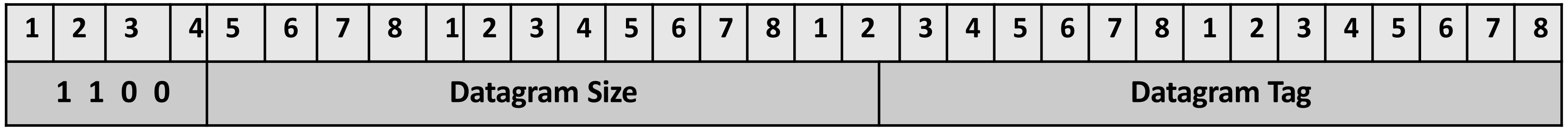
- **Dispatch:** Initiates communication
- **0,1**: Identifier for Dispatch Type
- **Dispatch**:
  - 6 bits
  - Identifies the next header type
- **Type Specific Header:**
  - Determined by Dispatch header

Header Type: Mesh Addressing Header

1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8
1 0		V	F	Hops Left				Originator Address												Final Address											

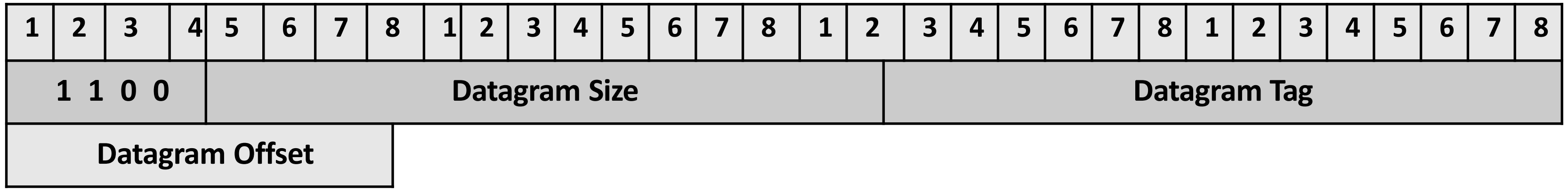
- **1,0**: ID for Mesh Addressing Header
- **V**: ‘0’ if originator is 64-bit extended address, ‘1’ if 16-bit address
- **F**: ‘0’ if destination is 64-bit addr., ‘1’ if 16-bit addr.
- **Hops Left**: decremented by each node before sending to next hop

Header Type: Fragmentation Header



<table>
<tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr>
<tr><td colspan="4">1 1 0 0</td><td colspan="14">Datagram Size</td><td colspan="14">Datagram Tag</td></tr>
</table>

(a) First Fragment

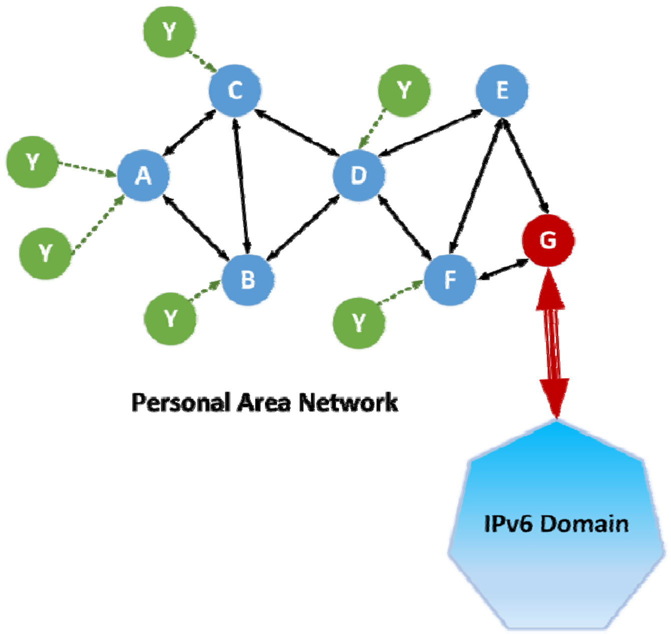


<table>
<tr><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td></tr>
<tr><td colspan="4">1 1 0 0</td><td colspan="14">Datagram Size</td><td colspan="14">Datagram Tag</td></tr>
<tr><td colspan="8">Datagram Offset</td></tr>
</table>

(b) Subsequent Fragment

6LoWPAN Routing Considerations

- ✓ Mesh routing within the PAN space.
- ✓ Routing between IPv6 and the PAN domain
- ✓ Routing protocols in use:
  - **LOADng**
  - **RPL**



LOADng Routing

- ✓ Derived from AODV and extended for use in IoT.
- ✓ Basic operations of LOADng include:
  - Generation of **Route Requests (RREQs)** by a LOADng Router (originator) for discovering a route to a destination,
  - **Forwarding of such RREQs** until they reach the destination LOADng Router,
  - Generation of **Route Replies (RREPs)** upon receipt of an RREQ by the indicated destination, and unicast hop-by-hop forwarding of these RREPs towards the originator.

Source: Clausen, T.; Colin de Verdiere, A.; Yi, J.; Niktash, A.; Igarashi, Y.; Satoh, H.; Herberg, U.; Lavenu, C. et al. (January 2016). *The Lightweight On-demand Ad hoc Distance-vector Routing Protocol - Next Generation (LOADng)*. IETF. I-D draft-clausen-lln-loadng-14

- If a route is detected to be broken, a **Route Error (RERR)** message is returned to the originator of that data packet to inform the originator about the route breakage.
- **Optimized flooding** is supported, reducing the overhead incurred by RREQ generation and flooding.
- Only the destination is permitted to respond to an RREQ.
- Intermediate LOADng Routers are explicitly prohibited from responding to RREQs, even if they may have active routes to the sought destination.
- RREQ/RREP messages generated by a given LOADng Router share a single unique, monotonically increasing sequence number.

Source: Clausen, T.; Colin de Verdiere, A.; Yi, J.; Niktash, A.; Igarashi, Y.; Satoh, H.; Herberg, U.; Lavenu, C. et al. (January 2016). *The Lightweight On-demand Ad hoc Distance-vector Routing Protocol - Next Generation (LOADng)*. IETF. I-D draft-clausen-lln-loadng-14

RPL Routing

- ✓ Distance Vector IPv6 **routing protocol for lossy and low power networks.**
- ✓ Maintains routing topology using low rate beaconing.
- ✓ Beaconing rate increases on detecting inconsistencies (e.g. node/link in a route is down).
- ✓ Routing information included in the datagram itself.
- ✓ **Proactive**: Maintaining routing topology.
- ✓ **Reactive**: Resolving routing inconsistencies.

Source: T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik , JP. Vasseur, R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", IETF, Standards Track, Mar. 2012

- ✓ RPL separates packet processing and forwarding from the routing optimization objective, which helps in Low power Lossy Networks (LLN).
- ✓ RPL supports message confidentiality and integrity.
- ✓ Supports Data-Path Validation and Loop Detection
- ✓ Routing optimization objectives include
  - minimizing energy
  - minimizing latency
  - satisfying constraints (w.r.t node power, bandwidth, etc.)

Source: T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik , JP. Vasseur, R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", IETF, Standards Track, Mar. 2012

- ✓ RPL operations require bidirectional links.
- ✓ In some LLN scenarios, those links may exhibit asymmetric properties.
- ✓ It is required that the reachability of a router be verified before the router can be used as a parent.

Source: T. Winter, P. Thubert, A. Brandt, J. Hui, R. Kelsey, P. Levis, K. Pister, R. Struik , JP. Vasseur, R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", IETF, Standards Track, Mar. 2012

RFID

Functionality-based IoT Protocol Organization

- ✓ **Connectivity** (6LowPAN, RPL)
- ✓ **Identification** (EPC, uCode, IPv6, URIs)
- ✓ **Communication / Transport** (WiFi, Bluetooth, LPWAN)
- ✓ **Discovery** (Physical Web, mDNS, DNS-SD)
- ✓ **Data Protocols** (MQTT, CoAP, AMQP, Websocket, Node)
- ✓ **Device Management** (TR-069, OMA-DM)
- ✓ **Semantic** (JSON-LD, Web Thing Model)
- ✓ **Multi-layer Frameworks** (Alljoyn, IoTivity, Weave, Homekit)

Source: Internet of Things Protocols (Online)

MQTT

Introduction

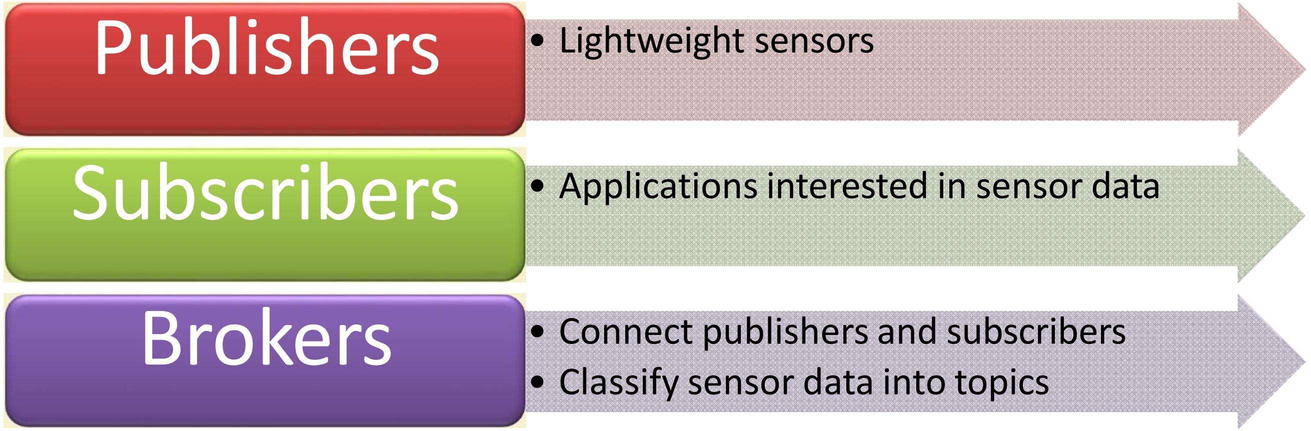
- ✓ **Message Queue Telemetry Transport.**
- ✓ ISO standard (ISO/IEC PRF 20922).
- ✓ It is a publish-subscribe-based lightweight messaging protocol for use in conjunction with the TCP/IP protocol.
- ✓ MQTT was introduced by IBM in 1999 and standardized by OASIS in 2013.
- ✓ Designed to provide connectivity (mostly embedded) between applications and middle-wares on one side and networks and communications on the other side.

Source: “MQTT”, Wikipedia (Online)

- A message broker controls the publish-subscribe messaging pattern.
- A topic to which a client is subscribed is updated in the form of messages and distributed by the message broker.
- Designed for:
    - Remote connections
    - Limited bandwidth
    - Small-code footprint

Source: “MQTT”, Wikipedia (Online)

MQTT Components



Publishers
- Lightweight sensors

Subscribers
- Applications interested in sensor data

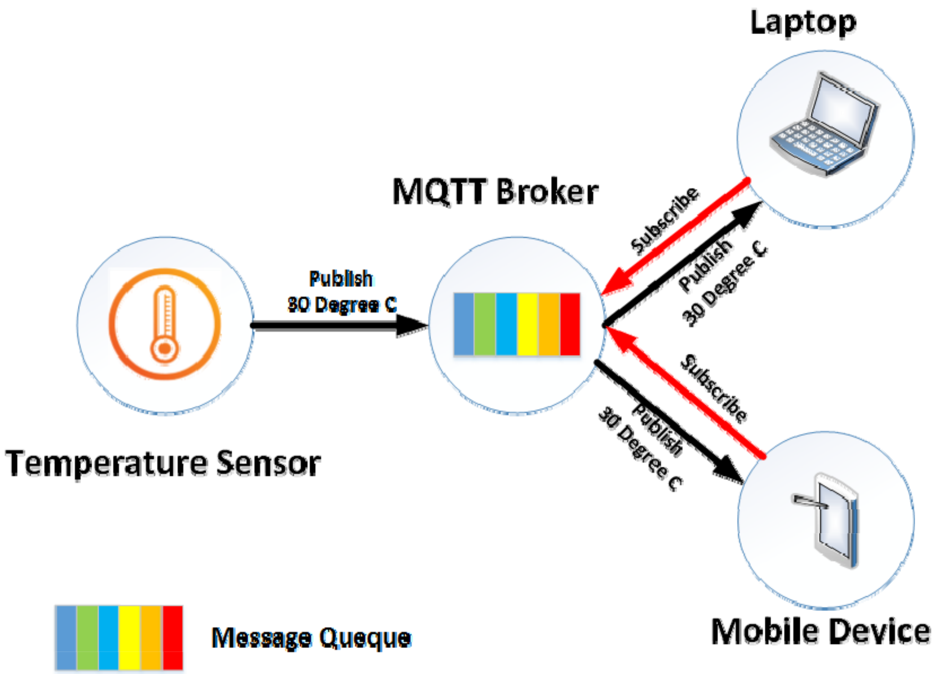
Brokers
- Connect publishers and subscribers
- Classify sensor data into topics

Source: “MQTT”, Wikipedia (Online)

MQTT Methods

- Connect
- Disconnect
- Subscribe
- Unsubscribe
- Publish

Source: “MQTT”, Wikipedia (Online)



Source: “MQTT 101 – How to Get Started with the lightweight IoT Protocol”, HiveMQ (Online)

Communication

- ✓ The protocol uses a **publish/subscribe** architecture (HTTP uses a request/response paradigm).
- ✓ Publish/subscribe is **event-driven** and enables messages to be pushed to clients.
- ✓ The central **communication point is the MQTT broker**, which is in charge of dispatching all messages between the senders and the rightful receivers.
- ✓ Each client that publishes a message to the broker, includes a **topic** into the message. The **topic is the routing information for the broker.**

Source: “MQTT 101 – How to Get Started with the lightweight IoT Protocol”, HiveMQ (Online)

- ✓ Each client that wants to receive messages subscribes to a certain topic and the broker delivers all messages with the matching topic to the client.
- ✓ Therefore the clients don’t have to know each other. They only communicate over the topic.
- ✓ This architecture enables highly scalable solutions without dependencies between the data producers and the data consumers.

Source: “MQTT 101 – How to Get Started with the lightweight IoT Protocol”, HiveMQ (Online)

MQTT Topics

- ✓ A topic is a **simple string** that can have more hierarchy levels, which are separated by a slash.
- ✓ A sample topic for sending temperature data of the living room could be *house/living-room/temperature*.
- ✓ On one hand the client (e.g. mobile device) can subscribe to the exact topic or on the other hand, it can use a **wildcard**.

Source: "MQTT 101 – How to Get Started with the lightweight IoT Protocol", HiveMQ (Online)

- ✓ The subscription to *house/+/temperature* would result in all messages sent to the previously mentioned topic *house/living-room/temperature,* as well as any topic with an arbitrary value in the place of living room, such as *house/kitchen/temperature*.
- ✓ The plus sign is a **single level wild card** and only allows arbitrary values for one hierarchy.
- ✓ If more than one level needs to be subscribed, such as, the entire sub-tree, there is also a **multilevel wildcard** (#).
- ✓ It allows to subscribe to all underlying hierarchy levels.
- ✓ For example *house/#* is subscribing to all topics beginning with *house*.

Source: "MQTT 101 – How to Get Started with the lightweight IoT Protocol", HiveMQ (Online)

Applications

- ✓ **Facebook Messenger** uses MQTT for online chat.
- ✓ **Amazon Web Services** use Amazon IoT with MQTT.
- ✓ **Microsoft Azure** IoT Hub uses MQTT as its main protocol for telemetry messages.
- ✓ The **EVRYTHNG IoT platform** uses MQTT as an M2M protocol for millions of connected products.
- ✓ **Adafruit** launched a free MQTT cloud service for IoT experimenters called Adafruit IO.

SMQTT

- ✓ **Secure MQTT** is an extension of MQTT which uses encryption based on lightweight attribute based encryption.
- ✓ The main advantage of using such encryption is the broadcast encryption feature, in which one message is encrypted and delivered to multiple other nodes, which is quite common in IoT applications.
- ✓ In general, the algorithm consists of four main stages: setup, encryption, publish and decryption.

Source: M. Singh, M. Rajan, V. Shivraj, and P. Balamuralidhar, "Secure MQTT for Internet of Things (IoT)," in Fifth International Conference on Communication Systems and Network Technologies (CSNT 2015), April 2015, pp. 746-751

- ✓ In the <u>setup phase</u>, the subscribers and publishers register themselves to the broker and get a master secret key according to their developer's choice of key generation algorithm.
- ✓ When the <u>data is published</u>, it is encrypted and published by the broker which sends it to the subscribers, which is finally decrypted at the subscriber end having the same master secret key.
- ✓ The key generation and encryption algorithms are not standardized.
- ✓ SMQTT is proposed only to enhance MQTT security features.

Source: M. Singh, M. Rajan, V. Shivraj, and P. Balamuralidhar, "Secure MQTT for Internet of Things (IoT)," in Fifth International Conference on Communication Systems and Network Technologies (CSNT 2015), April 2015, pp. 746-751

Thank you