IoT Architecture

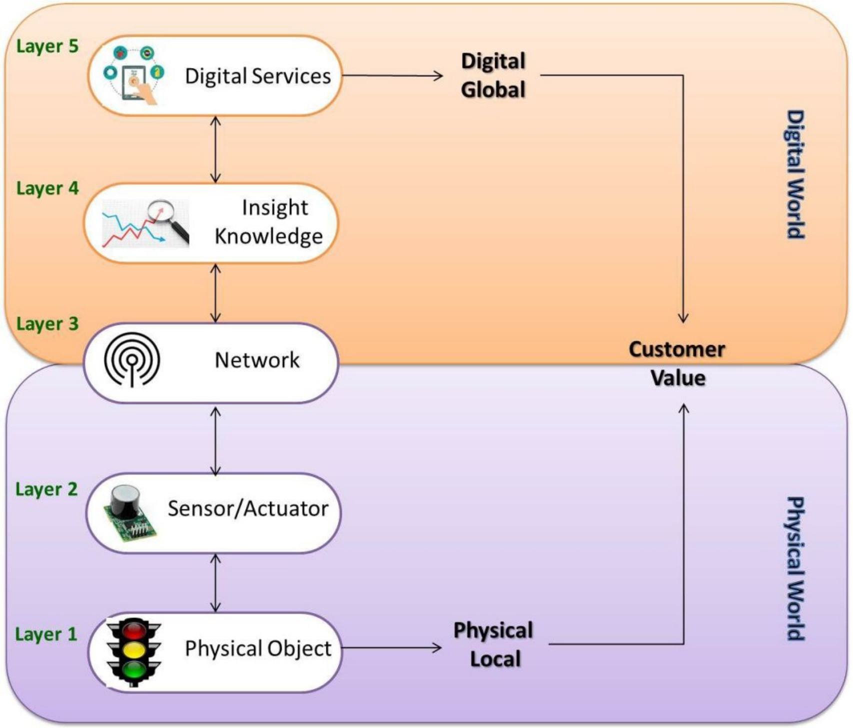
Middleware for Enterprise IoT

IoT Communication Protocols

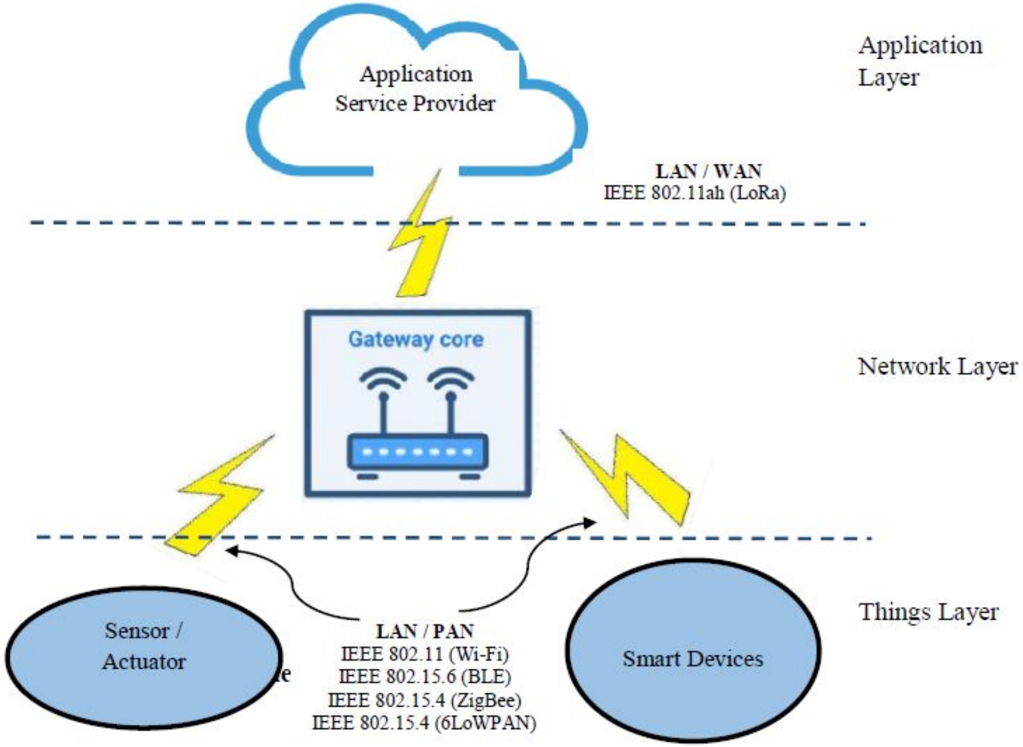
- REST & MQTT

Dr.L.S.Jayashree
Prof/CSE
PSG CT

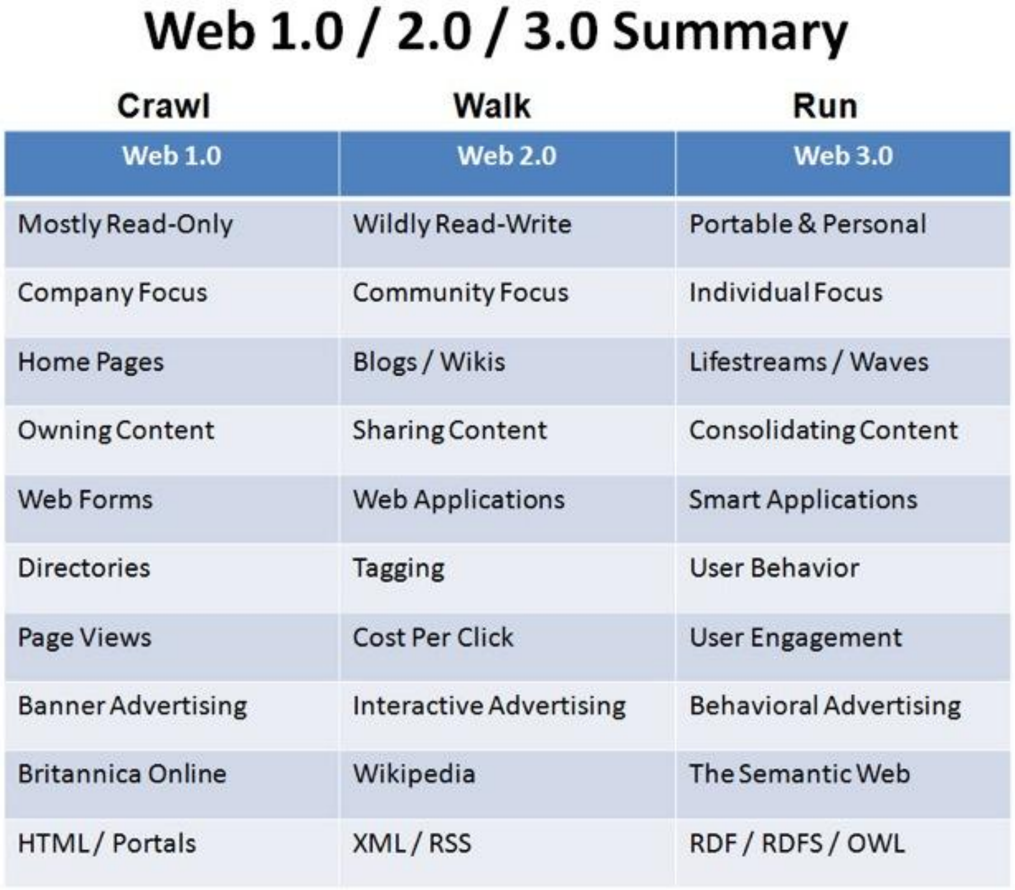
A Big Picture of IoT Digital Value Chain



IoT Communication Model



Web 1.0 to 3.0



Web 1.0 / 2.0 / 3.0 Summary

Crawl	Walk	Run
Web 1.0	Web 2.0	Web 3.0
Mostly Read-Only	Wildly Read-Write	Portable & Personal
Company Focus	Community Focus	Individual Focus
Home Pages	Blogs / Wikis	Lifestreams / Waves
Owning Content	Sharing Content	Consolidating Content
Web Forms	Web Applications	Smart Applications
Directories	Tagging	User Behavior
Page Views	Cost Per Click	User Engagement
Banner Advertising	Interactive Advertising	Behavioral Advertising
Britannica Online	Wikipedia	The Semantic Web
HTML / Portals	XML / RSS	RDF / RDFS / OWL

Web 1.0

It is the “readable” phrase of the World Wide Web with flat data.

sites and web users.

simply an information portal where users passively receive information without being given the opportunity to post reviews, comments, and feedback.

Web 2.0

It is the “writable” phrase of the World Wide Web with interactive data.

facilitates interaction between web users and sites, so it allows users to interact more freely with each other

encourages participation, collaboration, and information sharing.

Example applications are : Youtube, Wiki, Flickr, Facebook, and so on

Web 3.0

It is the “executable” phrase of Word Wide Web with dynamic applications, interactive services, and “machine-to-machine” interaction.

semantic web : computers can interpret information like humans and intelligently generate and distribute useful content tailored to the needs of users.

Machine generated data

.

- https://www.youtube.com/watch?v=bsNcjya56v8&feature=youtu.be

Web 4.0 – Web of Things

- The **Web of Things** (**WoT**) allows real-world objects to be part of the World Wide Web

- Similarly to what the Web (Application Layer) is to the Internet (Network Layer), the Web of Things provides an Application Layer that simplifies the creation of IoT applications

- Rather than re-inventing completely new standards, the Web of Things reuses existing and well-known Web standards used in the programmable Web (e.g., REST, HTTP, JSON)

Web 5.0-Tactile Internet

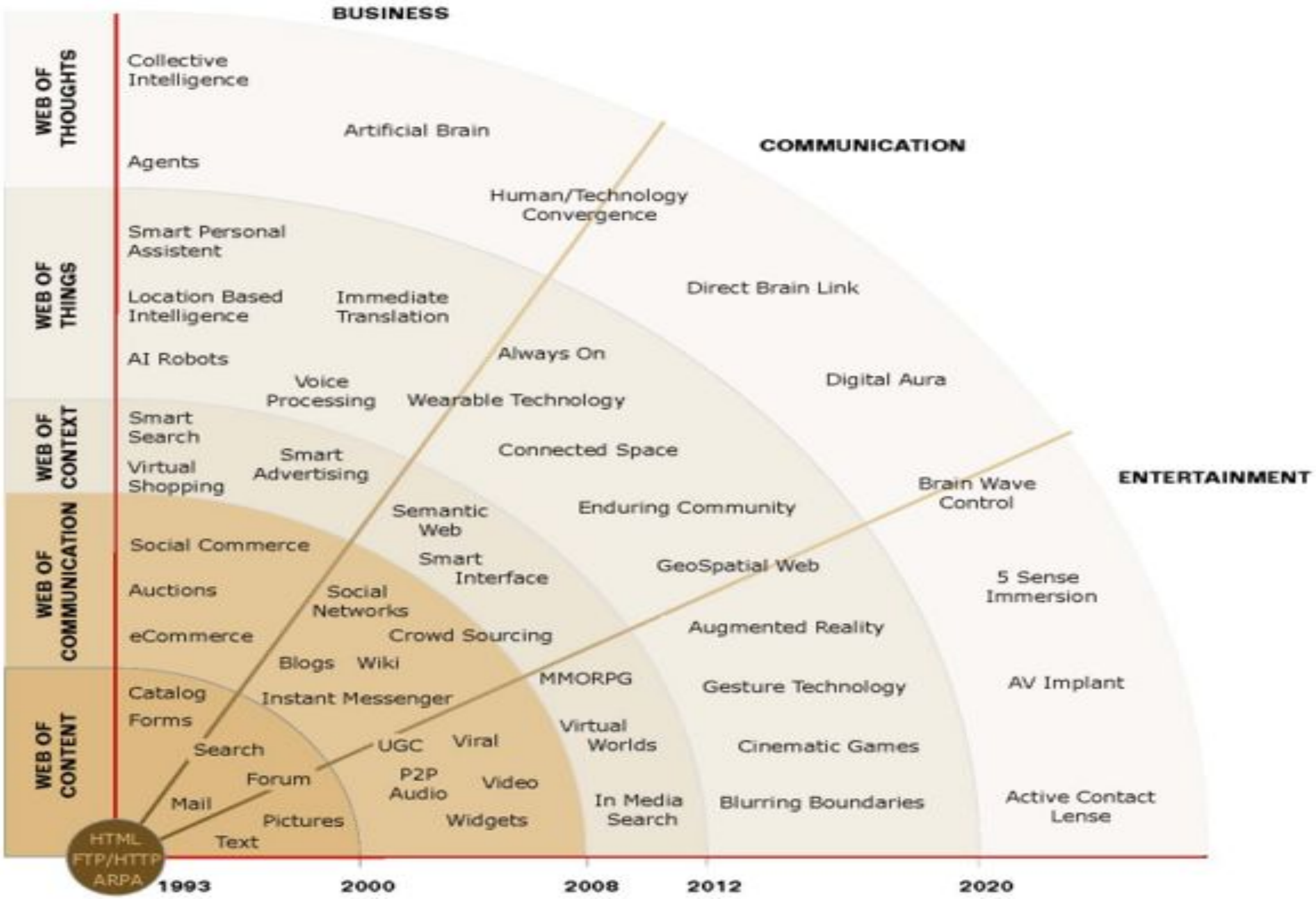
- Next evolution of IoT-
  - Humans and Machines interacting with their environment in real time with haptic interfaces
- Ultra low latency, extreme availability, reliability and security
- Haptics – sense of Touch – the perception and manipulation of objects using Touch and Proprioception (relative positioning of the part’s of one’s body and the strength of effort used in movement)
- 5G-Enabled Tactile Internet

What are the use cases?

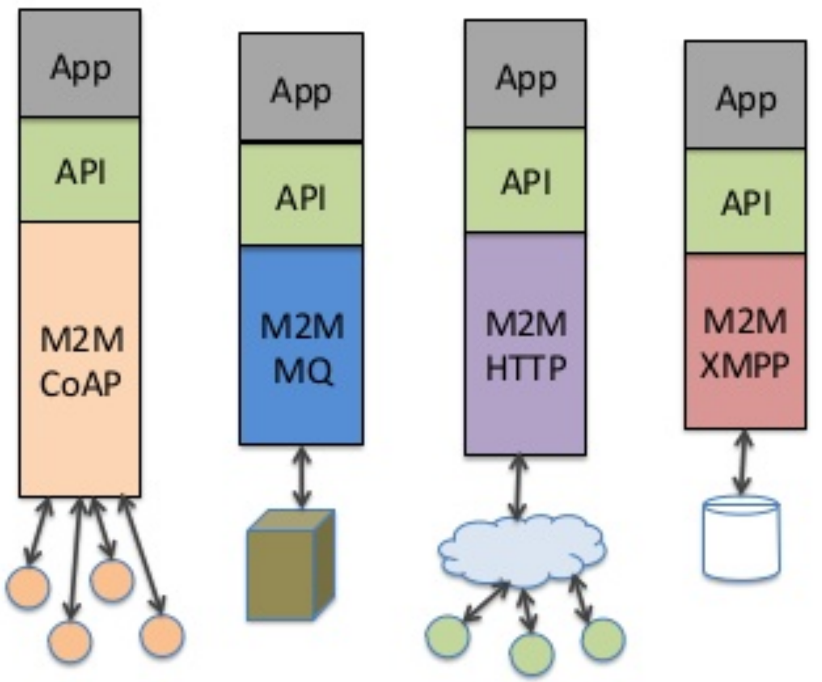
- Tele healthcare :The physician will be able to command a tele-robot at the patient's location, allowing remote physical examination with full AR and haptic feedback
- Robotic exoskeletons will be used for tele-rehabilitation
- Serious Games have wide applications in fields such as healthcare, education, training and simulation.
- Remote learning of skills – the concept of *hands-on* takes a whole new meaning

trendone
NILS MÜLLER COMMUNICATIONS

THE WEB EXPANSION
FROM WEB OF THINGS TO WEB OF THOUGHTS



IoT 1.0 – Things Connected to Apps

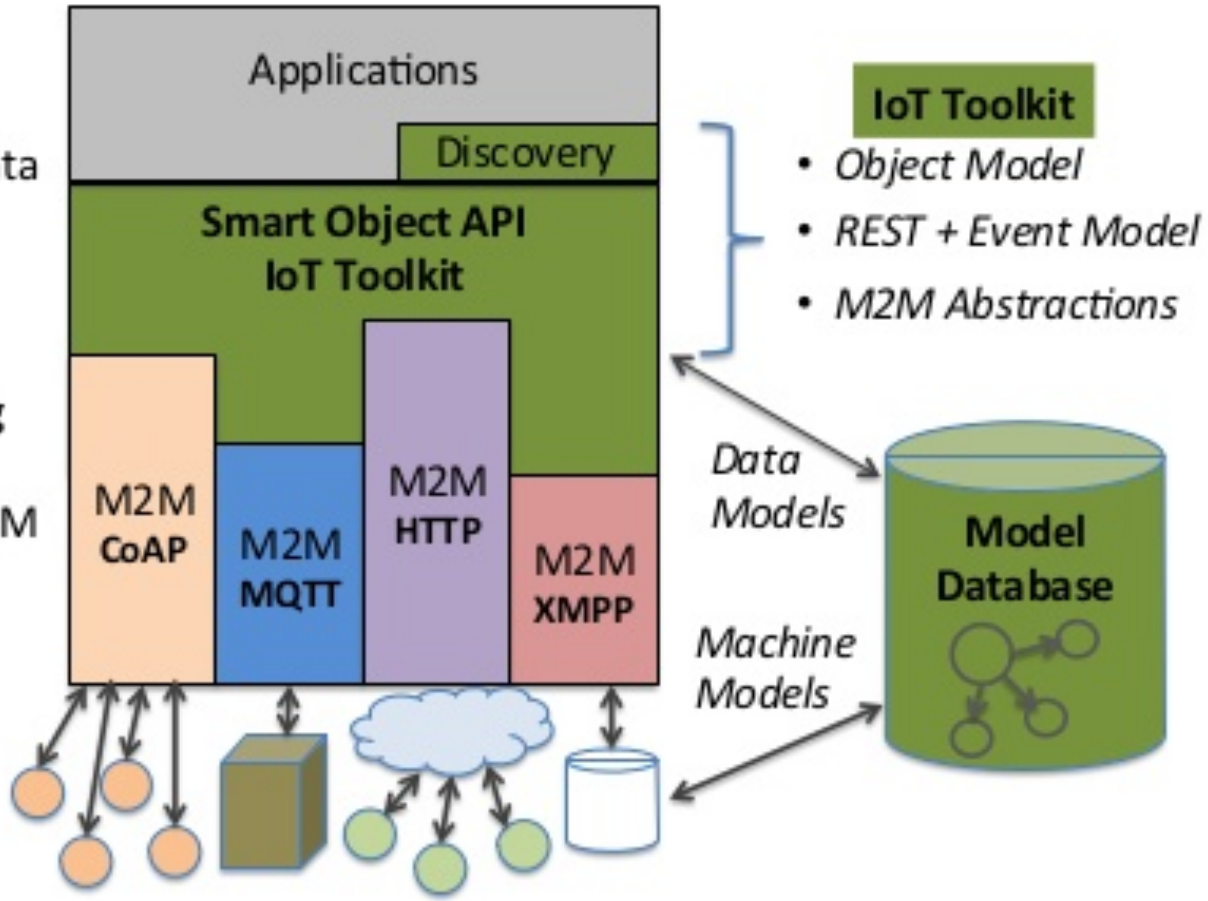


Connected Things, Sensors, Actuators, Data Sources

- App runs on single service – Single Points Of Failure
- Each app written to a custom API
- Diverse M2M is sometimes required but can inhibit interoperability
- Software, User data, and Things are trapped in **Silos**
- Difficult to connect new types of things and deploy new platforms
- Very difficult to share resources or connect across platforms
- Apps are not network-effect enabled

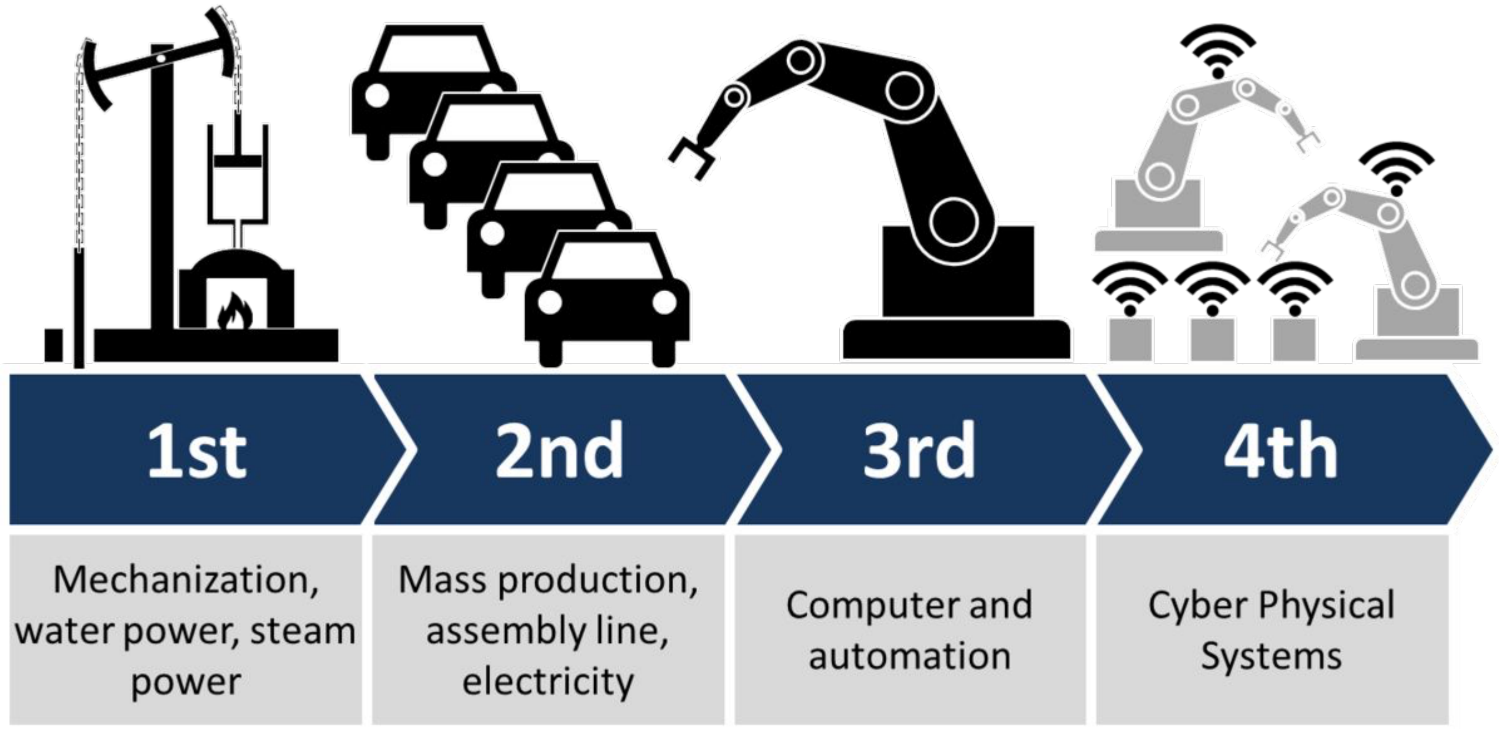
IoT 2.0 – Interoperability

- Easy to deploy new things and applications using data models
- Write once, run anywhere software
- **Any app** to **any thing** via **any M2M**, use-case appropriate M2M
- Network effect enabled



Connected Things, Sensors, Actuators, Data Sources

Industry 1.0 to 4.0



1st

Mechanization, water power, steam power

2nd

Mass production, assembly line, electricity

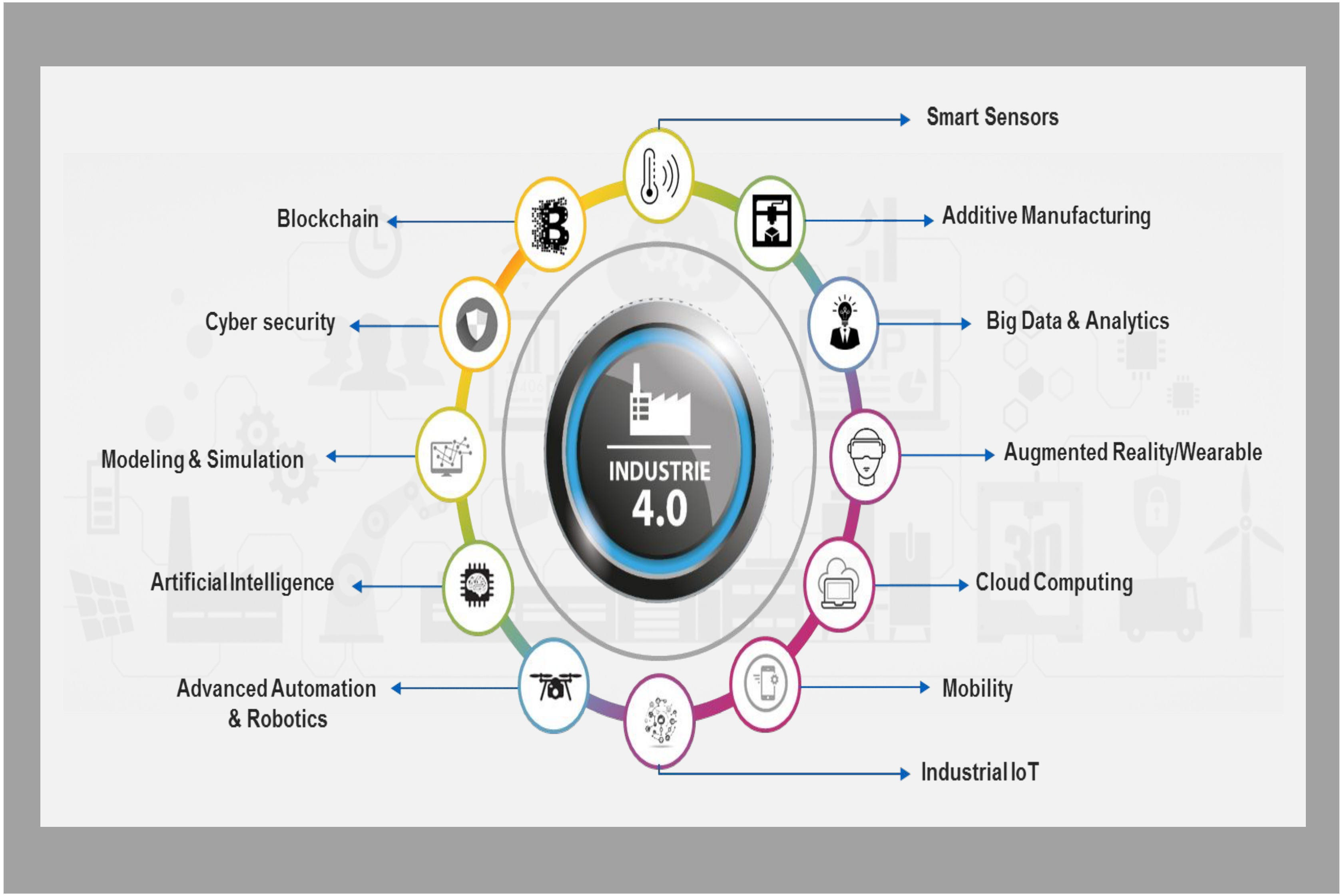
3rd

Computer and automation

4th

Cyber Physical Systems

Industry 4.0 – Key Technology Enablers



Changes in the working environment with adoption of Industry 4.0

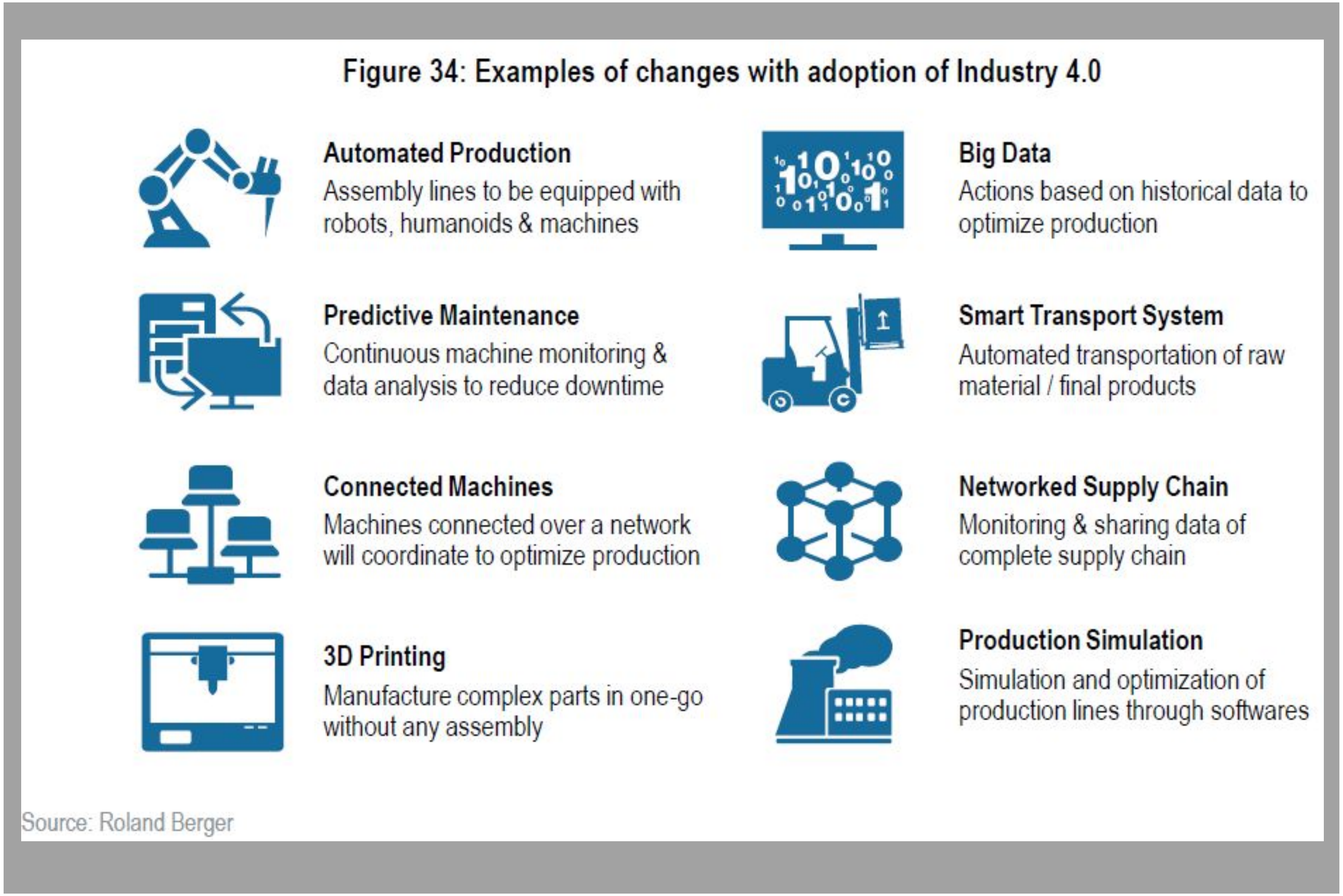


Figure 34: Examples of changes with adoption of Industry 4.0

Automated Production
Assembly lines to be equipped with robots, humanoids & machines

Big Data
Actions based on historical data to optimize production

Predictive Maintenance
Continuous machine monitoring & data analysis to reduce downtime

Smart Transport System
Automated transportation of raw material / final products

Connected Machines
Machines connected over a network will coordinate to optimize production

Networked Supply Chain
Monitoring & sharing data of complete supply chain

3D Printing
Manufacture complex parts in one-go without any assembly

Production Simulation
Simulation and optimization of production lines through softwares

Source: Roland Berger

Enterprise IoT

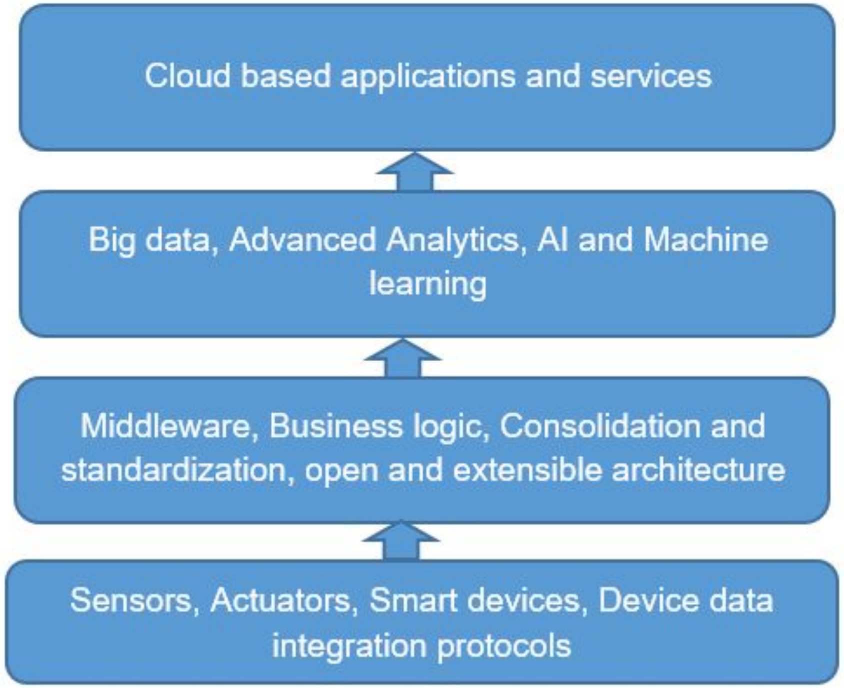
- Consumer IoT Vs Enterprise IoT
- Enterprise IoT – refers to the scope of application of IoT is bigger organizational perspective
- Different components of Enterprise IoT – from hardware layer to higher level applications
- The challenges

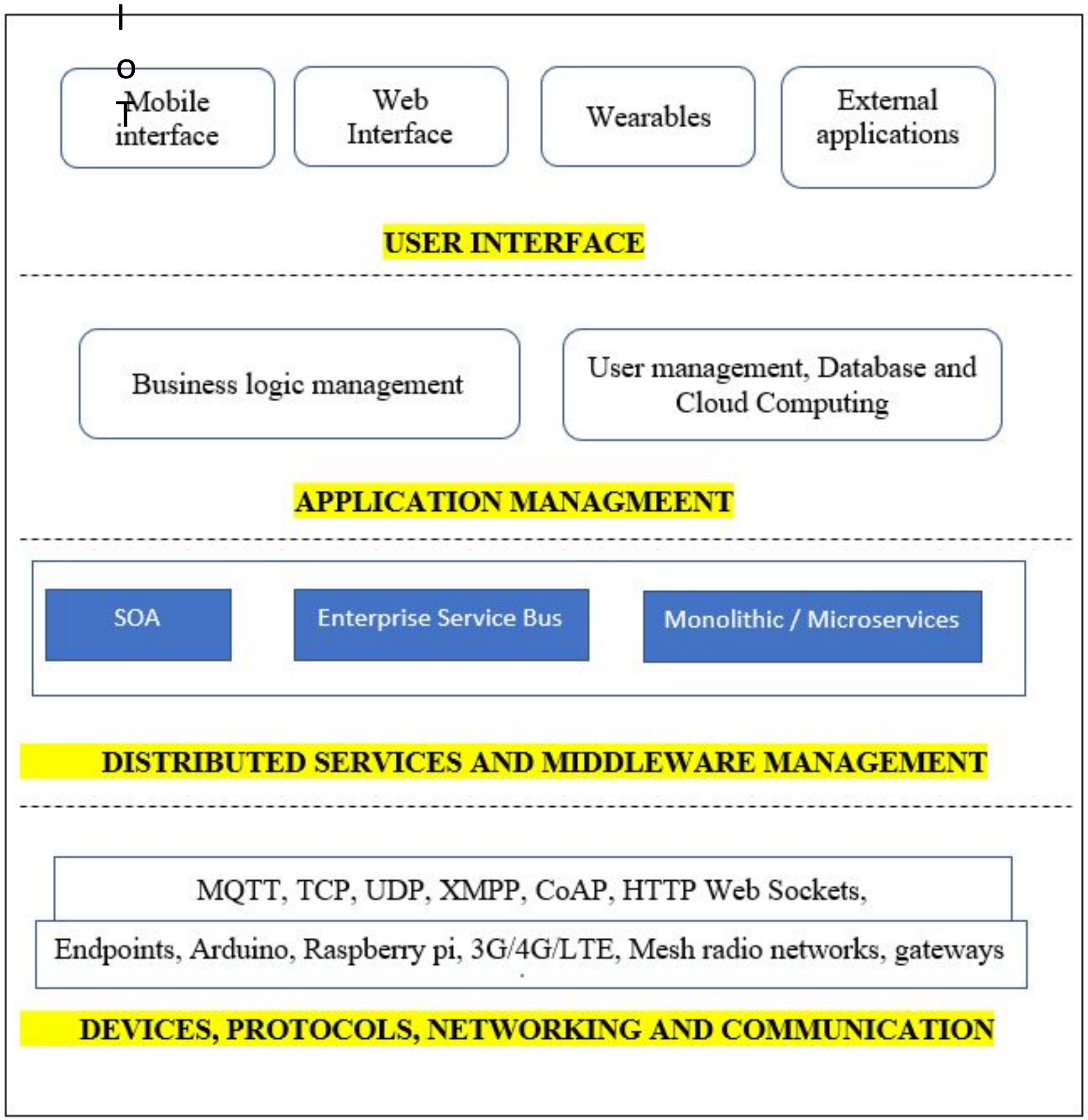
What is Enterprise IoT

What is Enterprise IoT

- Collecting data from various assets and resources
- Analysis of data in the backend

Integrated view of Enterprise IoT Application


Cloud based applications and services
Big data, Advanced Analytics, AI and Machine learning
Middleware, Business logic, Consolidation and standardization, open and extensible architecture
Sensors, Actuators, Smart devices, Device data integration protocols



I

o

T

Mobile interface

Web Interface

Wearables

External applications

USER INTERFACE

Business logic management

User management, Database and Cloud Computing

APPLICATION MANAGMEENT

SOA

Enterprise Service Bus

Monolithic / Microservices

DISTRIBUTED SERVICES AND MIDDLEWARE MANAGEMENT

MQTT, TCP, UDP, XMPP, CoAP, HTTP Web Sockets,

Endpoints, Arduino, Raspberry pi, 3G/4G/LTE, Mesh radio networks, gateways

DEVICES, PROTOCOLS, NETWORKING AND COMMUNICATION

IoT Middleware

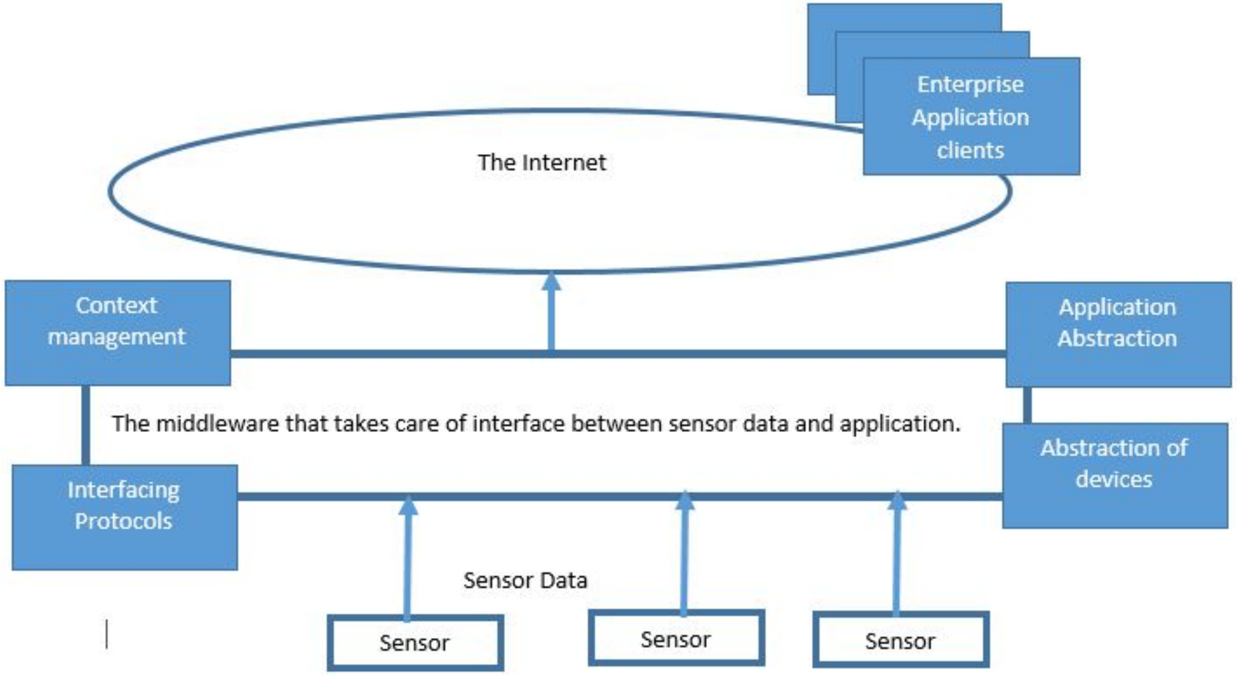
- functions as an interface between components of the IoT, making communication possible among disparate elements

- deals with the structure, format and encoding of the information that is being exchanged between different layers, devices and sensors.

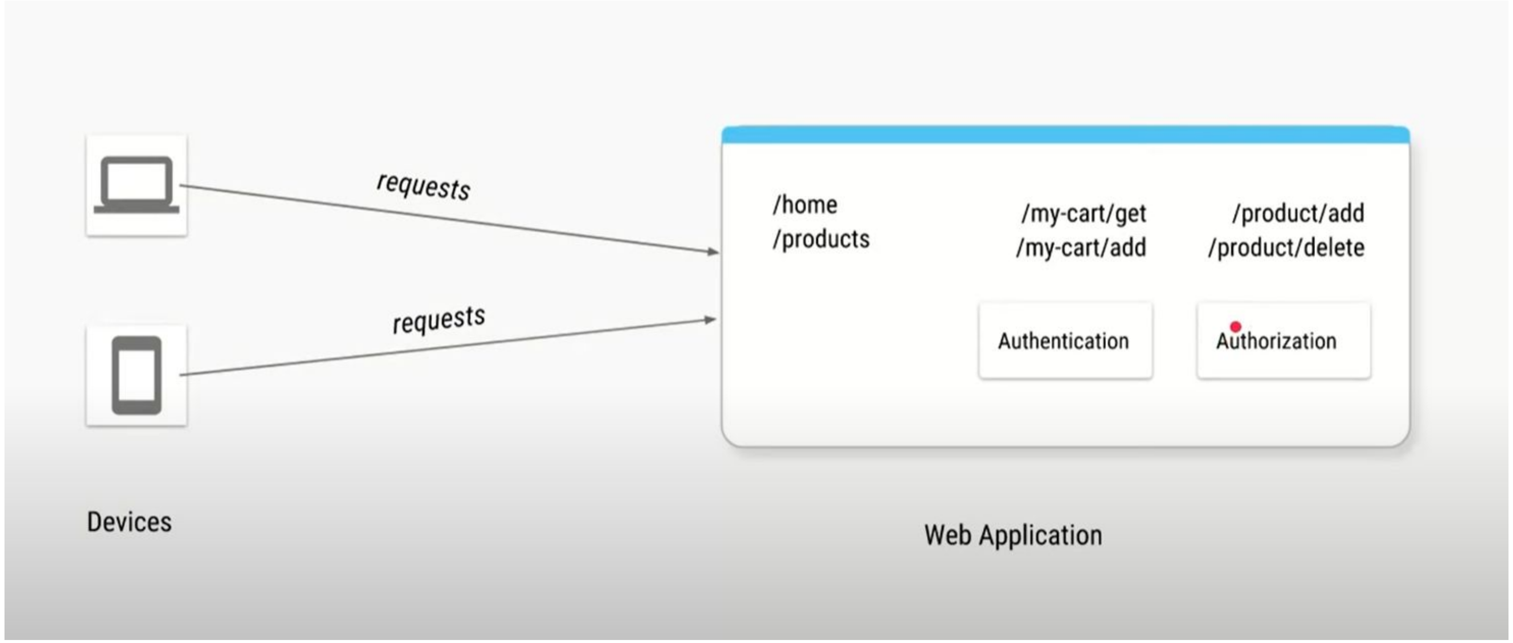
- act as a connecting bridge and common standard amongst the diversity of devices, sensors, OS and applications that will make up the IoT ecosystem architecture

- Integrates the data before it is being analyzed

Middleware functional components for Enterprise IoT



API


requests
requests
/home
/products
/my-cart/get
/my-cart/add
/product/add
/product/delete
Authentication
Authorization
Devices
Web Application

Why are APIs linked with IoT ?

- An API is a set of routines, protocols, and tools for building software applications;
- specifies how software components should interact.
- APIs tightly linked with IoT because they allow you to securely expose connected devices to customers,, and other applications
- APIs connect important “things,” like cars, medical devices, smart grids, and thermostats, to your ecosystem, it’s critical to deploy API management that is flexible, scalable, and secure.

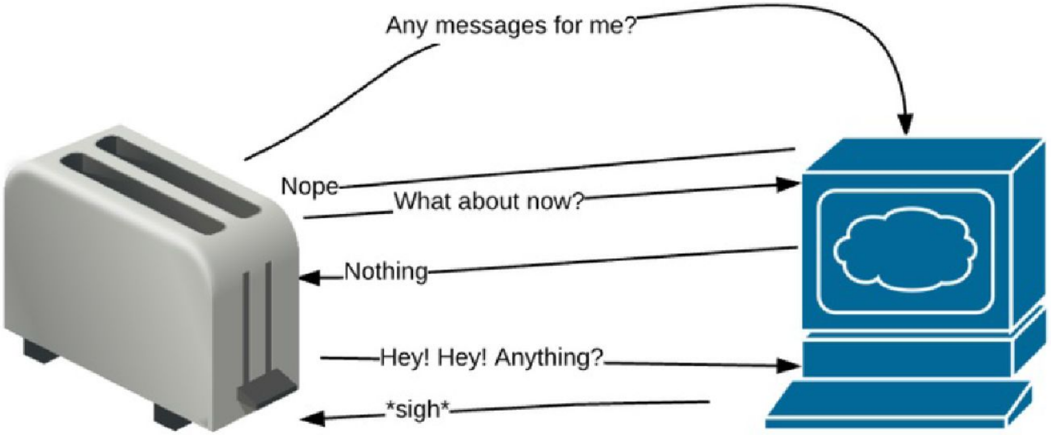
Current state of IoT -- Unstructured, Semi structured and Structured data – using Proprietary APIs

The need – holistic information design – things embedded with web technologies

IoT Communication Protocols

HTTP

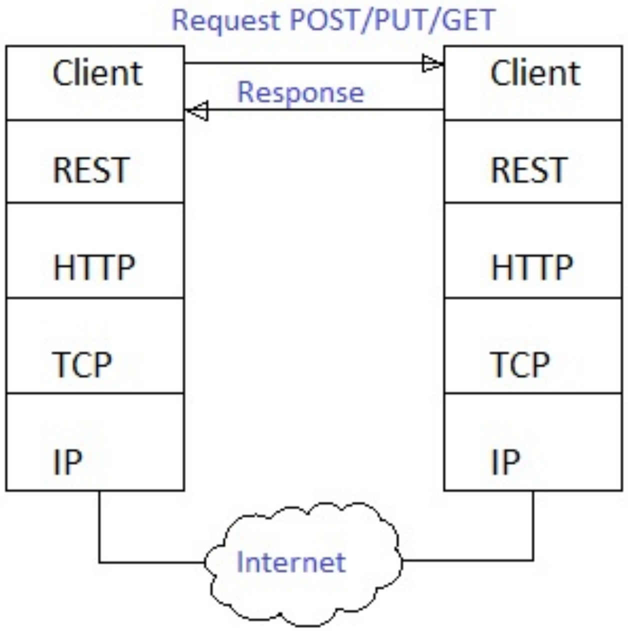
- HTTP is: asynchronous protocol, this means the client waits for the server to send the data.
- a one way street, only the client makes the request. one to one protocol, one client request at a time.
- data heavy, filled with headers and rules.


Any messages for me?
Nope
What about now?
Nothing
Hey! Hey! Anything?
sigh

RESTful Model

- Representational State Transfer
  concept, not a protocol.
- RESTful APIs are widely used in the modern web, and data transfer usually takes place using JSON or XML over HTTP.
- REST is a stateless concept- does not maintain a constantly open connection
- use methods of the HTTP -

  GET, POST, PUT,DELETE



REST PROTOCOL

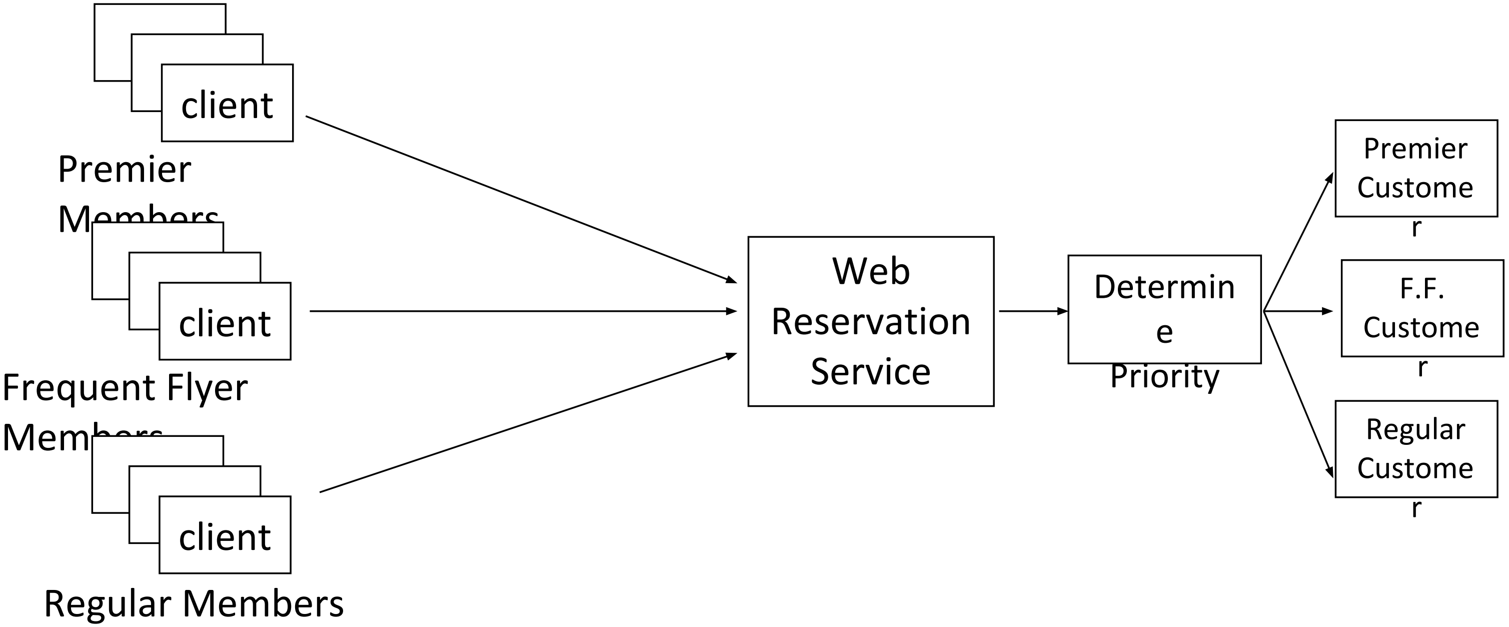
- All things should be exposing their services through a RESTful API

- If things offer RESTful APIs over HTTP, they get a URL and become seamlessly integrated to the World Wide Web

- enables developers to program devices just like they work with existing backend systems when building mobile apps

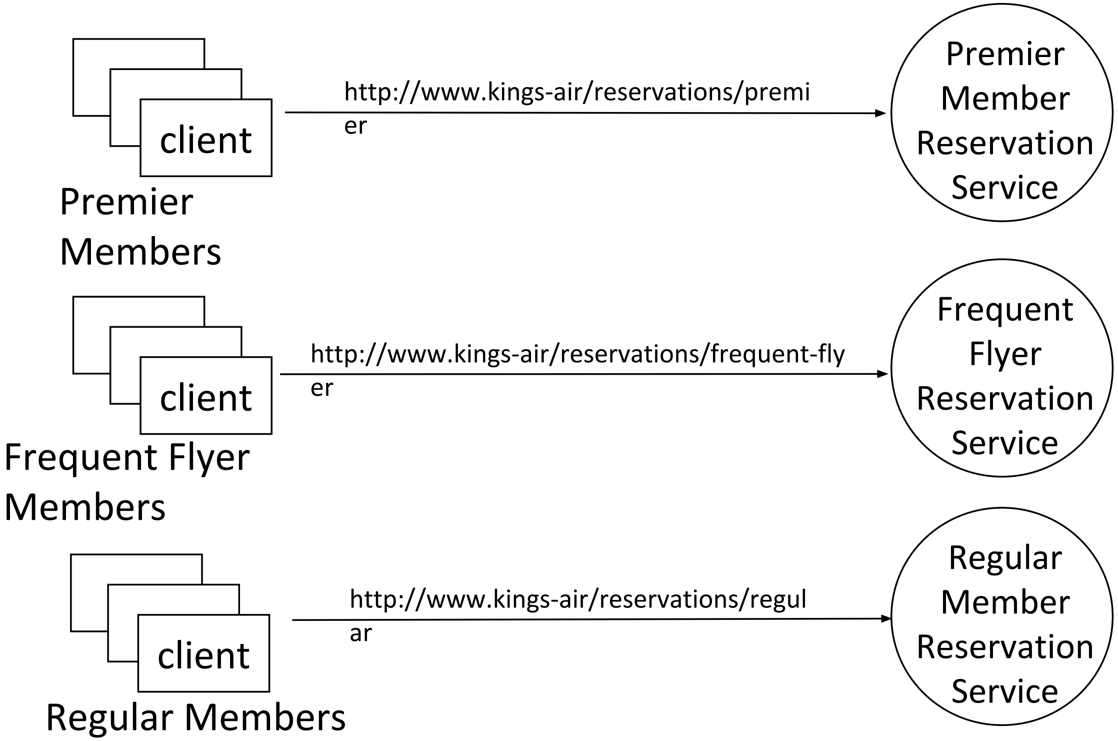
Airline Reservation Example

The airline provides a single URL. The Web service is responsible for examining incoming client requests to determine their priority and process them accordingly.



URLs are Cheap! Use Them! – The REST Design Pattern

The airline provides several URLs - one URL for premier members, a different URL for frequent flyers, and still another for regular customers.



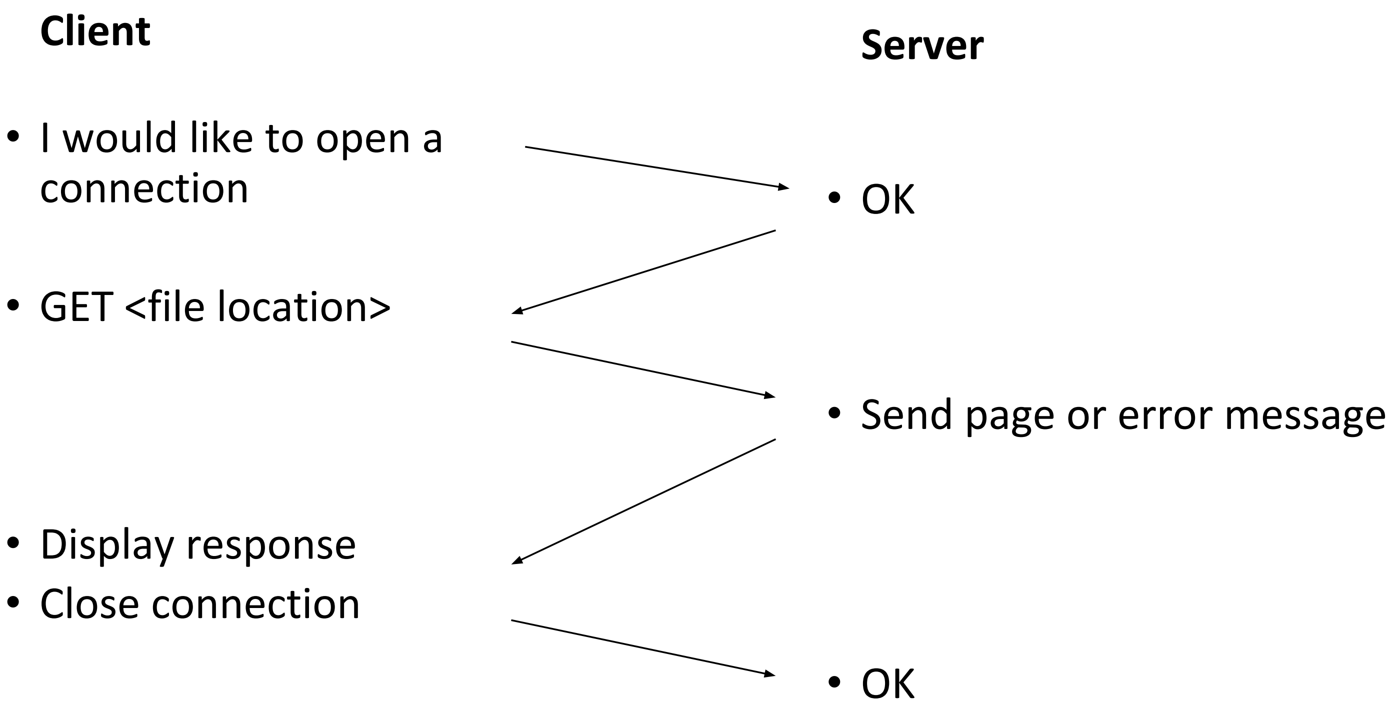
client
Premier Members
http://www.kings-air/reservations/premier
Premier Member Reservation Service
client
Frequent Flyer Members
http://www.kings-air/reservations/frequent-flyer
Frequent Flyer Reservation Service
client
Regular Members
http://www.kings-air/reservations/regular
Regular Member Reservation Service

REST

- Request / Response model – every time you need to do something
  - e.g turn the light bulb on using PUT request
- Tons of requests, lots of data, battery drain, slow response etc

MQTT

An HTTP conversation



Client

- I would like to open a connection
- GET <file location>
- Display response
- Close connection

Server

- OK
- Send page or error message
- OK

HTTP is the set of rules governing the format and content of the conversation between a Web client and server

What is MQTT ?

- MQTT = MQ Telemetry Transport
- request-response nature of HTTP does not match the event-driven nature of applications- MQTT
- Data sent using JSON format

What is MQTT ?

- Lightweight messaging protocol designed for sensors and devices with
  - Flaky network connectivity
  - Low computing power
  - Connections where bandwidth is at a premium
- Protocol specification is open source
- MQTT is nearly 10 years old
  - Mature and evolving

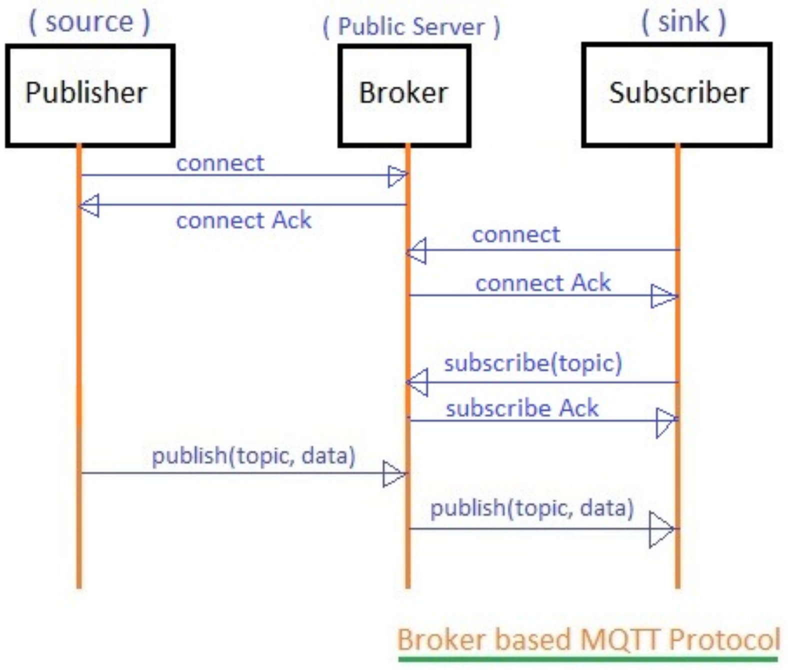
Features of MQTT

- Publish and subscribe to topics

- 3 qualities of service
  - 0 Best effort to deliver a message
  - 1 Deliver at least once
  - 2 Deliver exactly once

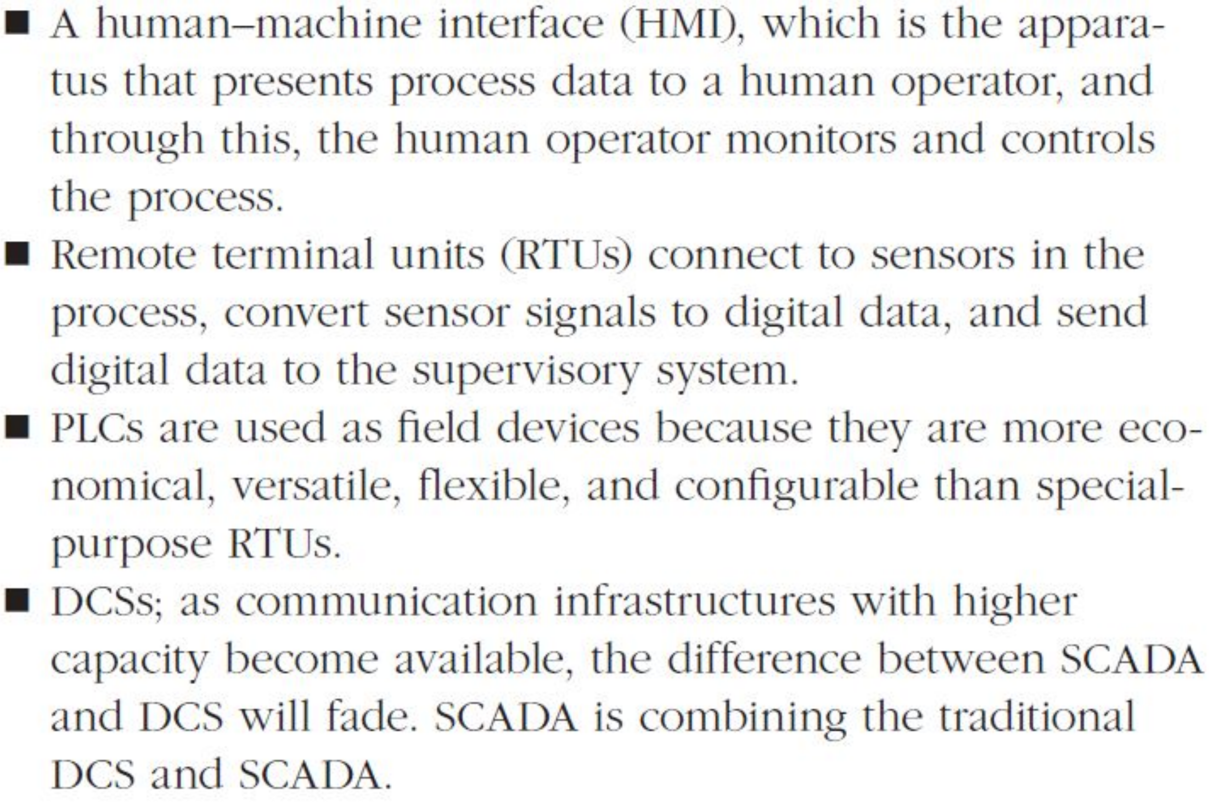
- Minimal transport overhead to reduce network traffic
  - As little as 2 bytes

MQTT : MQ Telemetry Transport



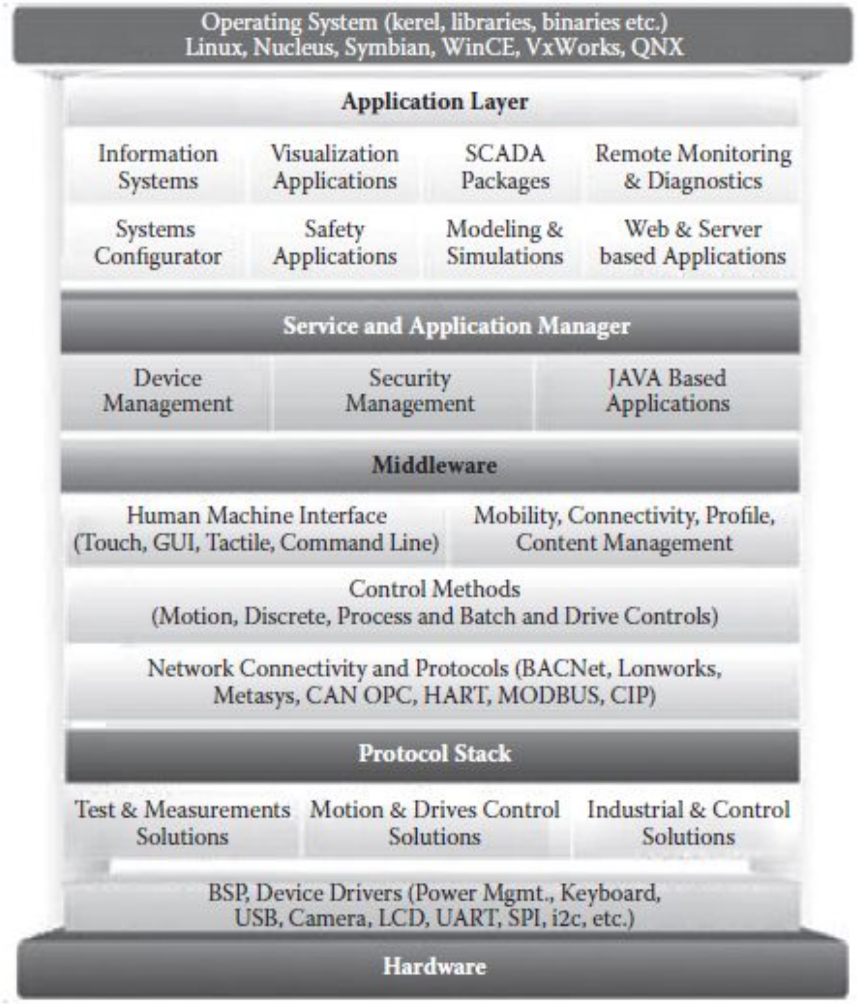
Features	**MQTT**	**HTTP**
Full Form	Message Queue Telemetry Transport	Hyper Text Transfer Protocol
Design Methodology	The protocol is data centric.	The protocol is document centric.
Architecture	It has publish/subscribe architecture. Here devices can publish any topics and can also subscribe for any topics for any updates.	It has request/response architecture.
Complexity	simple	more complex
Data security	YES	NO, hence HTTPS is used to provide data security.
Upper layer protocol	It runs over TCP.	It runs over TCP
message size	small, it is binary with 2Byte header.	Large, it is in ASCII format.
Service levels	3	1
Libraries	30KB C, 100KB Java	Large
Port number	1883	80 or 8080
Data distribution	1 to 0/1/N	one to one only

SCADA components



- A human–machine interface (HMI), which is the apparatus that presents process data to a human operator, and through this, the human operator monitors and controls the process.
- Remote terminal units (RTUs) connect to sensors in the process, convert sensor signals to digital data, and send digital data to the supervisory system.
- PLCs are used as field devices because they are more economical, versatile, flexible, and configurable than special-purpose RTUs.
- DCSs; as communication infrastructures with higher capacity become available, the difference between SCADA and DCS will fade. SCADA is combining the traditional DCS and SCADA.

Middleware Based SCADA System



Operating System (kerel, libraries, binaries etc.)
Linux, Nucleus, Symbian, WinCE, VxWorks, QNX

Application Layer

Information Systems

Visualization Applications

SCADA Packages

Remote Monitoring & Diagnostics

Systems Configurator

Safety Applications

Modeling & Simulations

Web & Server based Applications

Service and Application Manager

Device Management

Security Management

JAVA Based Applications

Middleware

Human Machine Interface (Touch, GUI, Tactile, Command Line)

Mobility, Connectivity, Profile, Content Management

Control Methods (Motion, Discrete, Process and Batch and Drive Controls)

Network Connectivity and Protocols (BACNet, Lonworks, Metasys, CAN OPC, HART, MODBUS, CIP)

Protocol Stack

Test & Measurements Solutions

Motion & Drives Control Solutions

Industrial & Control Solutions

BSP, Device Drivers (Power Mgmt., Keyboard, USB, Camera, LCD, UART, SPI, i2c, etc.)

Hardware

Industrial Automation Stack

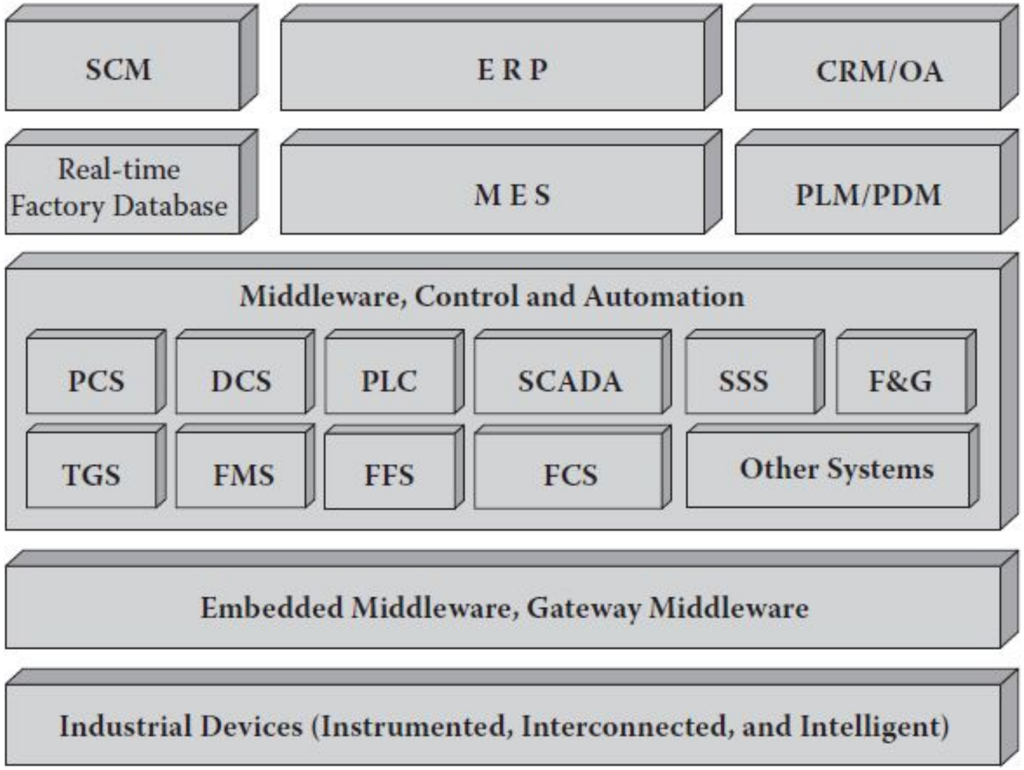

SCM
E R P
CRM/OA
Real-time Factory Database
M E S
PLM/PDM
Middleware, Control and Automation
PCS
DCS
PLC
SCADA
SSS
F&G
TGS
FMS
FFS
FCS
Other Systems
Embedded Middleware, Gateway Middleware
Industrial Devices (Instrumented, Interconnected, and Intelligent)

Figure 4.8 The industrial automation stack. FCS = Field Bus Control System; DCS = Distributed Control System; PLC = Programmable Logic Controller; SCADA = Supervisory Control and Data Acquisition; TMS = Tank Management System; FMS = Flow Metering System; F&G = Fire and Gas; SSS = Safety Shutdown System; FFS = Firefighting System; MES = Manufacturing Execution System; ERP = Enterprise Resource Planning.