SUP'COM
Higher School of Communication of Tunis

INDP2: IoT & Cloud

ARCHITECTING IoT DEVICE

Part I: IoT Solution Architecture & Components

A. Ghazel, O. Ben Rhouma, C. Rebai, M. Ben Romdhane, F. Tlili

adel.ghazel@supcom.tn, ons.benrhouma@supcom.tn, chiheb.rebai@supcom.tn, manel.benromdhane@supcom.tn, fethi.tlili@supcom.tn

Course Objectives

Target Knowledge

- Be aware about IoT concept motivation, application ecosystem and architecture models and standards
- Learn specific requirements for IoT connectivity and major wireless network technologies used for IoT solutions deployment
- Understand design challenges for an IoT Device in terms of sensing and acting techniques and technologies on external environment, embedded processing technologies and energy efficiency for massive and critical IoT applications

Target Skills

- Understand IoT architecture models for Application oriented IoT Solution design
- Master M2M wireless connectivity standards specifications and operational modes
- Learn basic techniques and technologies to build IoT Device smart sensors and actuators
- Master specific digital processing units and embedded software implementation design
- Understand the challenging requirement for IoT Device efficiency and master state of the art harvesting techniques and technologies

Teaching activities

Course Organization

Teaching element	Hours	Course	Ex.	Lab.
IoT Device Lectures	15	15		
IoT Device Exercices	6			6

Evaluation modes

Test type	Number	Weighting
Exam	1	
Homework	1	

Course outlines

Part I	**IoT Solution Architecture & Components**	*Adel Ghazel*
Part II	**IoT Device Connectivity**	*Ons Ben Rhouma*
Part III	**IoT Device Power efficient design**	*Chiheb Rebai*
Part IV	**IoT Device smart sensors**	*Manel Ben Romdhane*
Part V	**IoT Device digital processing**	*Fethi Tlili*
Part VI	**Exercises**	

SUP'COM

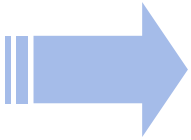
Part I

IoT Solution Architecture & Components

Outlines

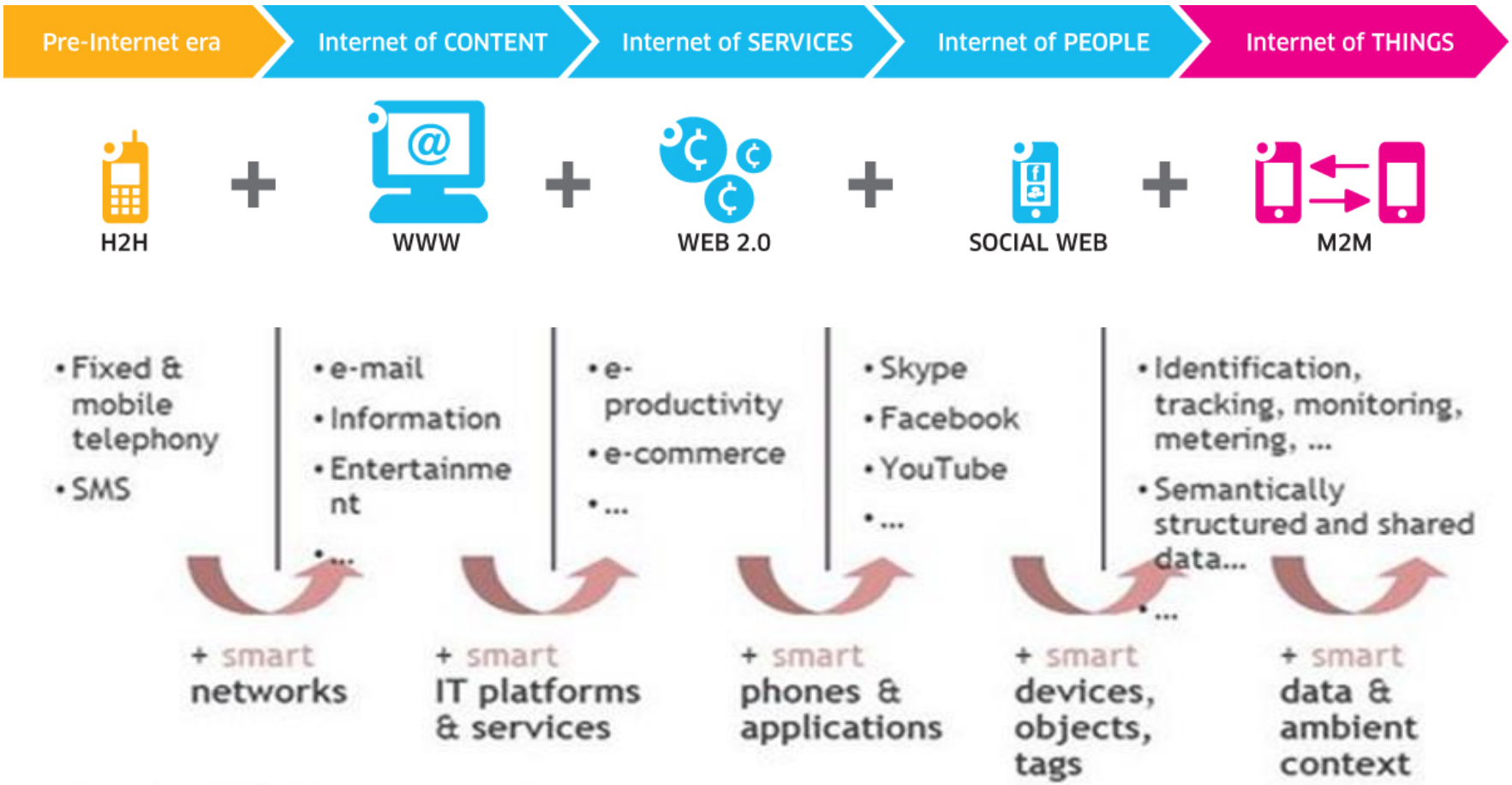
1. IoT Concept Definition & Motivation
2. IoT Applications & Ecosystem
3. IoT Architecture Models & Standards
4. IoT Solution Components & Technologies

Part I-1



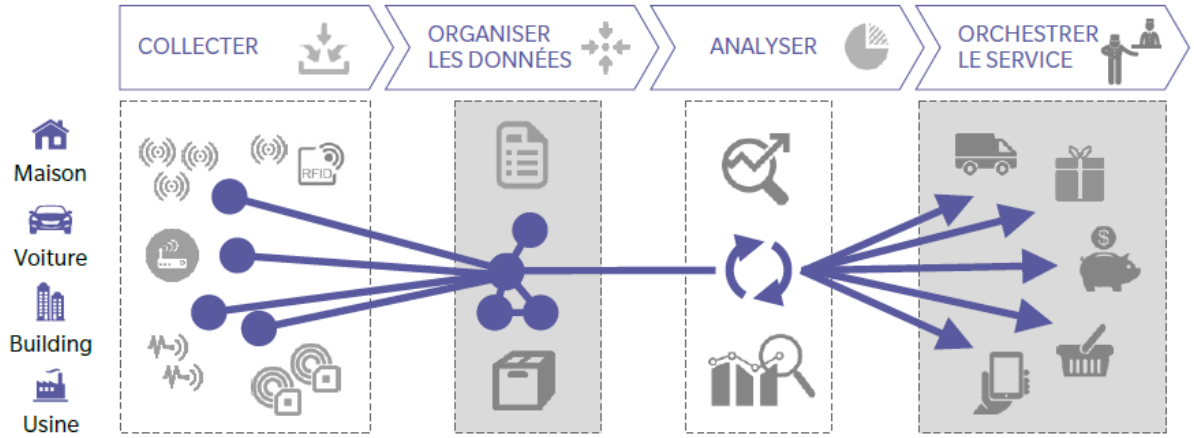
IoT Concept Definition & Motivations

Internet Evolution Towards IoT



IoT Definition by the ITU

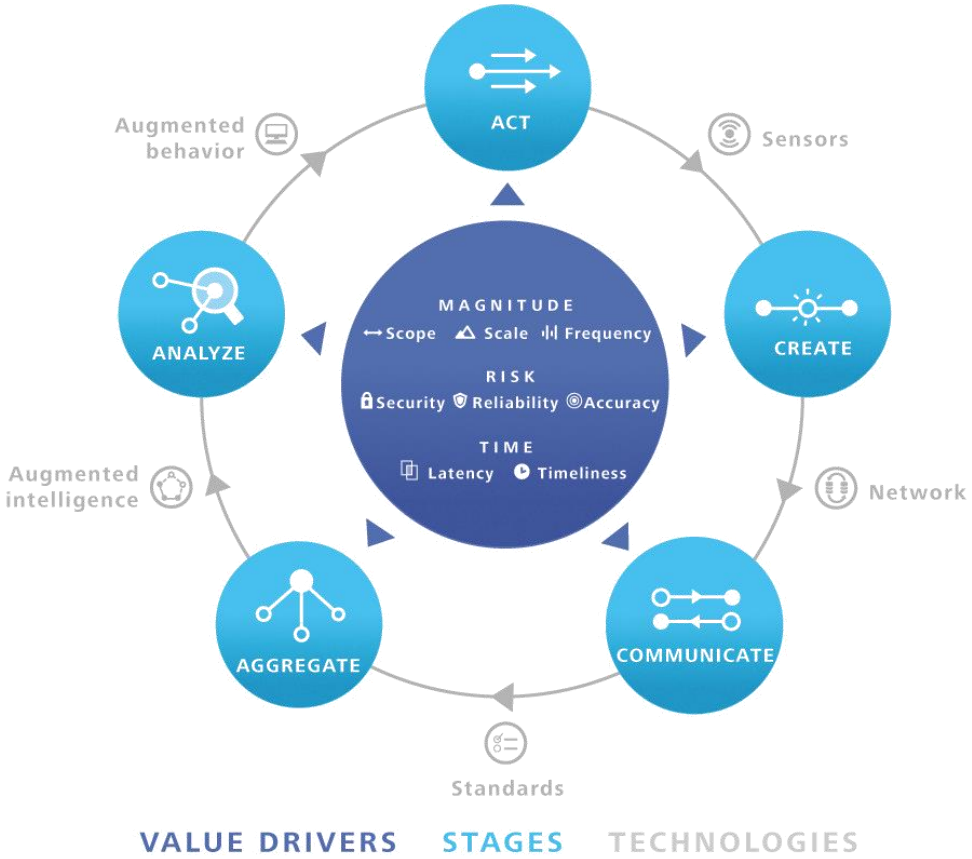
The International Telecommunication Union (ITU) defines the Internet of Things as "***a global infrastructure for the information society, which provides advanced services by interconnecting objects (physical or virtual) using existing interoperable ICT or evolving*** ".



- **Physical Objects:** having on-board sensor, intelligence and connectivity technologies, allowing them to communicate with other objects;
- **Electronic communications networks:** used to transport data from objects;
- **Cloud computing**, more or less distributed, provides the tools for storing, correlating and analyzing this data (decision-making processes capable of feedback on physical objects).

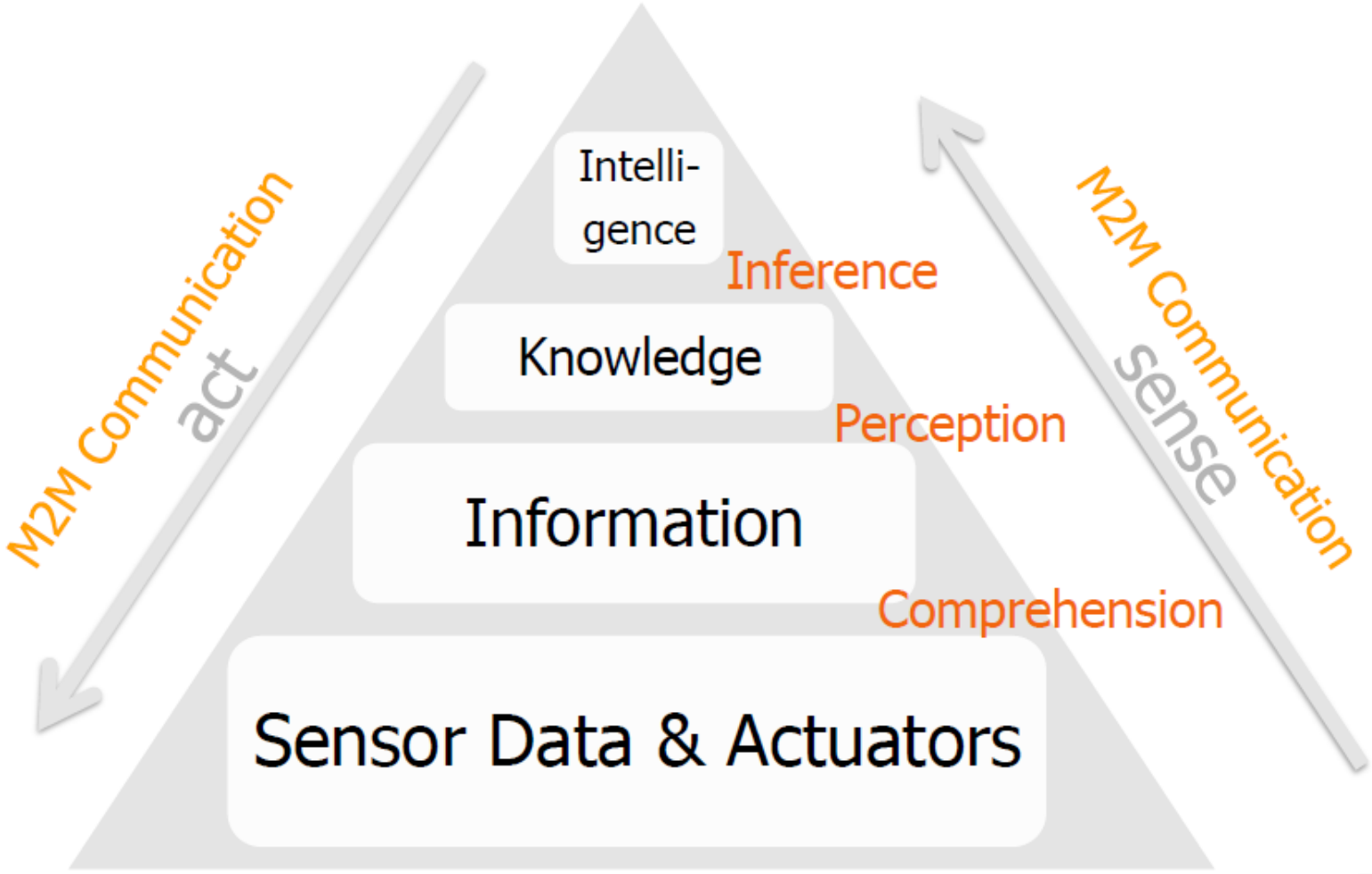
IoT Definition by the IEEE

The **Internet of Things** (IoT) is a framework in which all objects have a representation and a presence on the Internet. More specifically, Internet of Things aims to offer new applications and services linking the physical and virtual worlds, in which **Machine-to-Machine** communications represent the basic communication that allows the interactions between objects and applications in the cloud.



Source: Deloitte analysis.

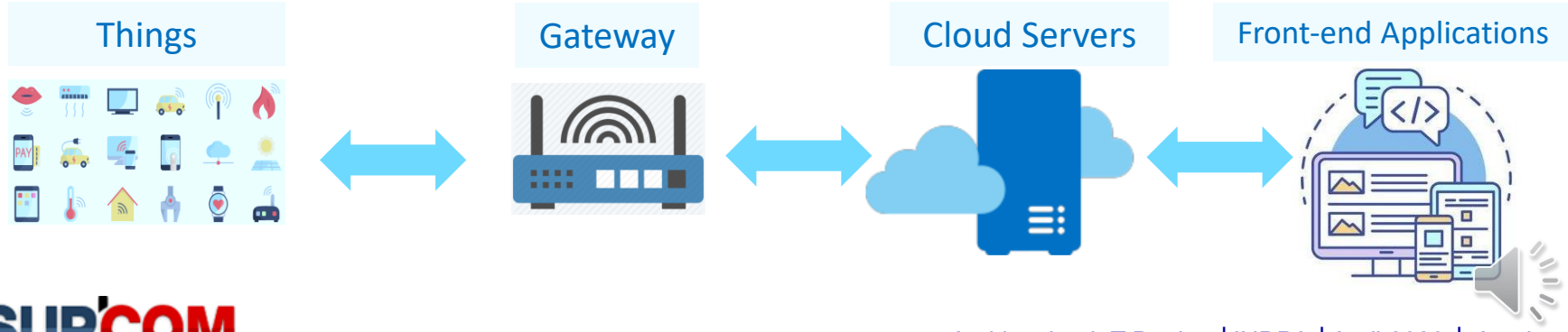
IoT Processing Hierarchy



Evolution from M2M to IoT Solution Architecture

Traditional M2M Solution Architecture

IoT Solution Architecture

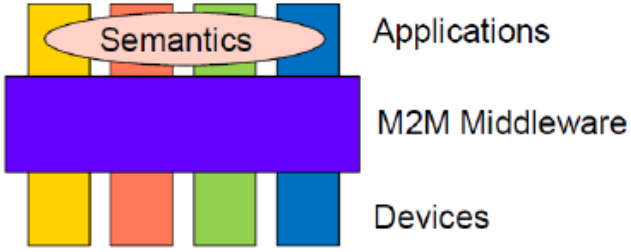


M2M vs IoT

M2M

Low level information

Measurement: temperature, level, image
Control: on/off, speed control,…



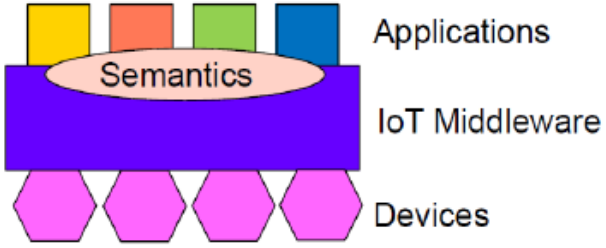
- Keep strong/fixed associations between applications and devices
- Treat information passed between devices and applications as black box, middleware is not aware of what is being communicated

IoT

High level information

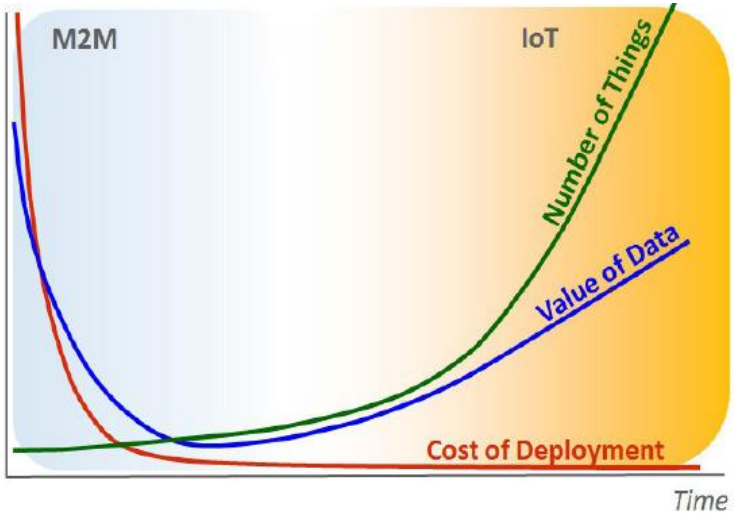
Car, house, people modelled as **Things** having **Attributes**

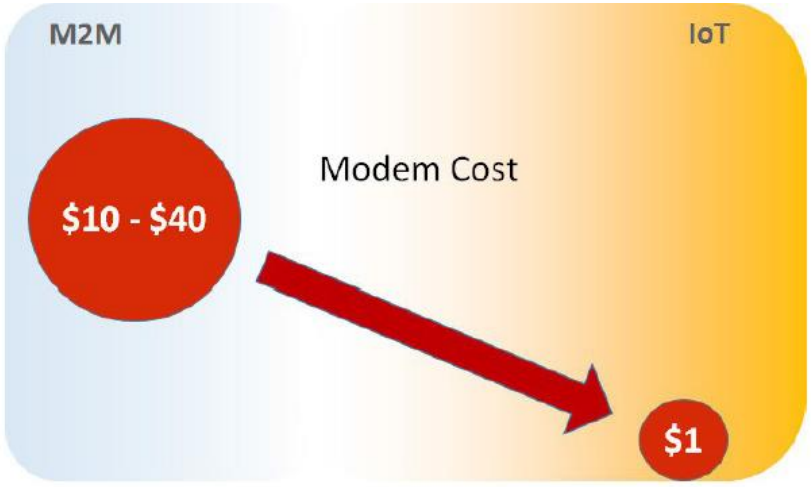
E.g. *Thing people has Attribute blood pressure*

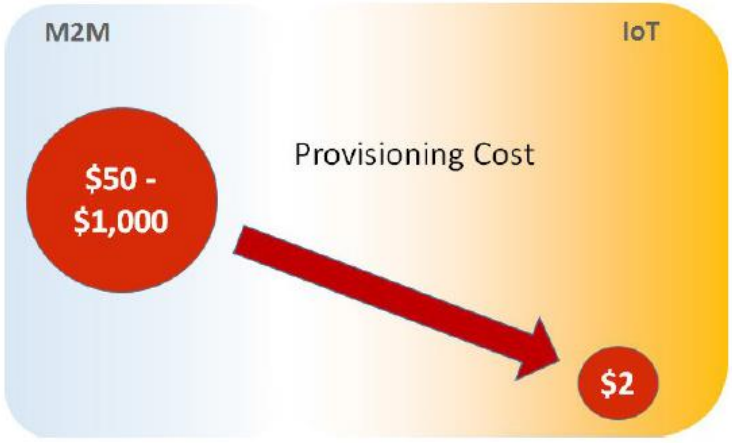


- Things are core concepts, middleware is aware of Things
- Applications interact on Thing---level
- Requires semantic descriptions of Things and associations between Things and devices / higher--level sources
- Requires functionalities for discovery and processing of information / actuation

IoT Economic Promotion Vectors







Source : nickhurn.com

IoT Market Segment Categories

Massive IoT Connectivity

- *Requirements:* high connection volumes, low cost, low power consumption and limited data traffic.
- *Applications:* smart buildings, transport logistics, fleet management, smart meters, agriculture.
- *Connectivity:* thanks to capillary networks that use ubiquity, security and management of wireless and cellular networks

Critical IoT Connectivity

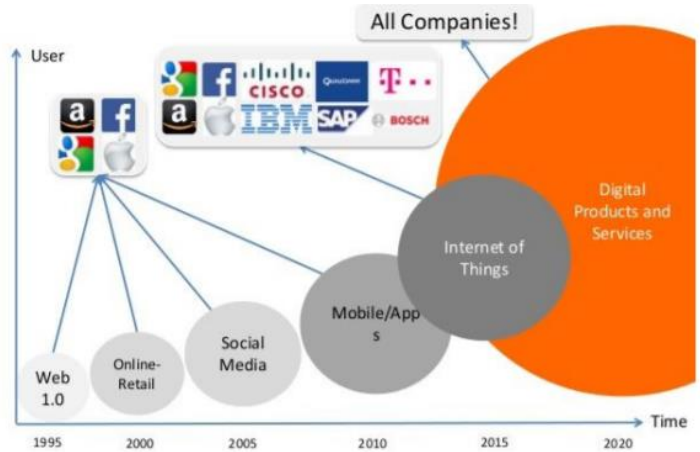
- *Requirements:* ultra-reliability and availability, with very low latency.
- *Applications:* road safety, autonomous cars, industrial applications, remote manufacturing and healthcare, including remote surgery..
- *Connectivity:* the decline in modem costs is making devices connected to LTE increasingly viable, which allows new applications with very low latency. The advanced functionality in 5G capabilities, is expected to expand the range of addressable applications for critical IoT deployments.

Key Challenges in Deploying IoT

- Interaction and communication between heterogeneous objects raising questions of interoperability, energy consumption, processing and transformation of their information.
- Scalable and flexible standard architecture that eliminates or minimizes dependence on target data processing technology, a specific programming language, and communication technology.
- Some objects in critical environments and dependent on remote intelligence require a permanent and reliable connection.
- The penetration of IoT depends on the materialization of demand through adapted services, and on the approval of "refractory" to the new technologies for which these services must materialize through an established framework.
- Upstream joint thinking between the actors involved in order to anticipate economic and social repercussions through the implementation of regulations concerning security, safety of installations and data, respect for privacy, exposure to electromagnetic fields , environmental protection, commercial efficiency,…

IoT Development Objectives

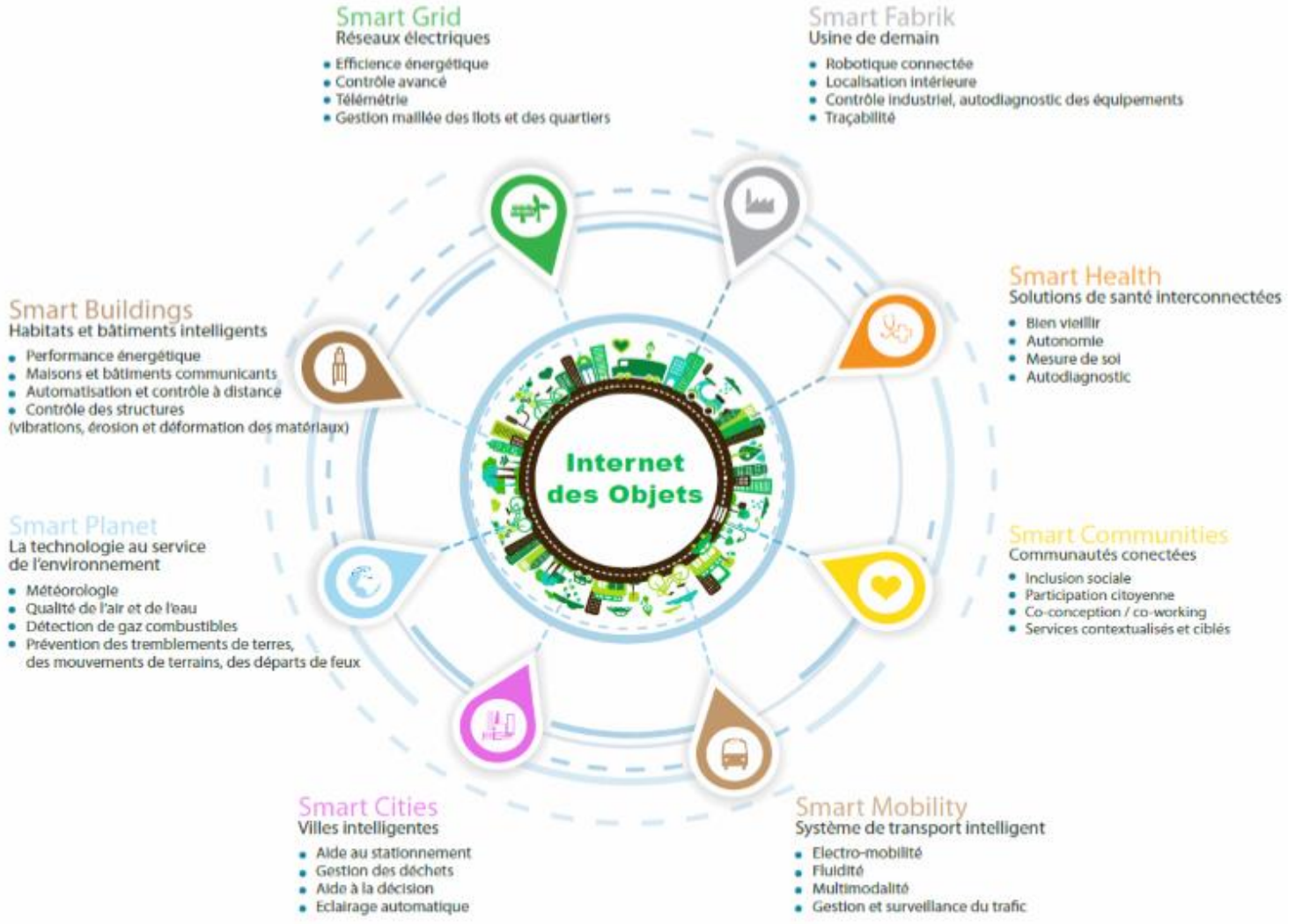
- **Architecture**: Development and improvement of structural reference frameworks for:
  - the physical and logical arrangement of hardware and computer components
  - Object identification, virtualization and decentralization
  - Ensure interoperability between application sectors
- **Security and trust**: development of mechanisms and frameworks ensuring that all users have confidence in applications and enabling them to maintain control of their data throughout the information life cycle.
- **Software platforms**: support for the processing and analysis of data flows from heterogeneous sensors, objects and information systems.
- **Interfaces**: integration of multimodal approaches to enrich human-machine interactions in order to change both the user experience and manage the density of information.
- **Intelligent sensors**: integration of embedded intelligence into networks and energy recovery devices.



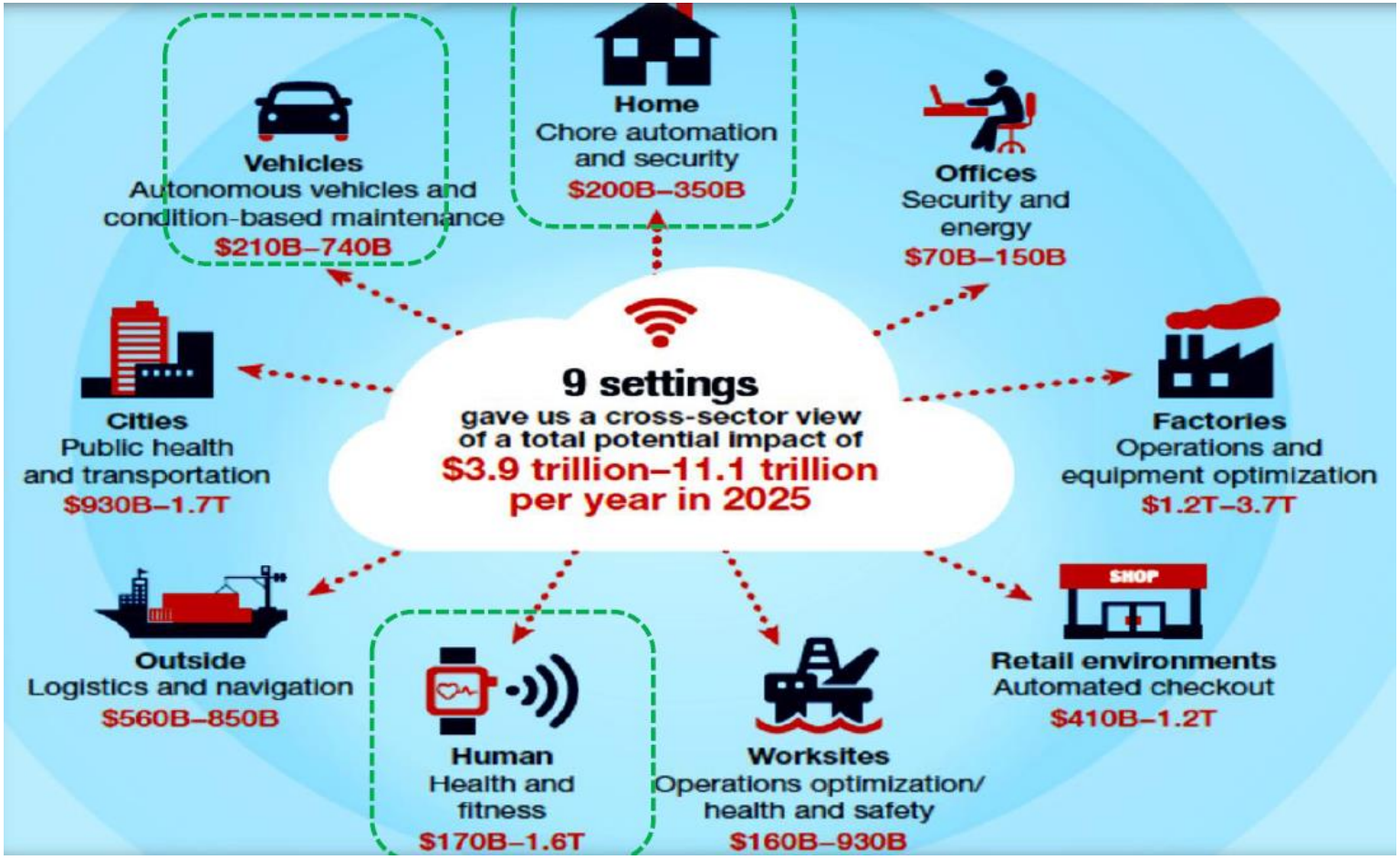
Part I-2

IoT Applications & Ecosystem

IoT Applications Perspectives



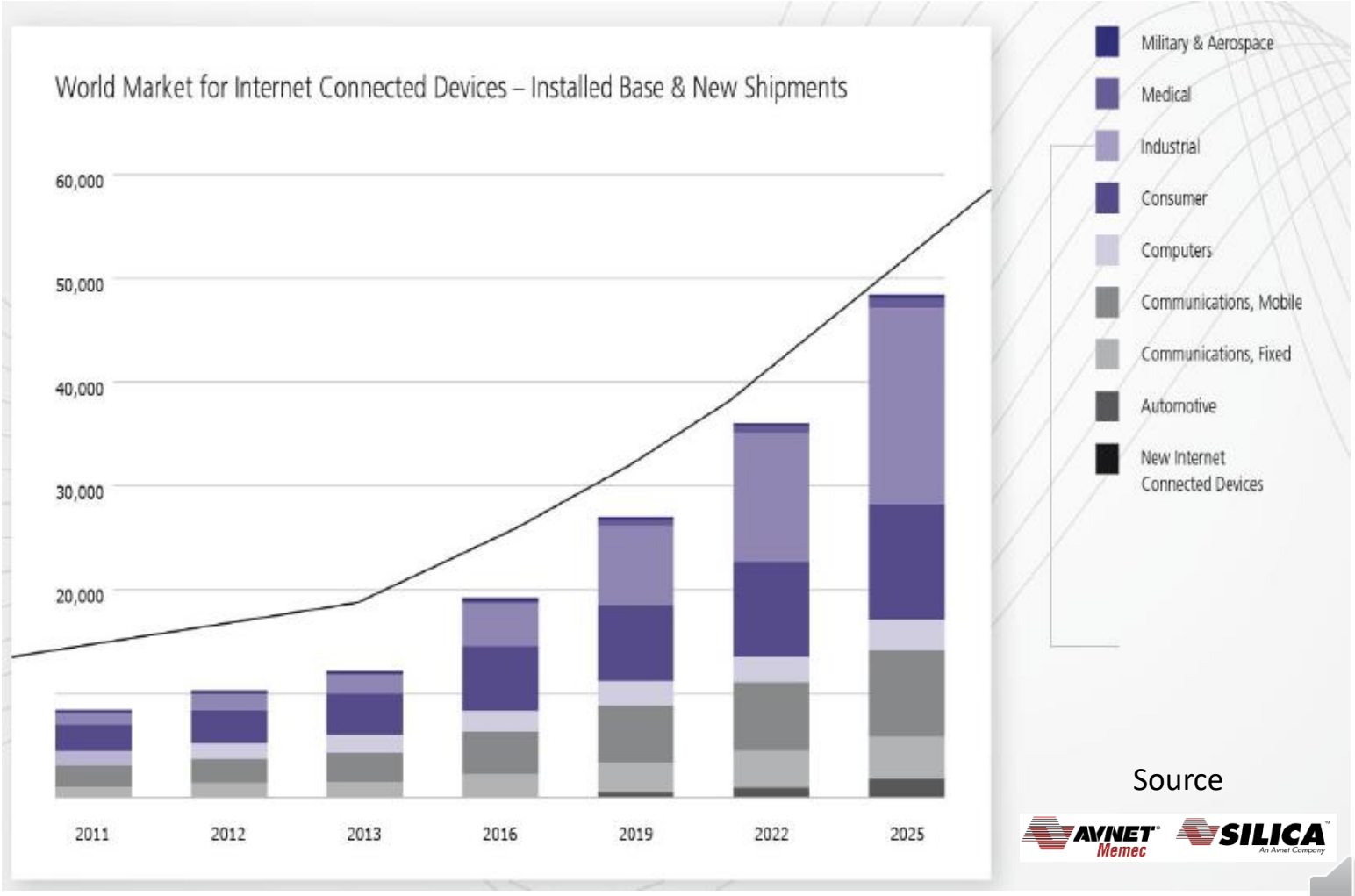
IoT Market Forecast for 2025



LPWAN will represent +26% of IoT Market

Source: McKinsey, June 2015

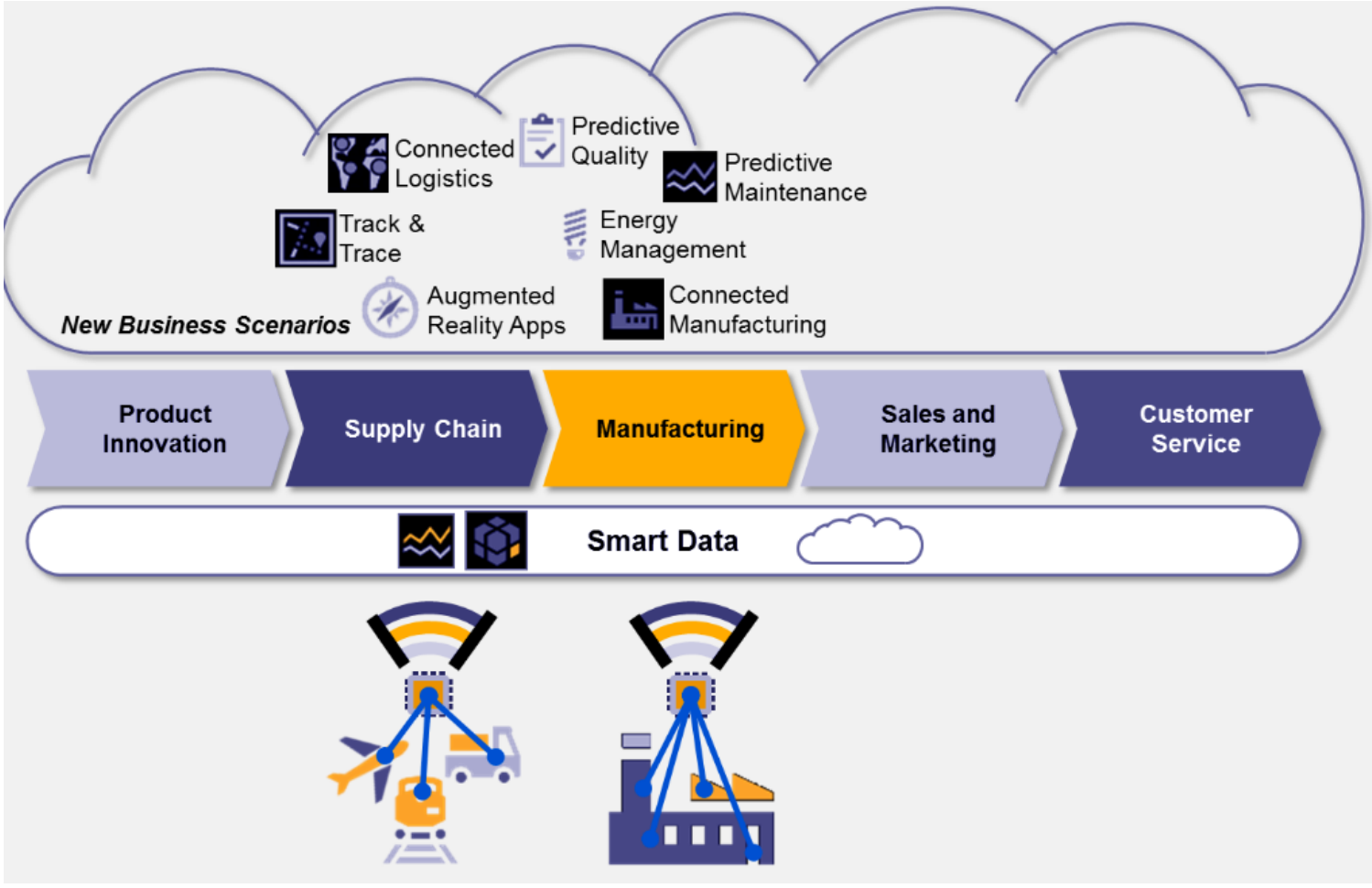
Connected Devices Market Evolution



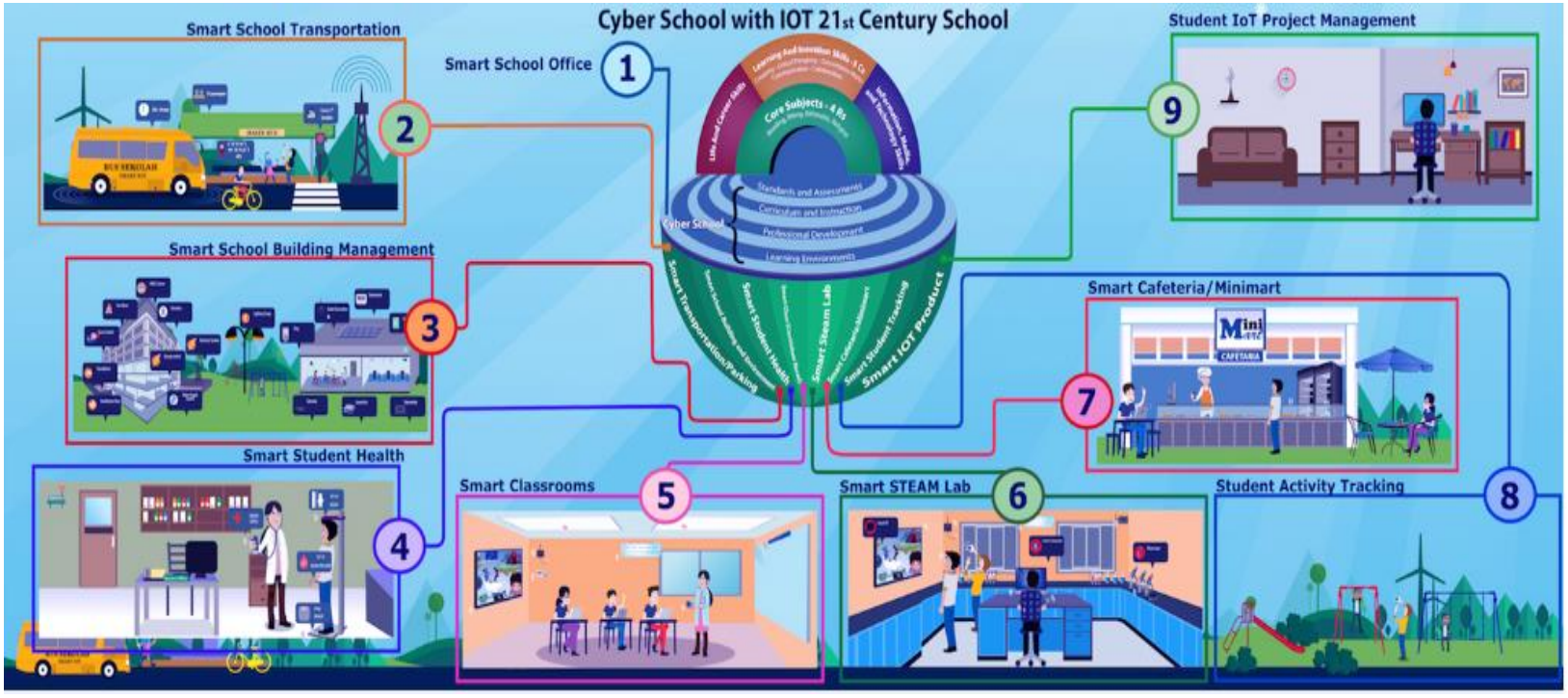
Source

AVNET Memec SILICA An Avnet Company

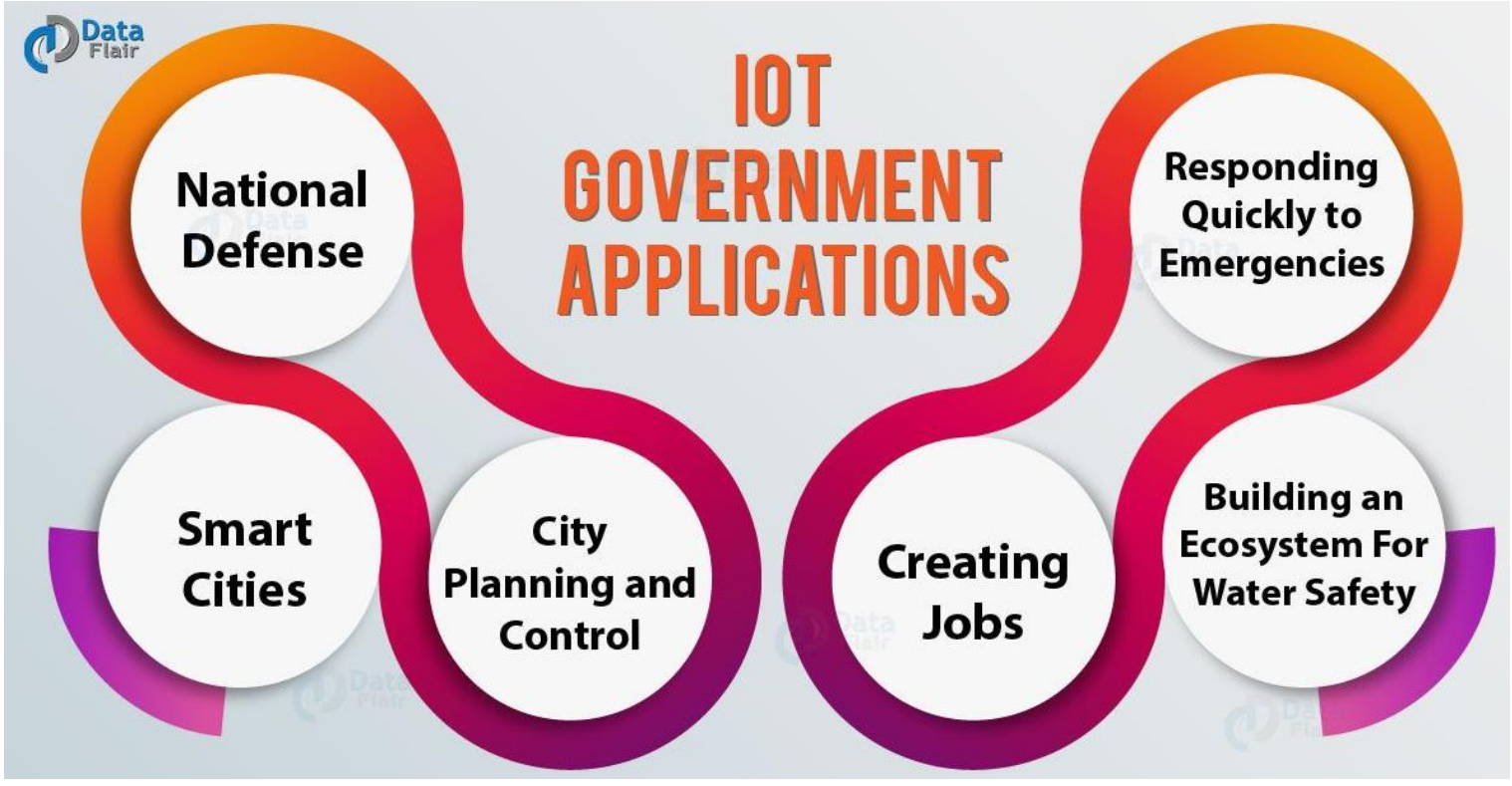
IoT Industrial Applications



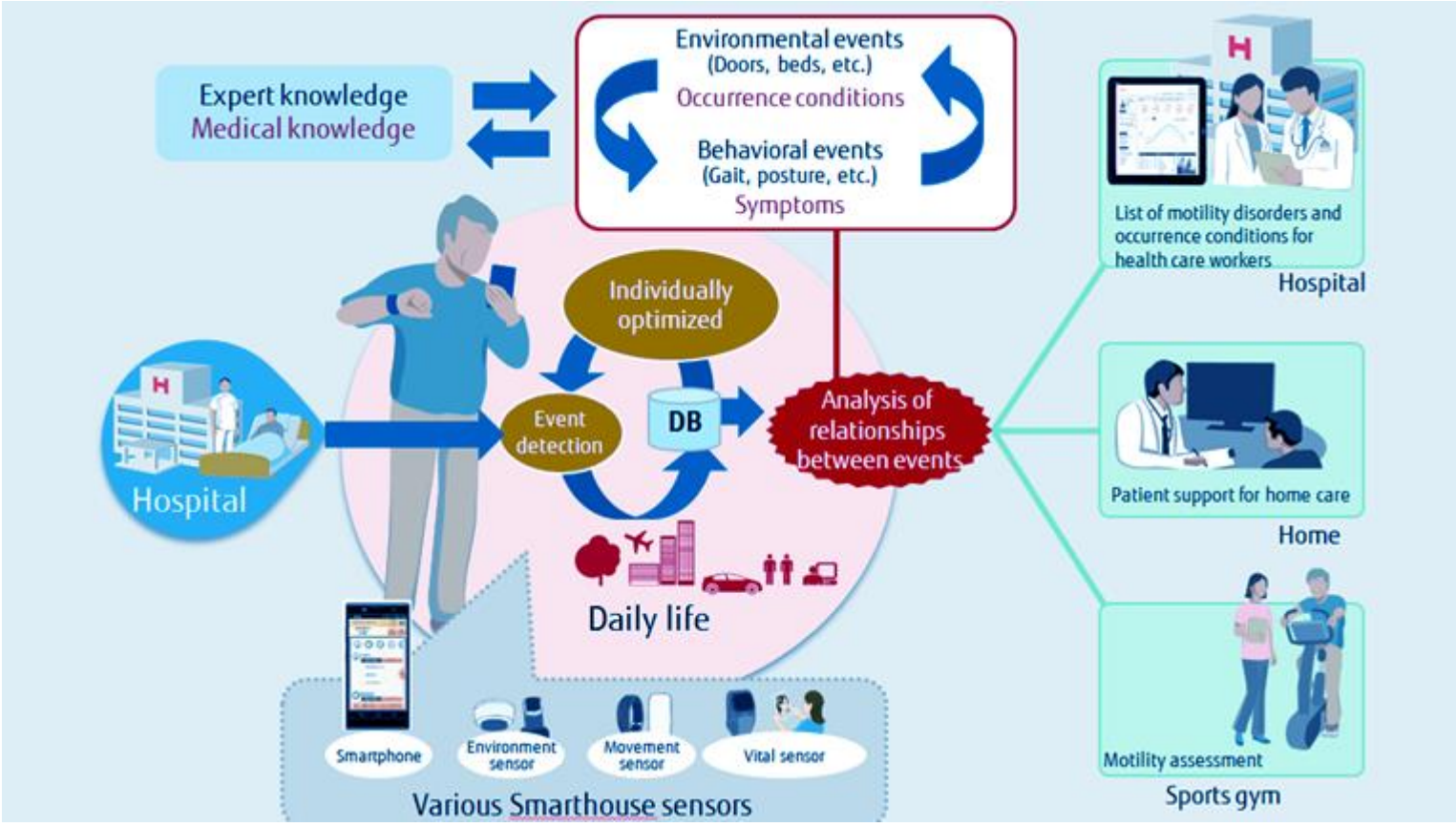
IoT Learning Applications



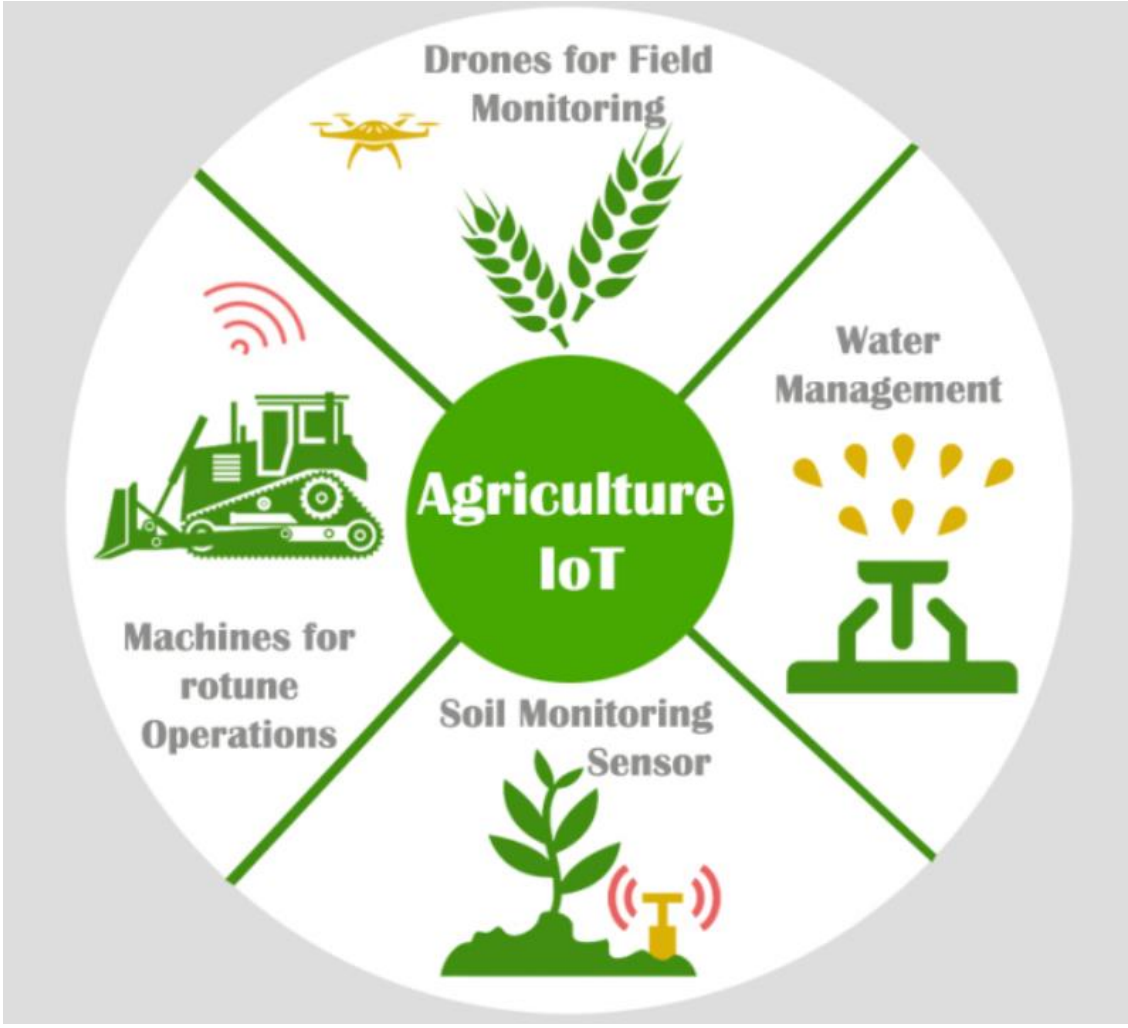
IoT Government Applications



IoT Health Applications

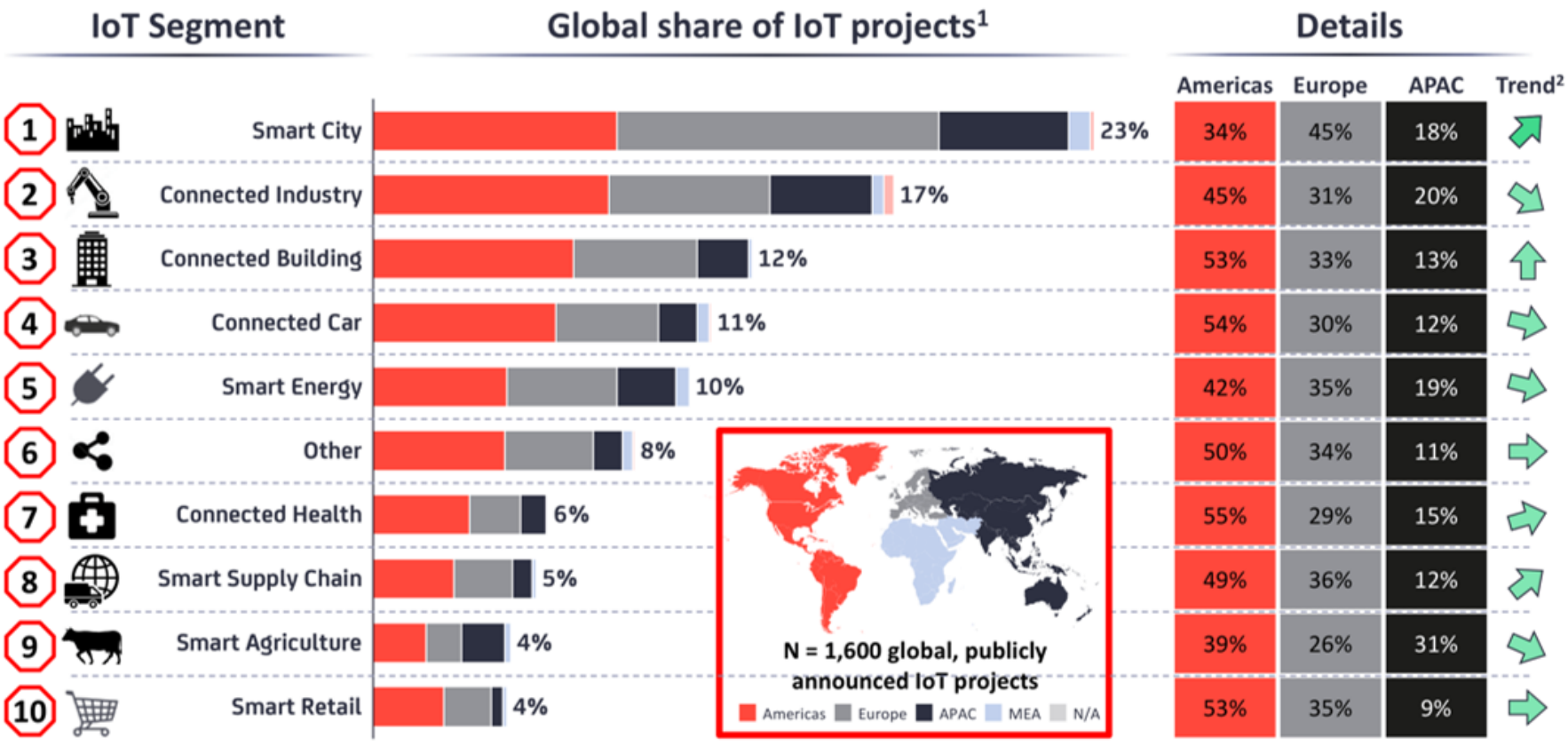


IoT Agriculture Applications

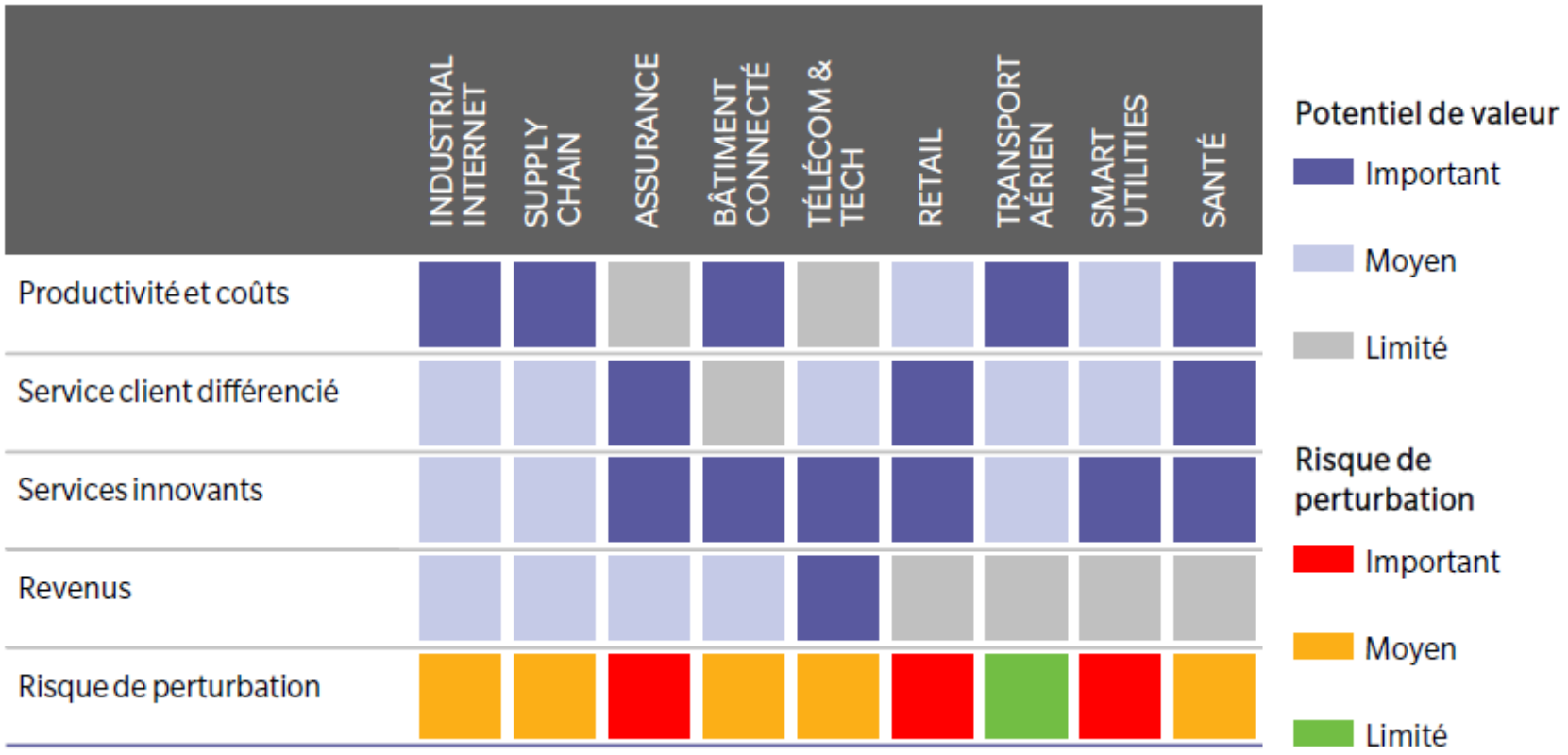


IoT Smart Building Applications

IoT Projects Share per Economic Sector

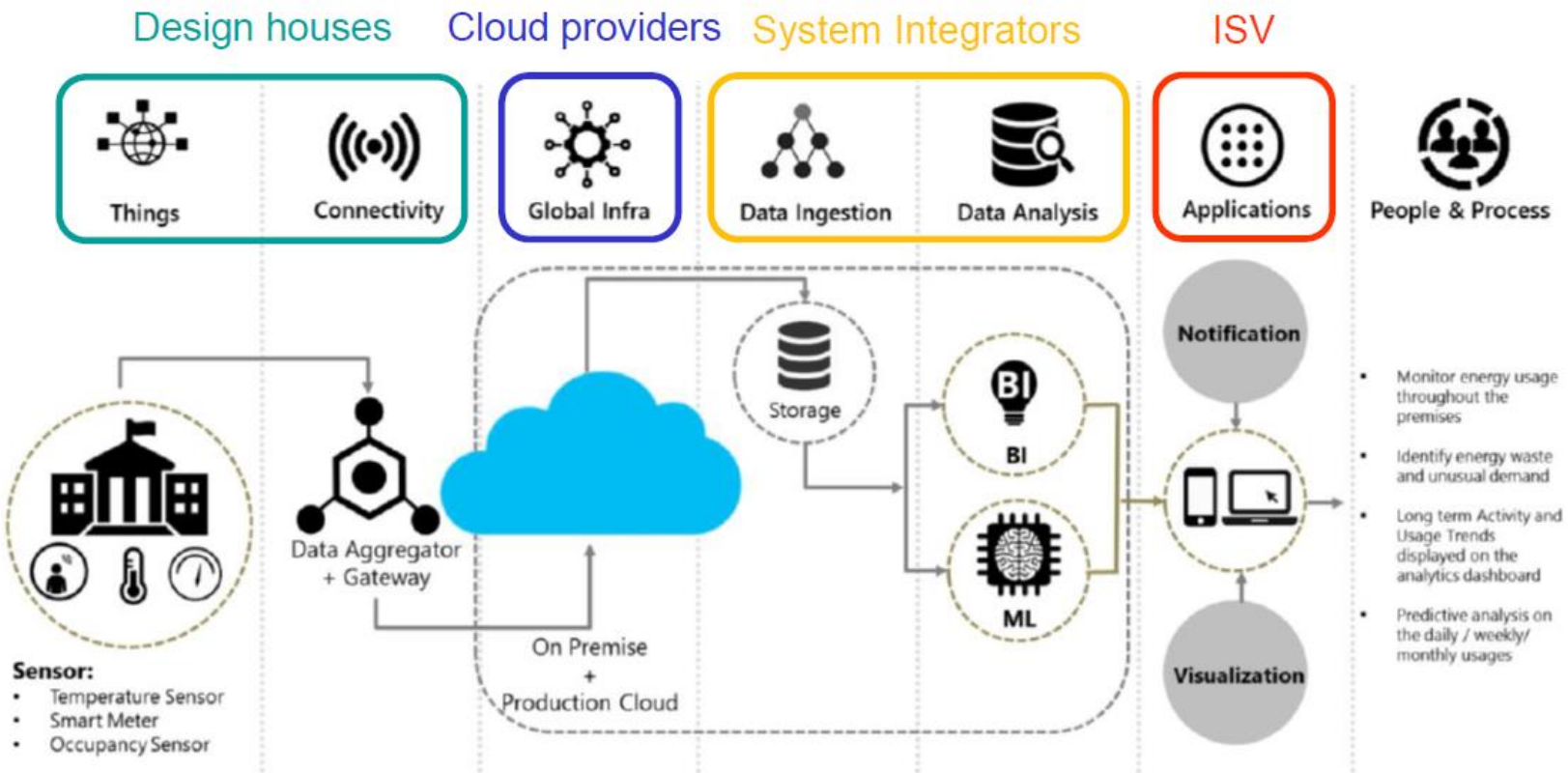


Potential Value / Risk Level of IoT by Vertical



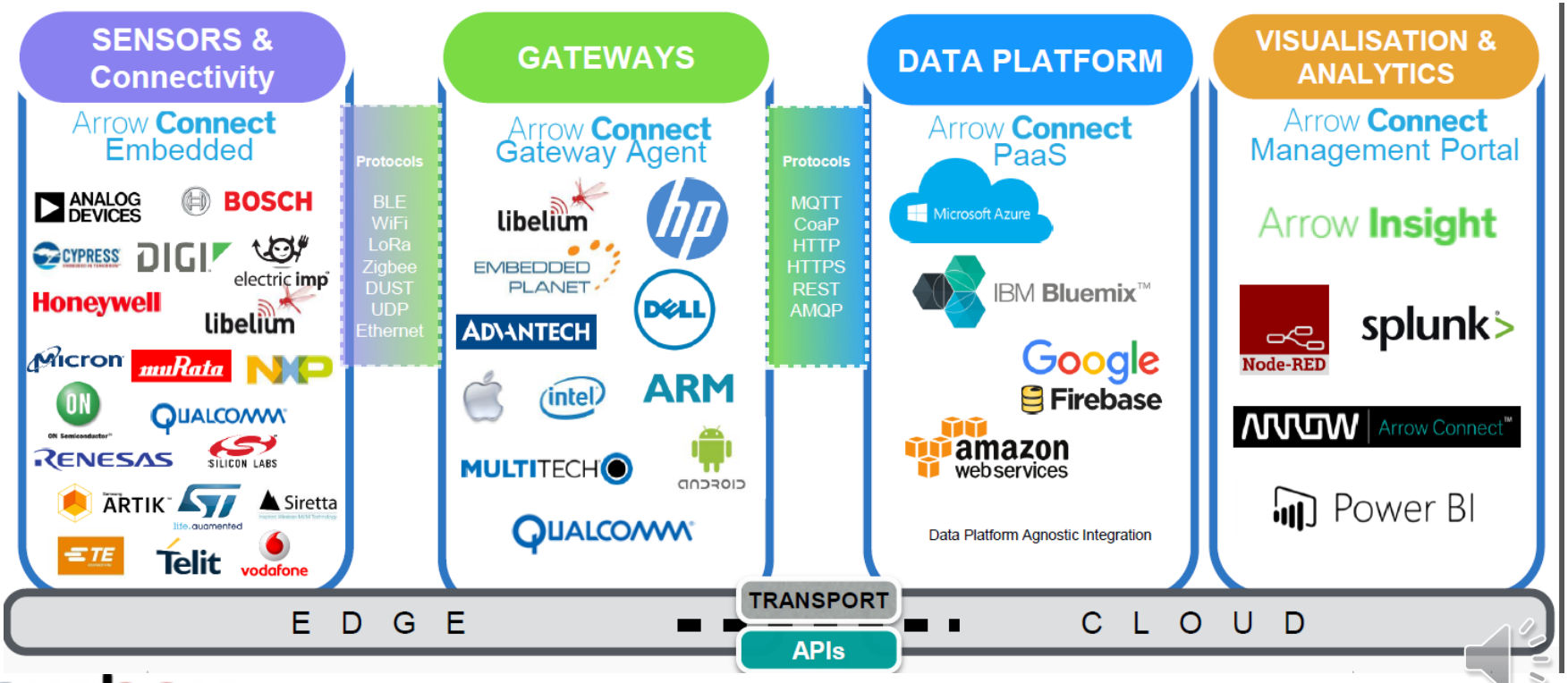
Source: OLIVER WYMAN

IoT Ecosystem Actors



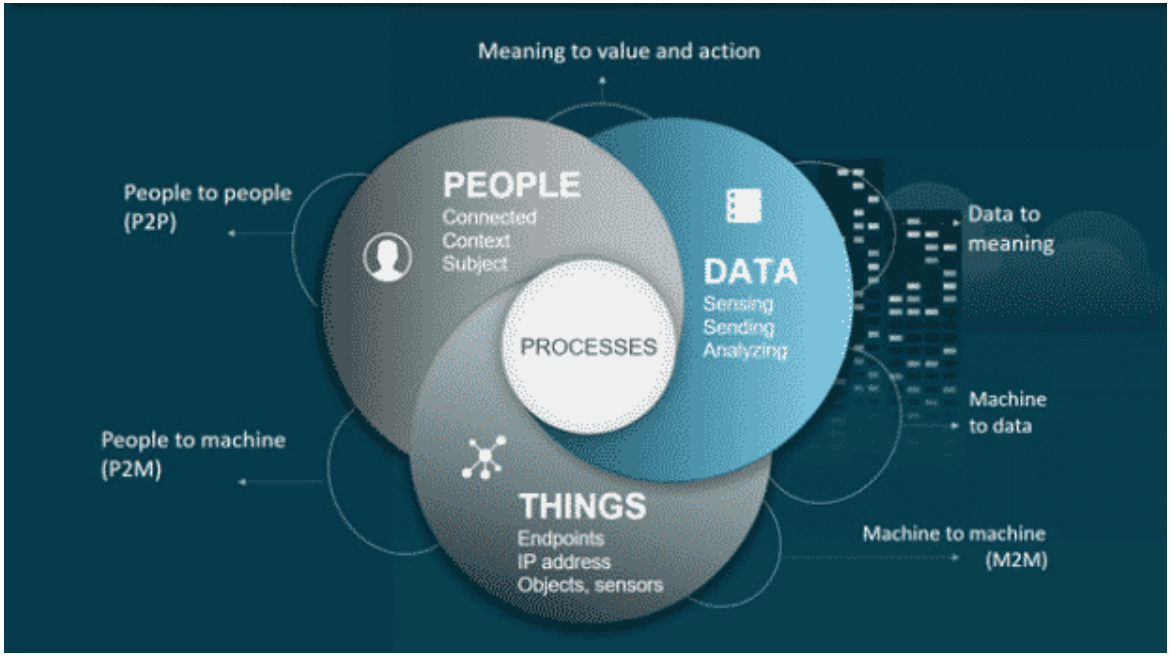
IoT Technology Providers

IBM Google Intel Microsoft Cisco



IoT Start-ups in Europe

Evolution towards Internet of Everything (IoE)



IoE brings together people (humans), processes (manages how people, data, and objects work together), data (rich information), and elements (inanimate objects and devices) to make network connections more relevant and more useful by transforming information into actions that create new capabilities, richer experiences and unprecedented economic opportunities for companies, individuals and countries

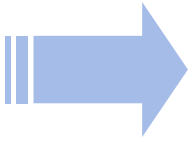
IoT Driving Jobs for the Futur

Digital Engineering Professions

- Data scientist
- Robotics engineer
- Embedded Electronic Engineer
- User eXperience Designer

- Chief Happiness Officer (CHO),
- Senior Well-being Consultant
- Body engineer for transplants,
- Nano-medicine specialist,
- Vertical farmer - urban farmer,
- Unused data manager,
- Climate controller,
- Avatar manager,
- Responsible for the ethics of technology,
- Virtual home designer,
- IOT data creator,
- Energy creator,

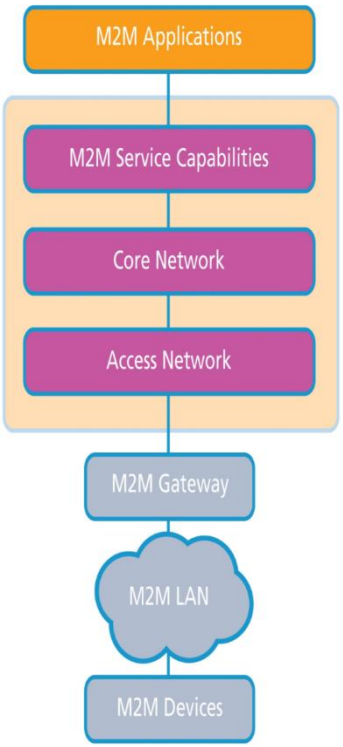
Part I-3



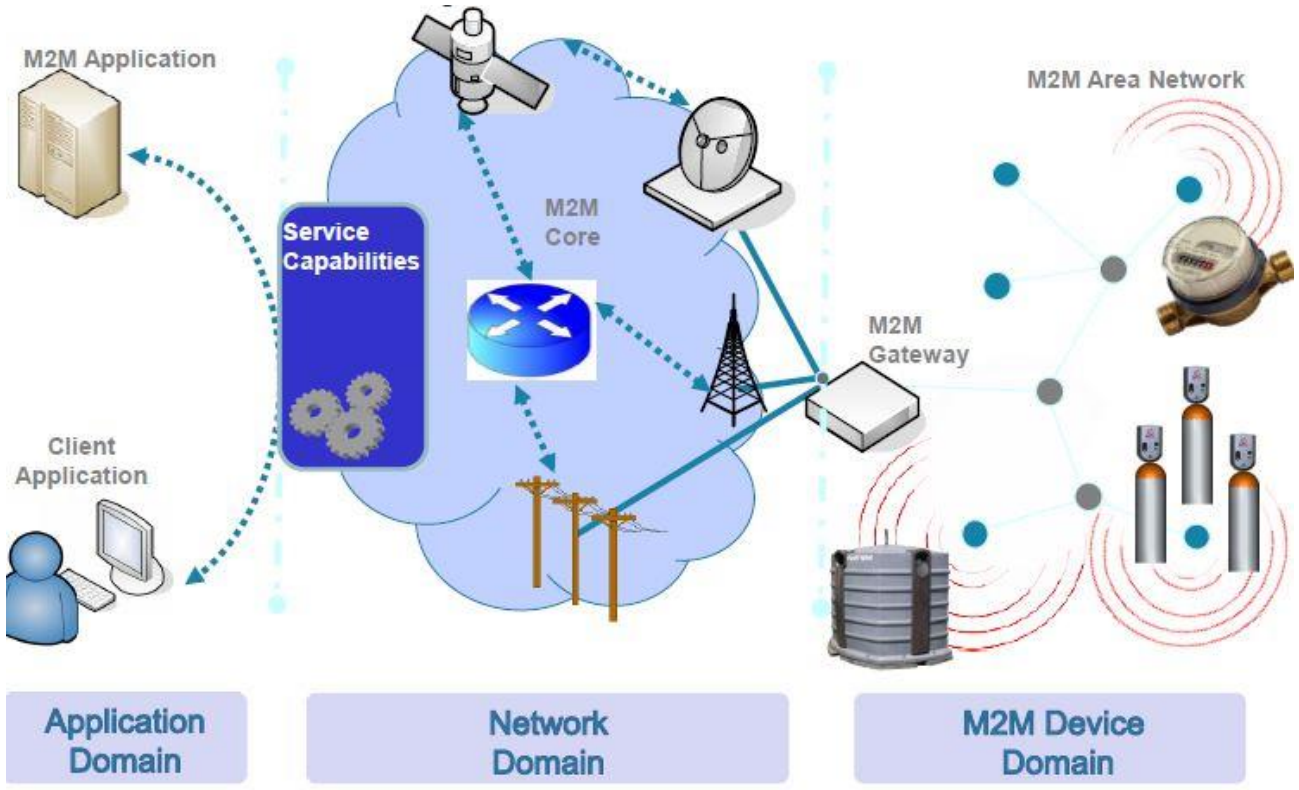
IoT Architecture Models

M2M Architecture Model - ETSI

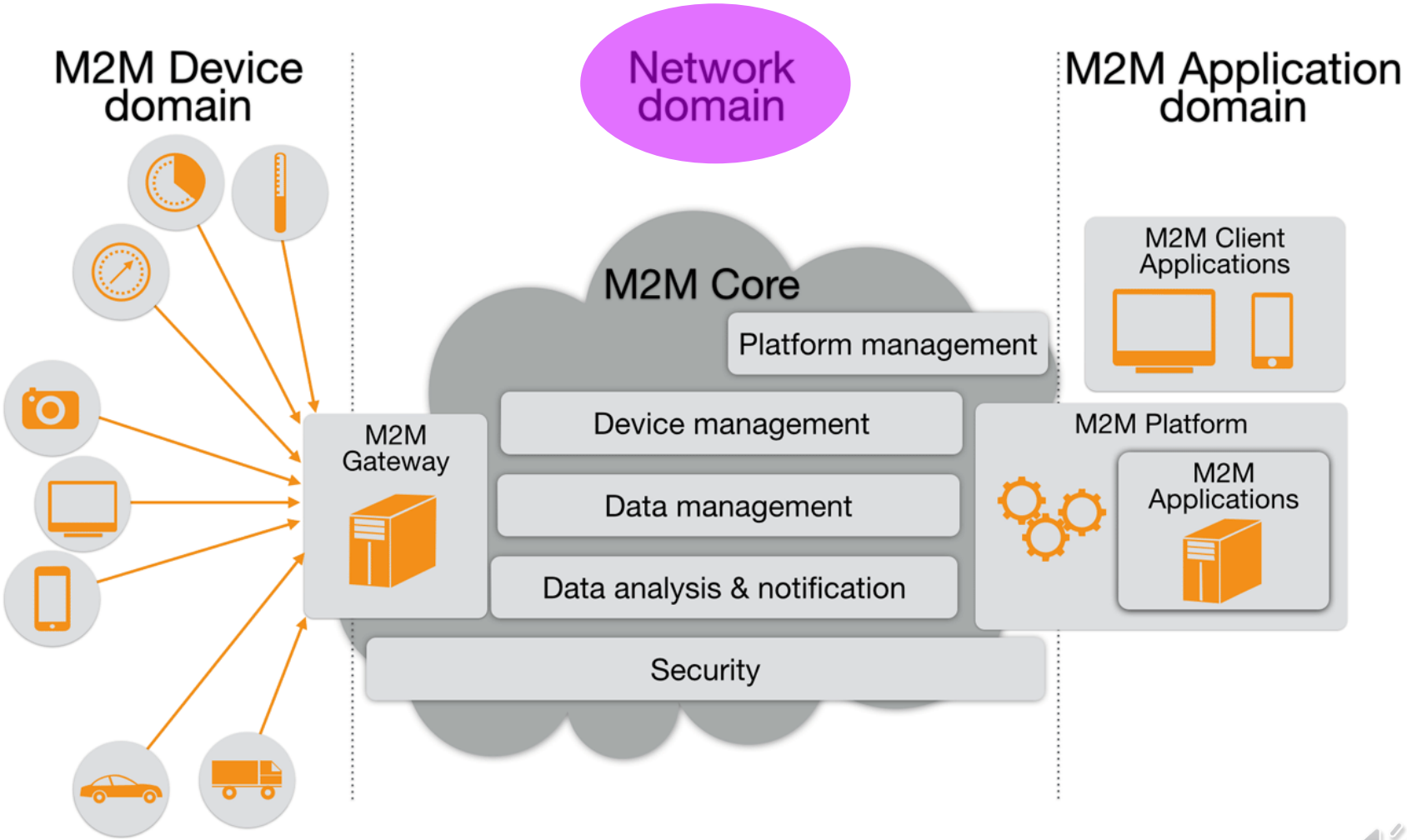
Generic Architecture



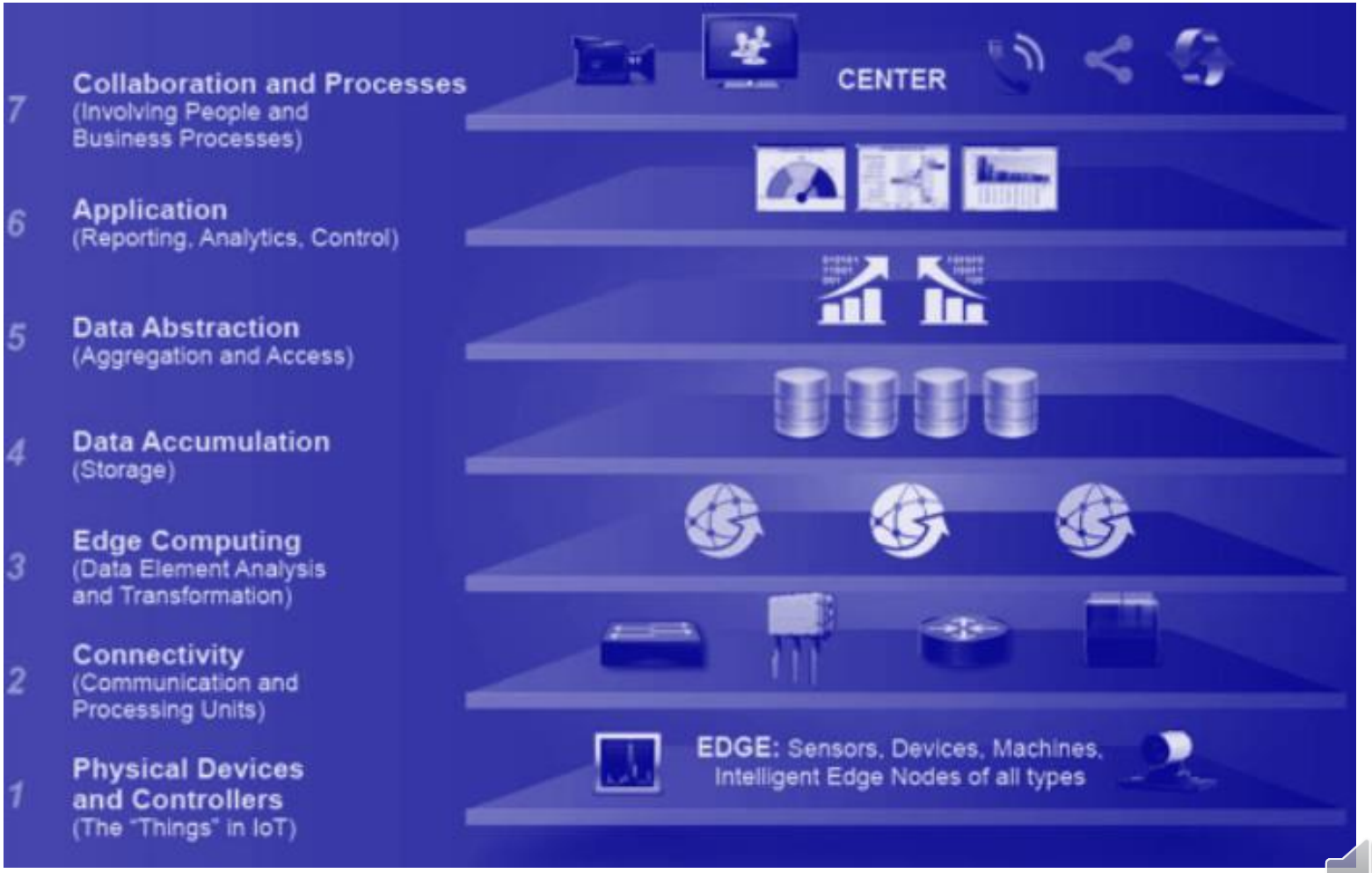
ETSI Architecture Model



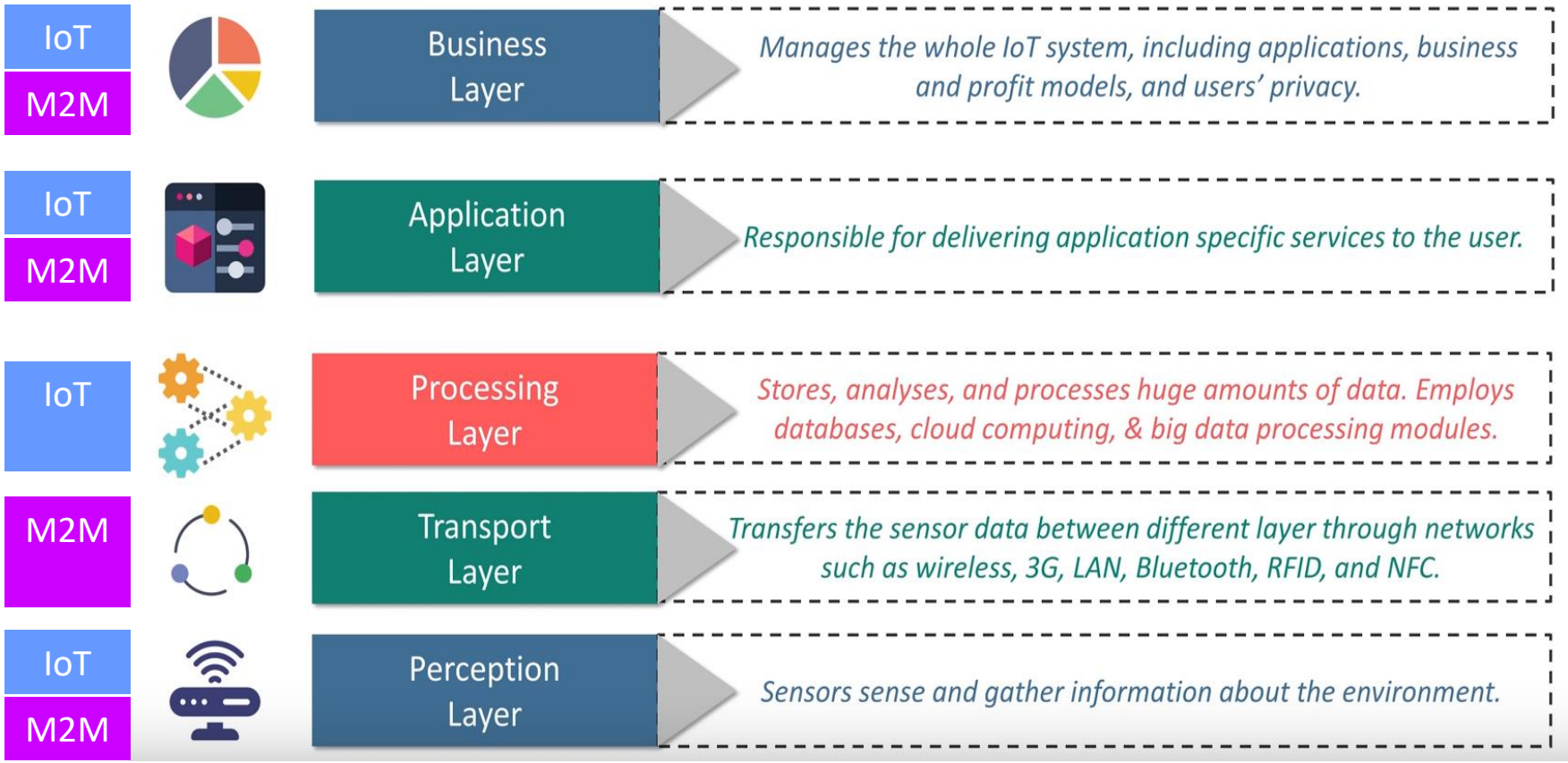
M2M Network Domain Services



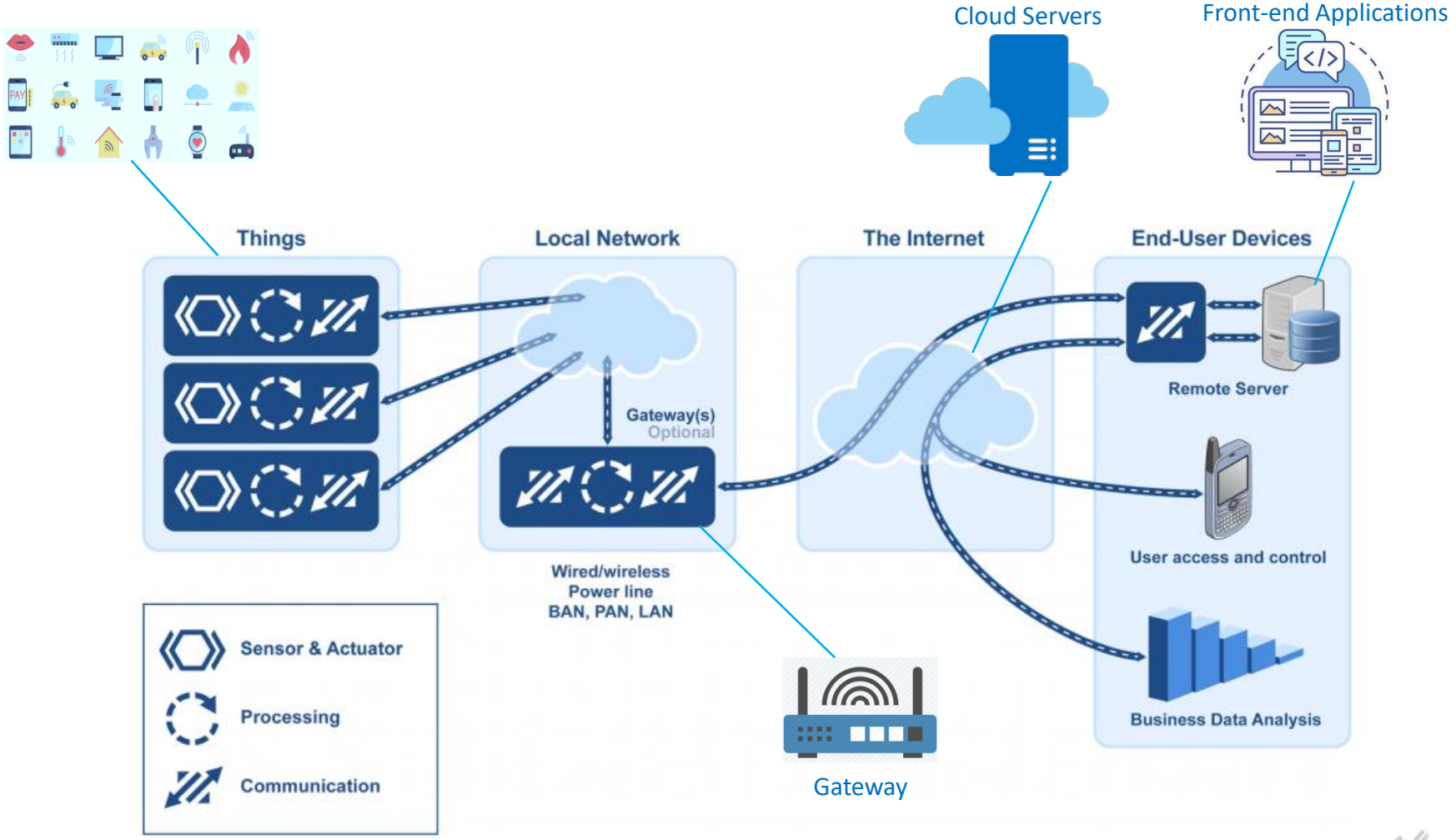
Functional Requirements for IoT Architecture



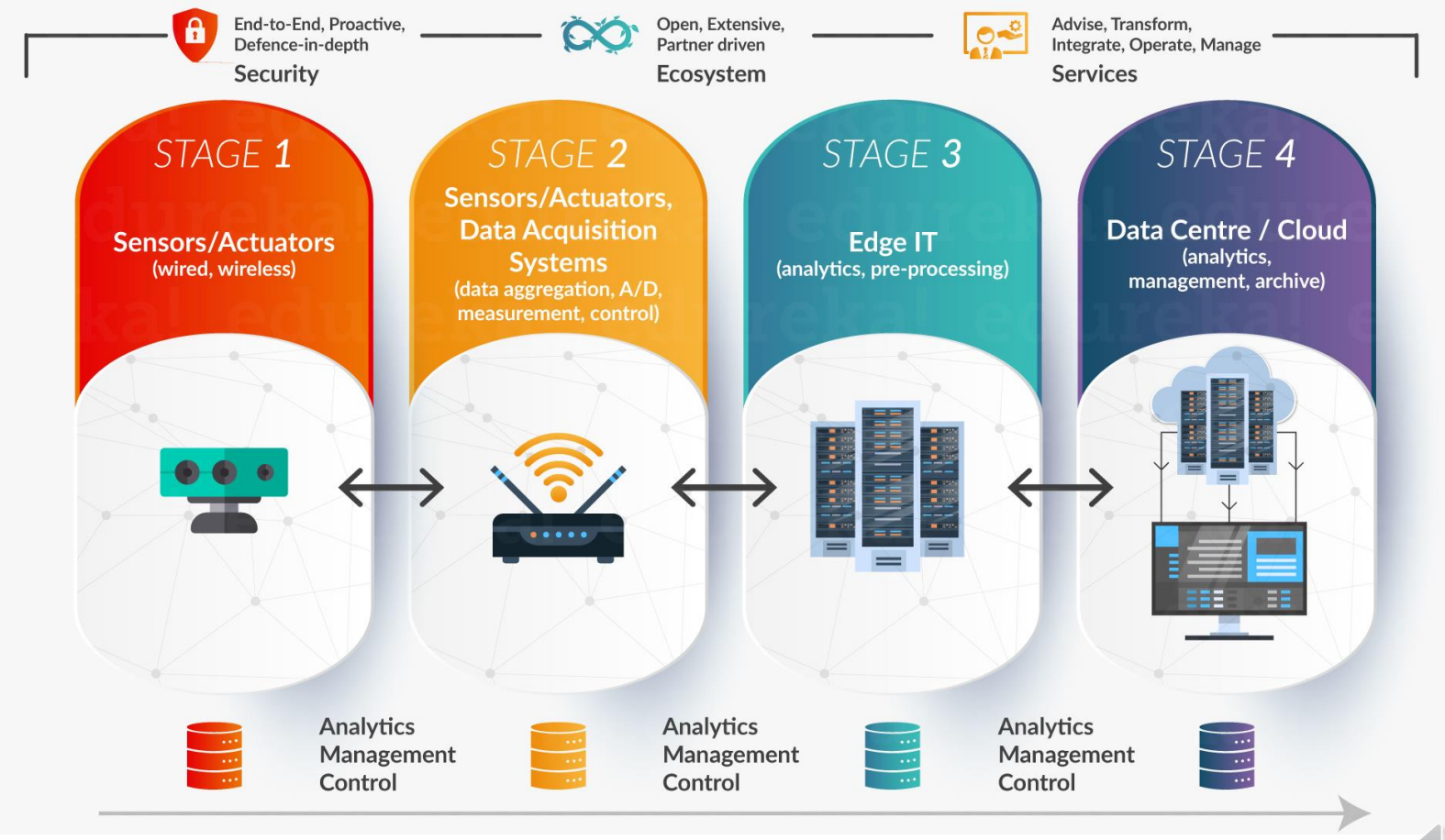
IoT Architecture Layers



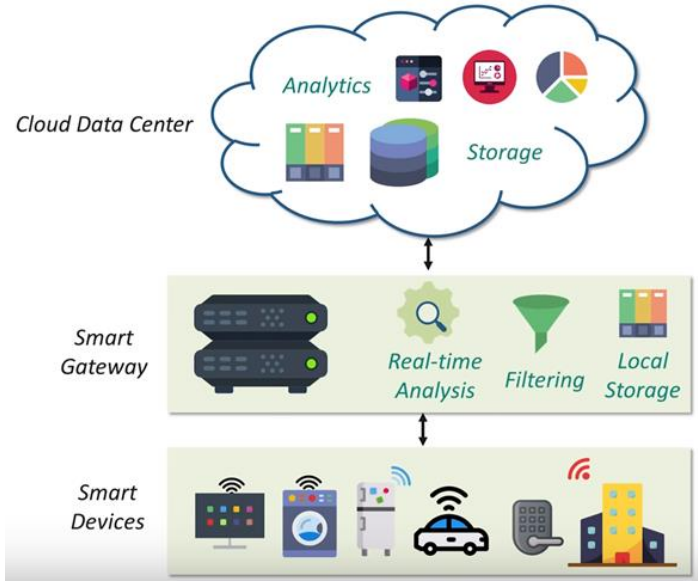
IoT Architecture Structural Model



IoT Architecture Functional Model

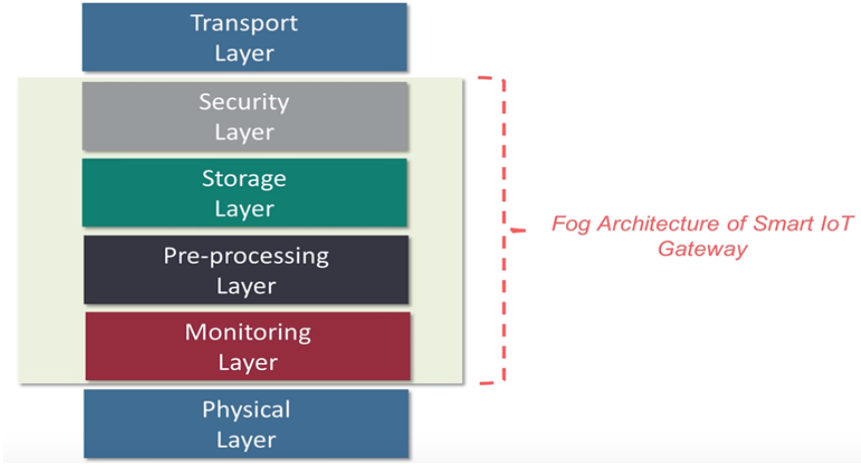


Cloud Computing vs Fog Computing



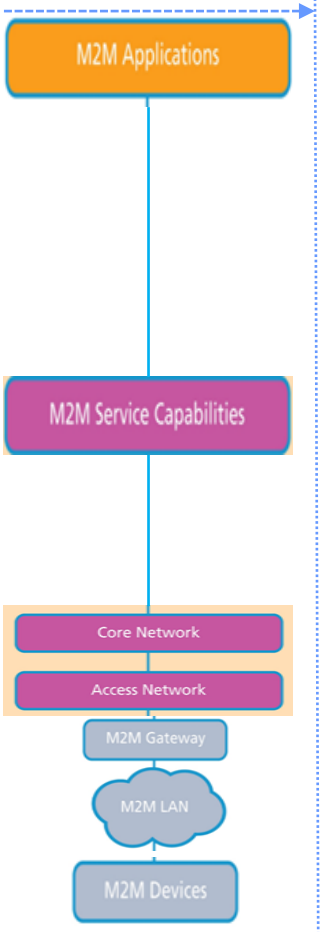
Fog architecture →

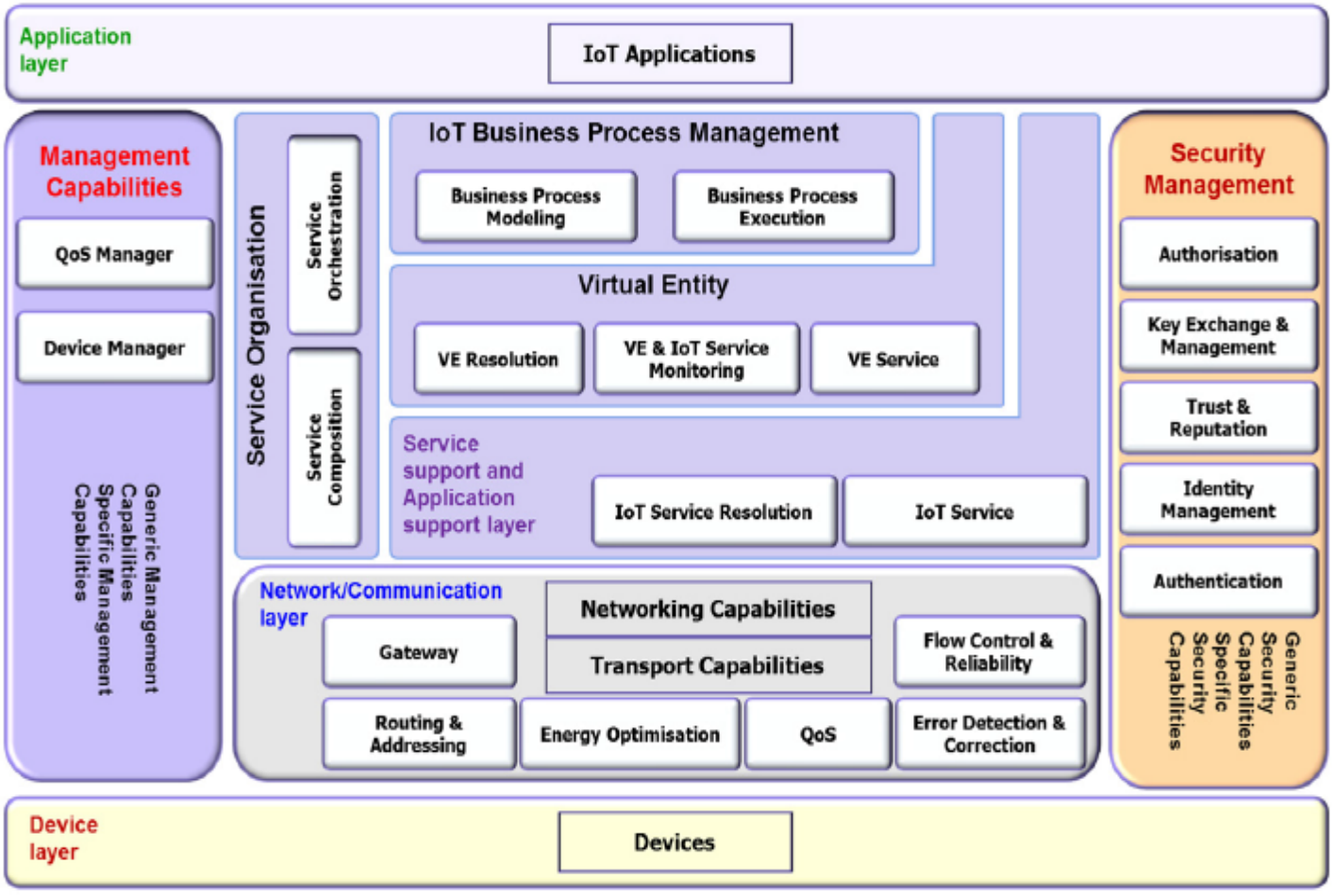
Insertion between the physical layer and the transport layer of 4 layers: monitoring, pre-processing, storage and security



IoT Architecture Formel Model

M2M Architecture



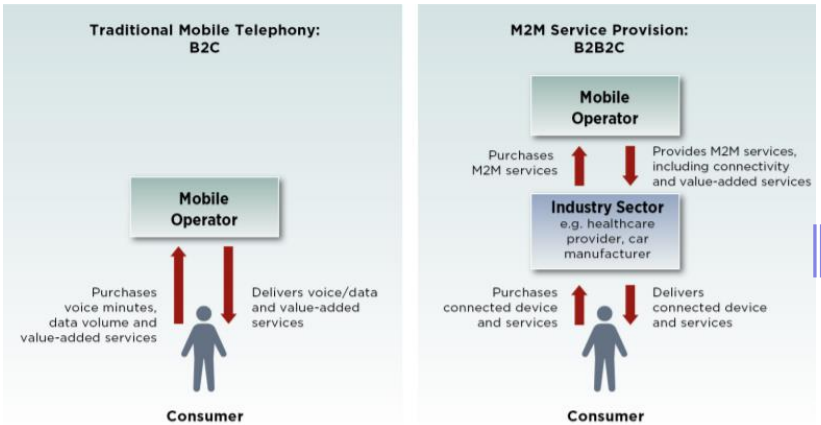


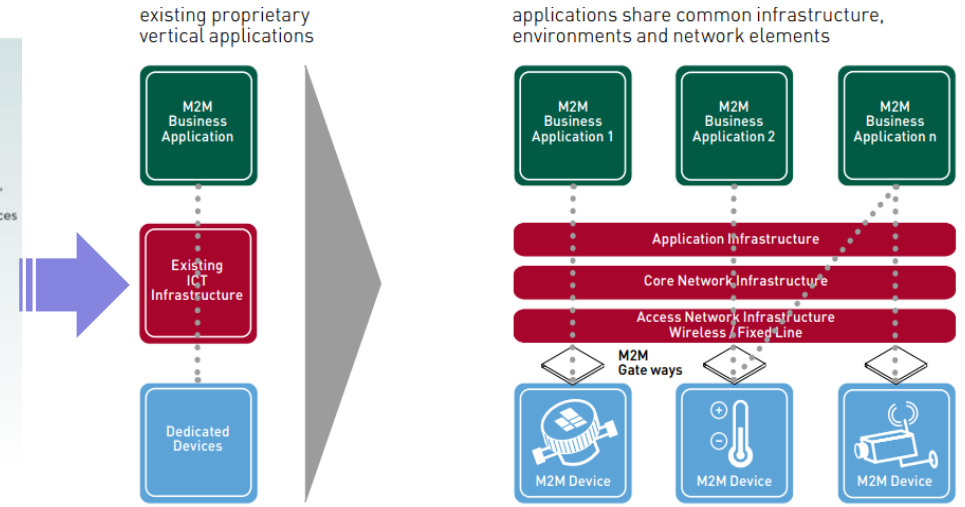
Source Dr. Peter Friess Dr. Ovidiu Vermesan
Internet of Things – From Research and Innovation to Market Deployment

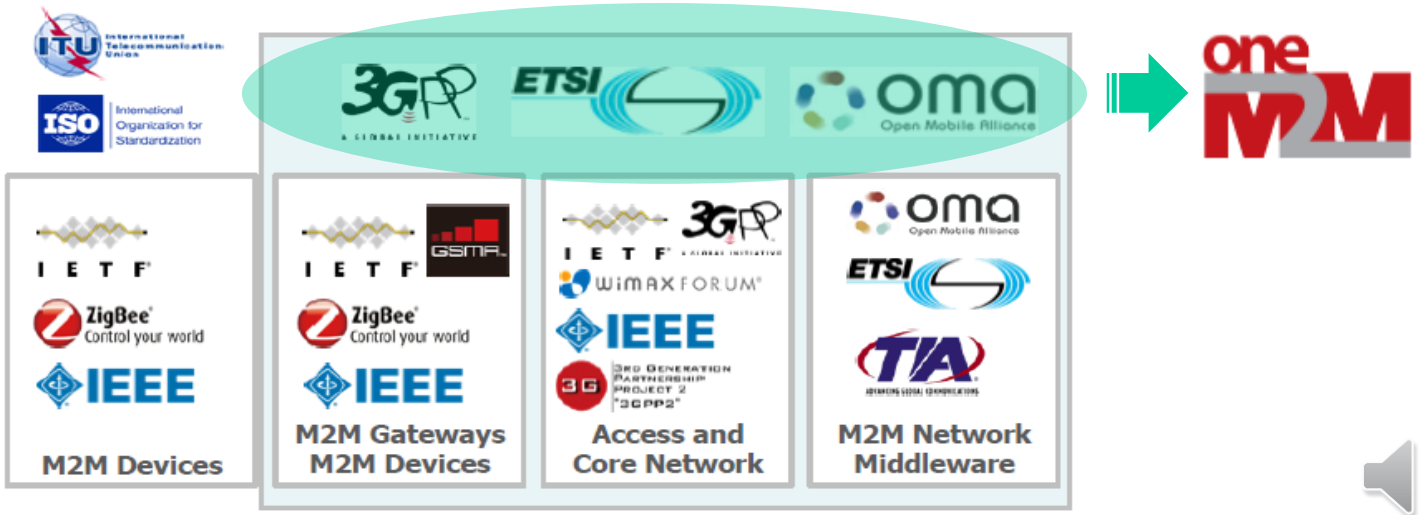
IoT Communication Standards Landscape

Need for MTC Standardization

MTC: Machine Type Communication





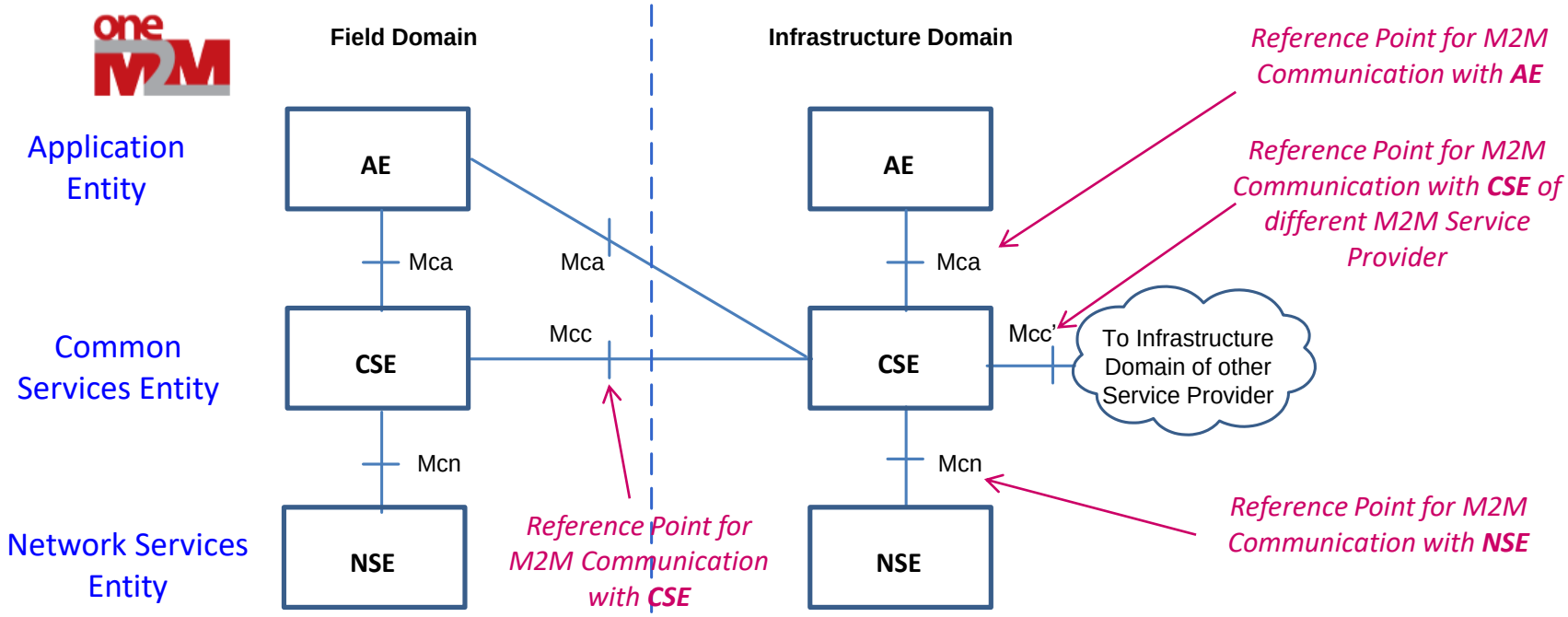


OneM2M Standard Scope Statement

OneM2M covers the necessary set of Technical Specifications and Technical Reports for:

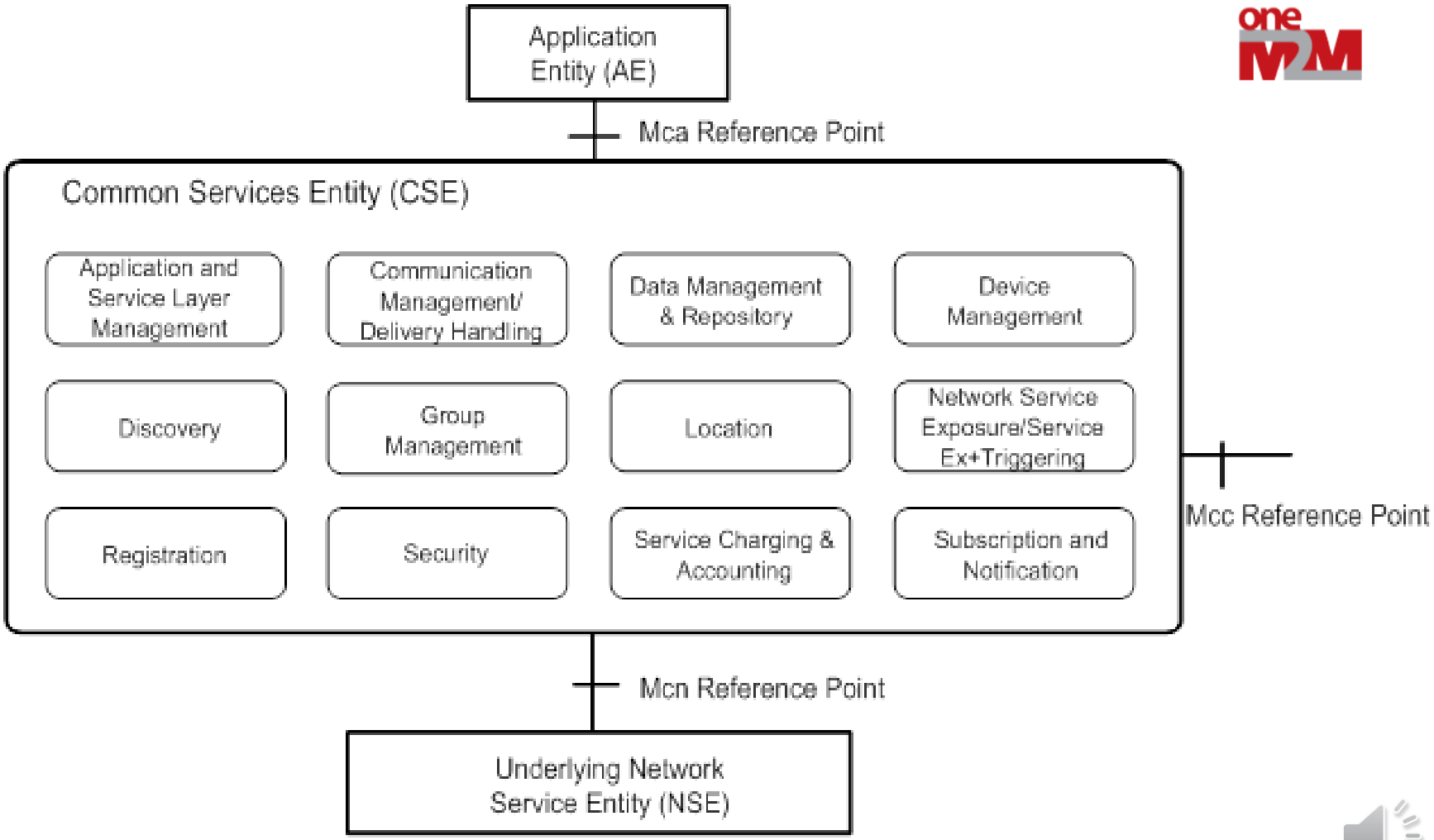
- Use cases and requirements for a common set of Service Layer capabilities;
- Service Layer aspects with high level and detailed service architecture, in light of an access independent view of end-to-end services;
- Protocols/APIs/standard objects based on this architecture (open interfaces & protocols);
- Security and privacy aspects (authentication, encryption, integrity verification);
- Reachability and discovery of applications;
- Interoperability, including test and conformance specifications;
- Collection of data for charging records (to be used for billing and statistical purposes);
- Identification and naming of devices and applications;
- Information models and data management (including store and subscribe/notify functionality);
- Management aspects (including remote management of entities); and
- Common use cases, terminal/module aspects, including Service Layer interfaces/APIs between:
  - Application and Service Layers;
  - Service Layer and communication functions

OneM2M Functional Architecture



- **Application Entity (AE)**: instantiation of application service logic, identified with a unique AE-ID – Ex: fleet tracking application, remote blood sugar monitoring application,..
- **Common Services Entity (CSE)**: instantiation of "common service functions" identified with a unique CSE-ID – Ex: Data Management, Device Management, M2M Service Subscription Management, …
- **Network Services Entity (NSE)**: provides services from the underlying network to the CSEs – Ex: device management, location services and device triggering.

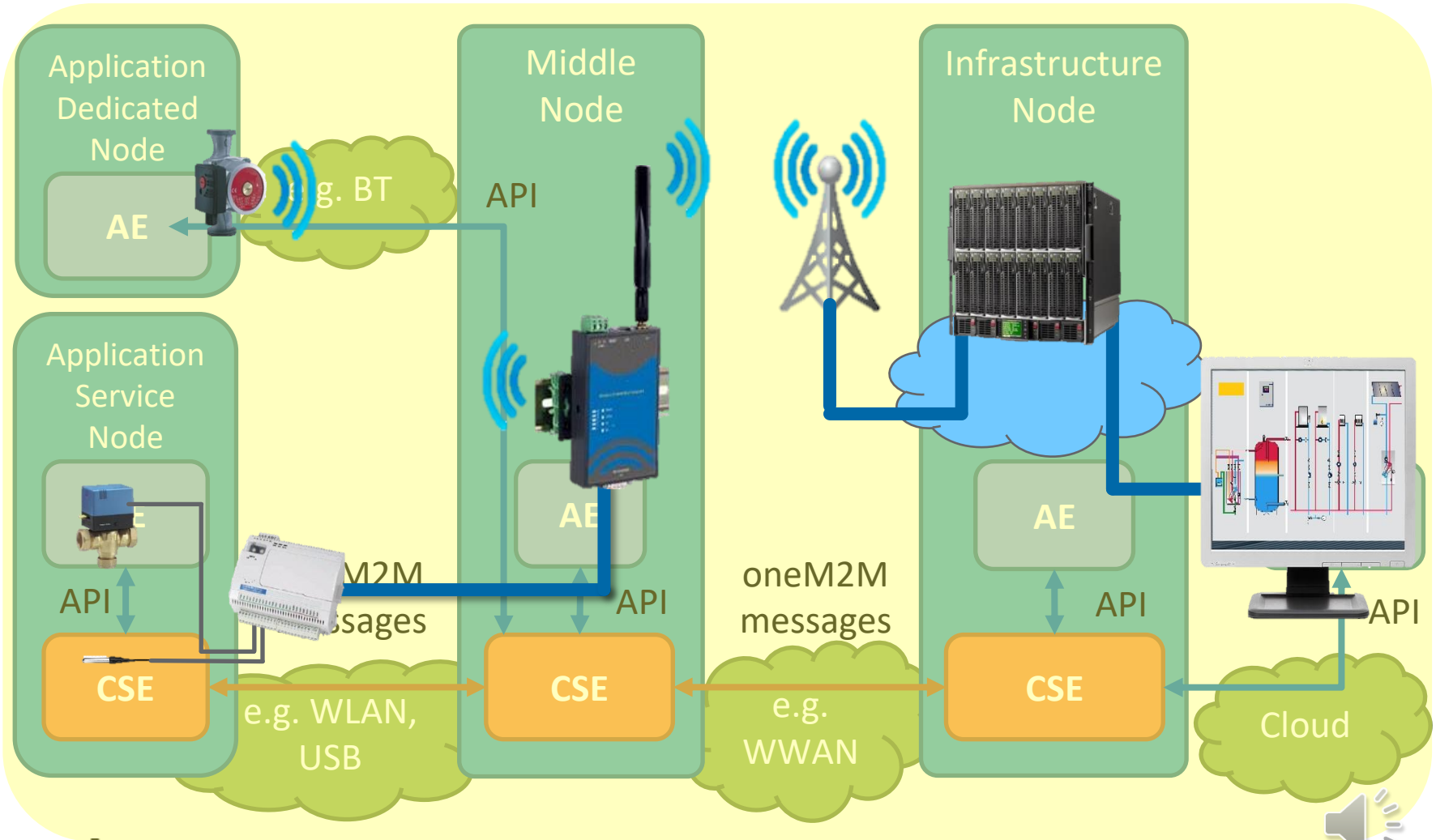
OneM2M Common Services Functions



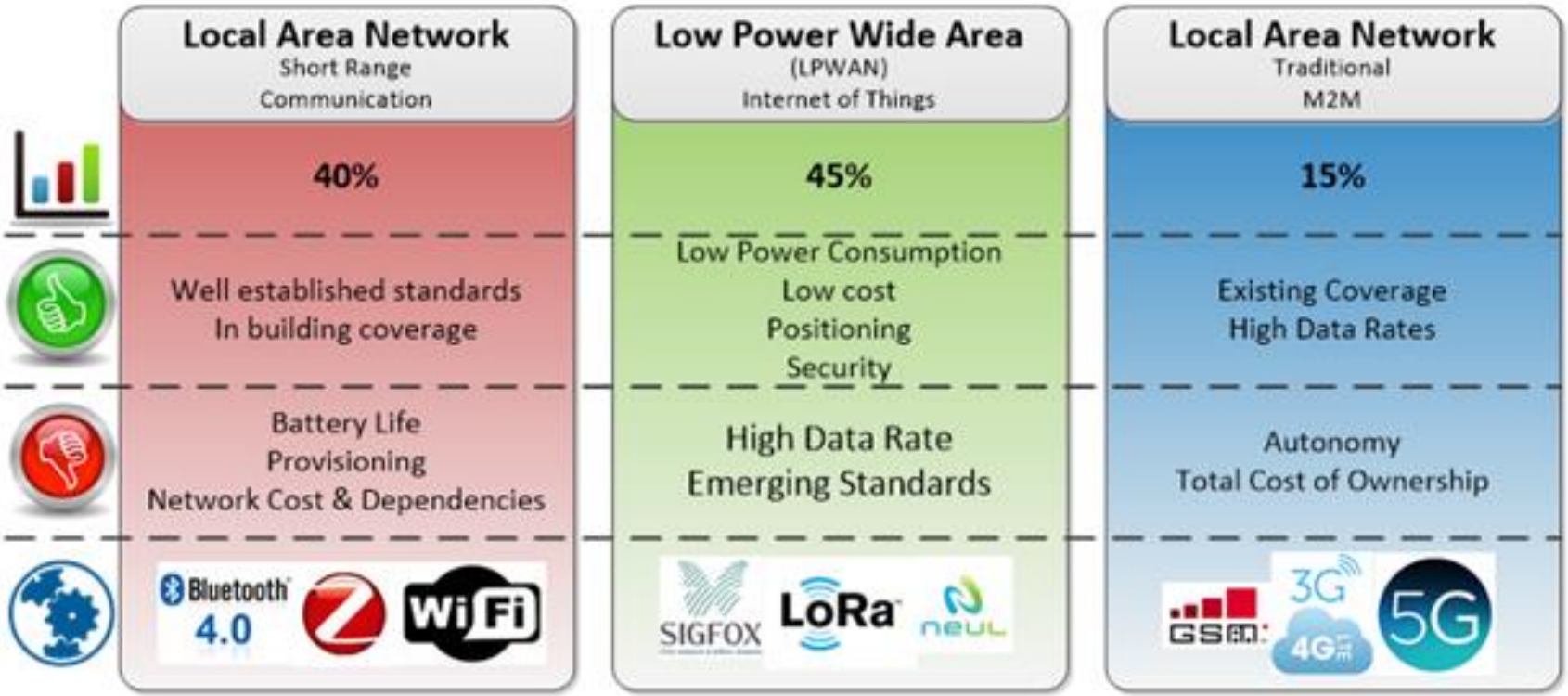
Functional Requirements Categories

- Management Requirements – example: *support management and configuration of M2M Gateways/ Devices including resource constrained M2M Devices.*
- Abstraction Requirements – example: *provide a generic structure for data representation.*
- Semantics Requirements – example: *capabilities to manage semantic descriptions of resources and M2M Applications, e.g, create, retrieve, update, delete, associate/link.*
- Security Requirements – *example: incorporate protection against threats to its availability such as Denial of Service attacks*.
- Charging Requirements – example: *support collection of charging specific information related to the individual services facilitated by the M2M System (e.g. Data Management, Device Management and/or Connectivity Management).*
- Operational Requirements – example: *provide the capability for monitoring and diagnostics of M2M Applications*
- Communication Request Processing Requirements – example: *support M2M Gateways and M2M Devices that offer communication services to M2M Application to buffer incoming messages for communicating data to another M2M Gateway/Device/Infrastructure Domain.*

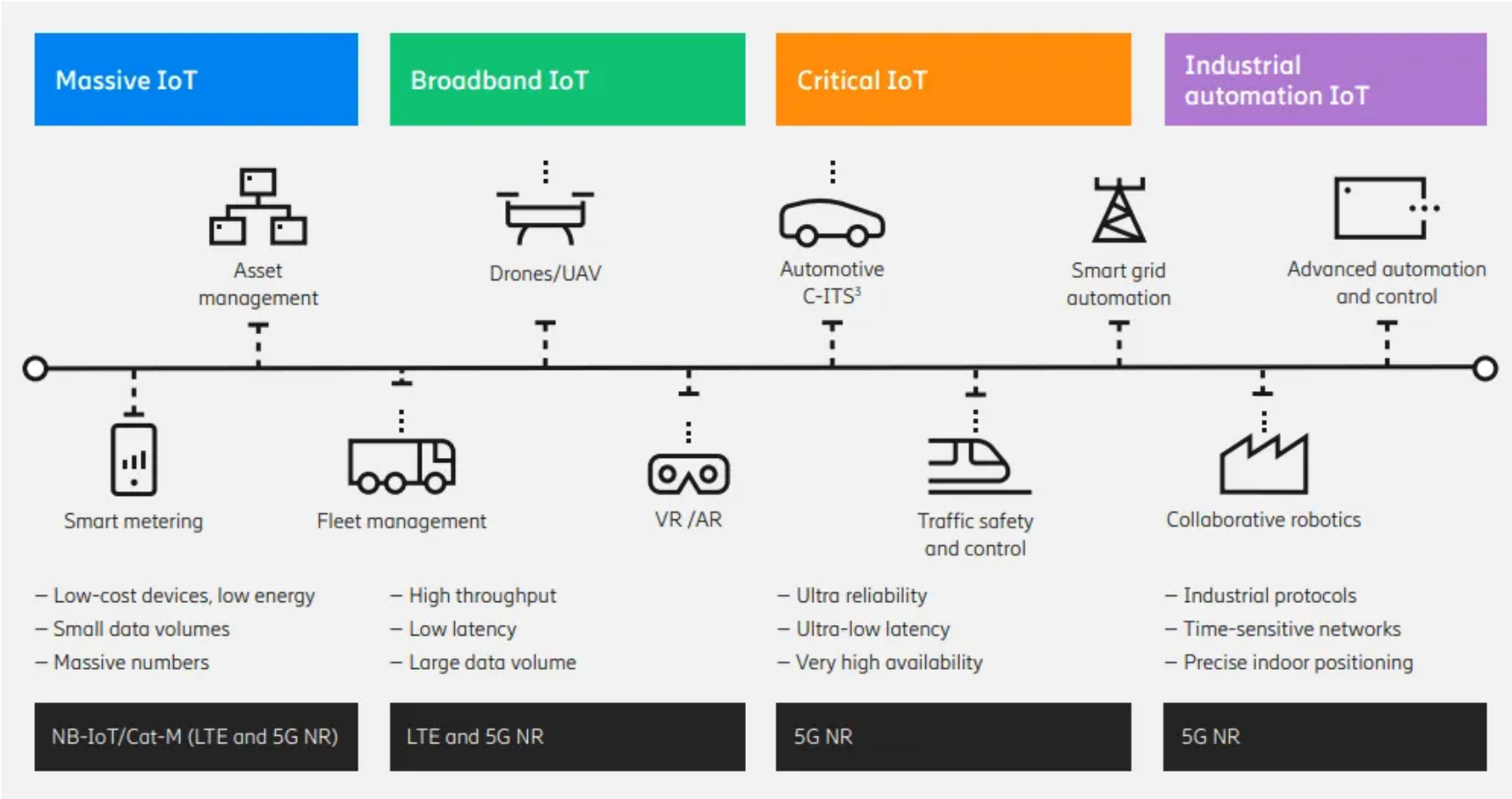
Configuration Example of oneM2M Architecture



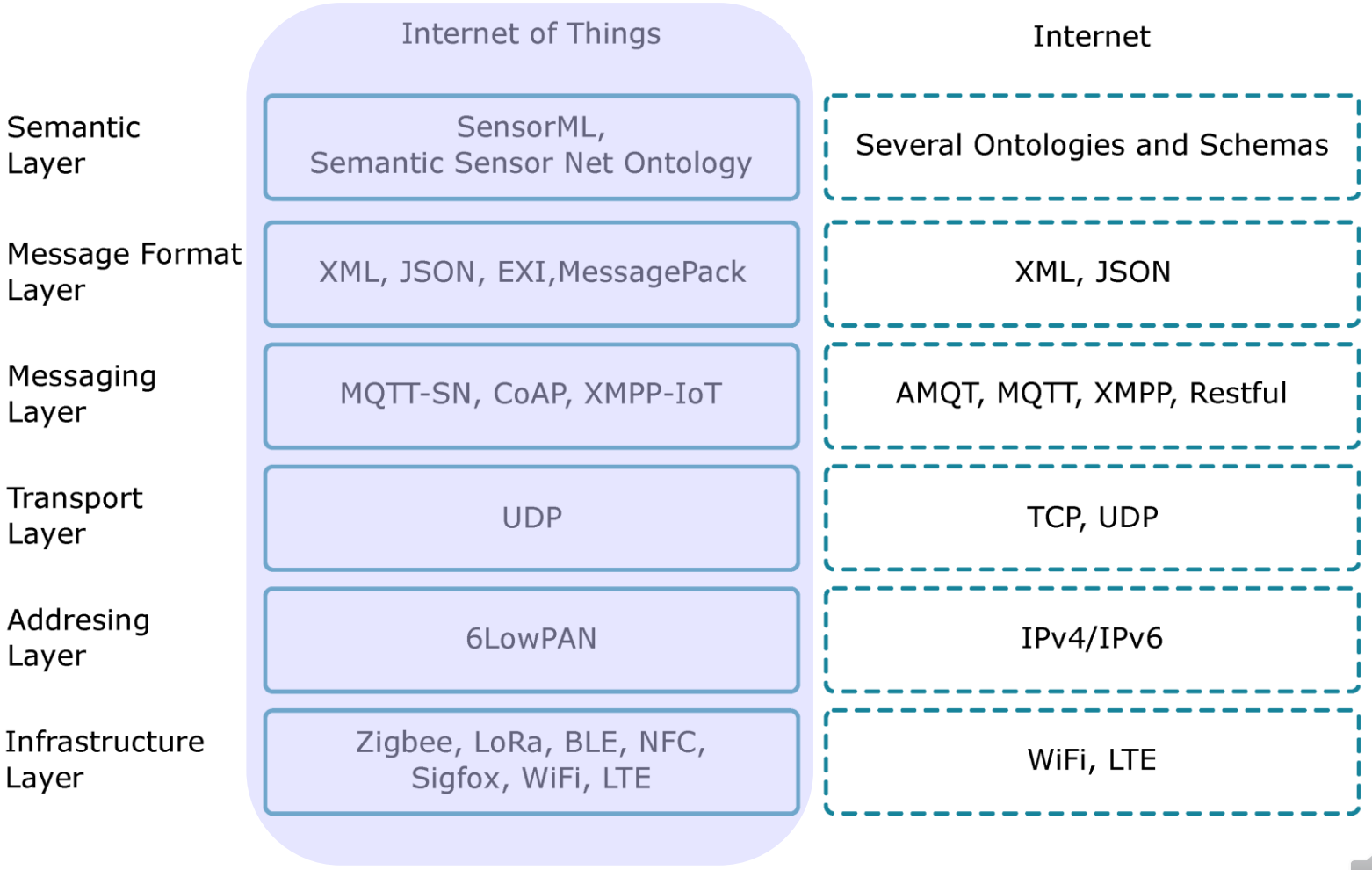
Wireless Networks for IoT/M2M Communication



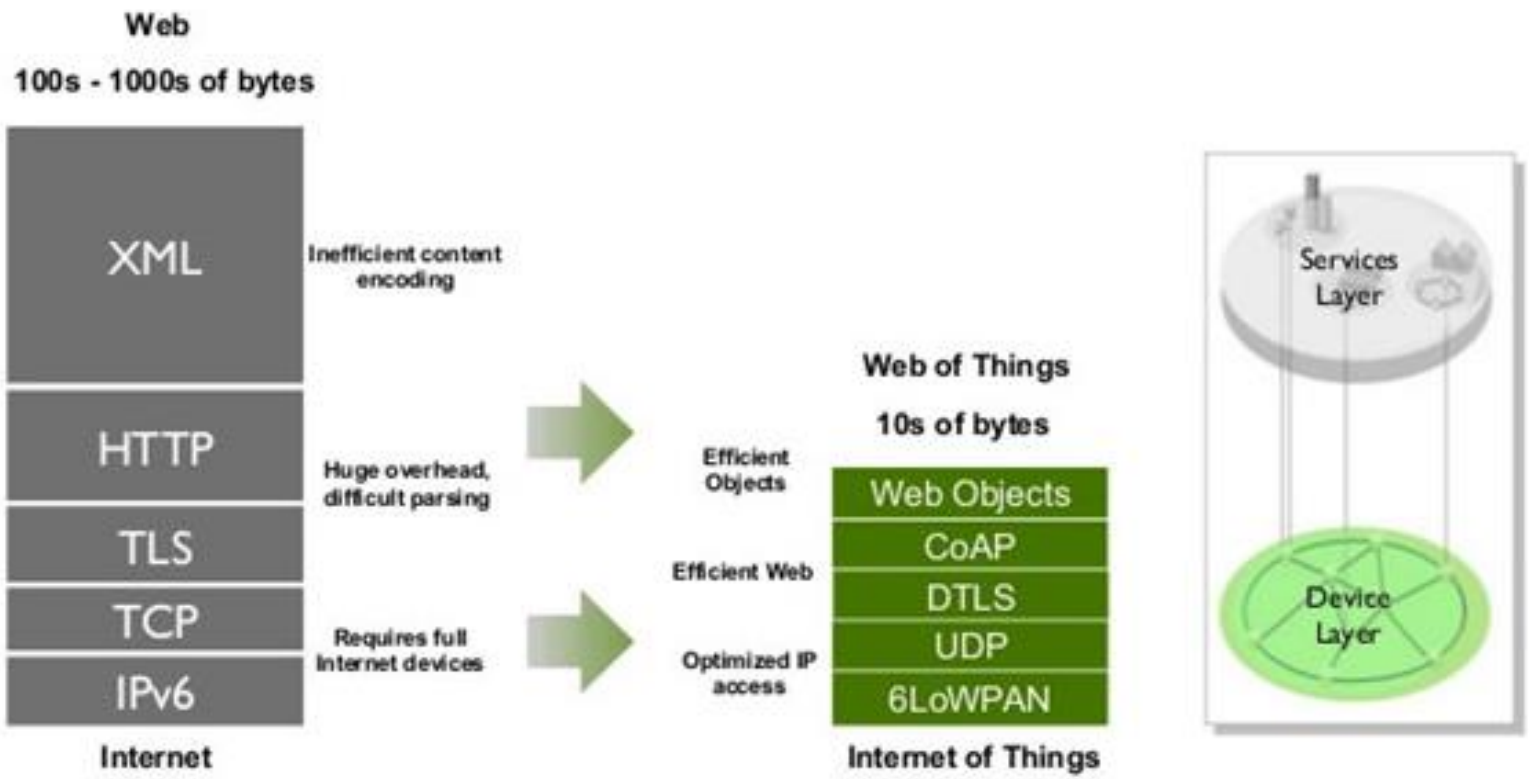
IoT/M2M in 4G and 5G Cellular Networks



IoT Protocols vs Internet Protocols



Internet Protocol Simplification

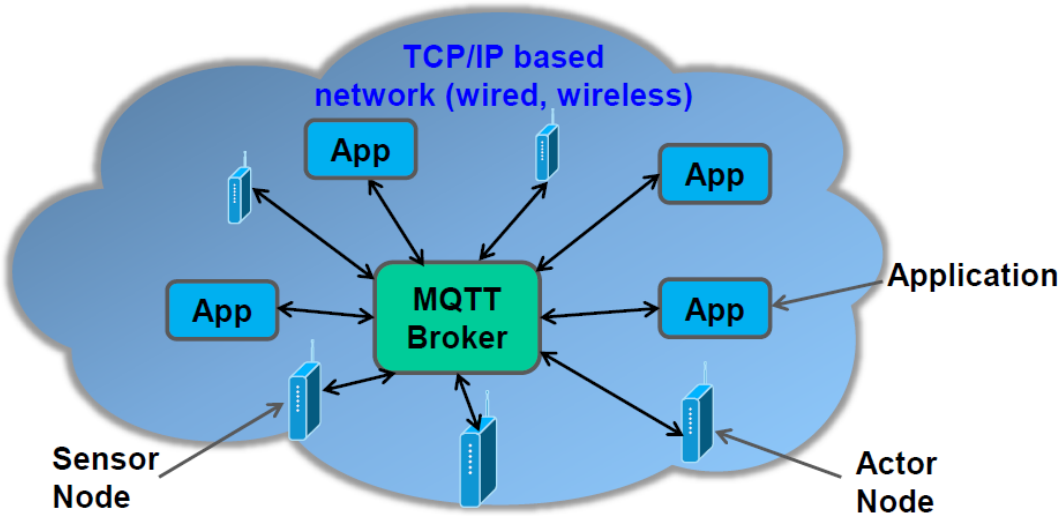


MQTT Protocol Definition

- MQTT (MQ Telemetry Transport), open standard, is a lightweight message queueing and transport protocol suited for the transport of telemetry data (sensor and actor data).

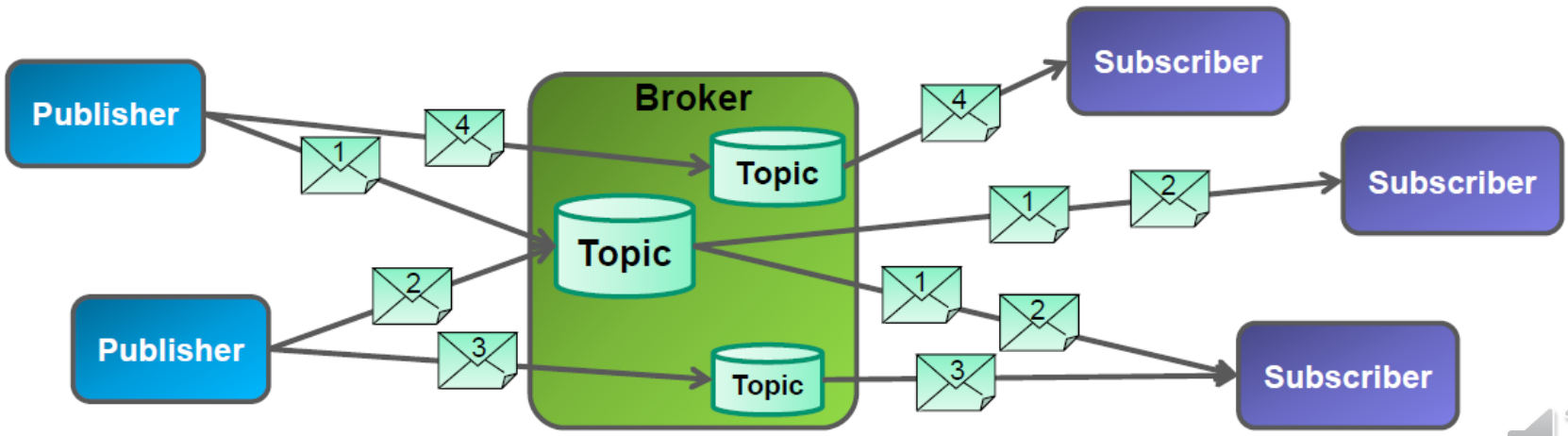
Example:

- Light sensor continuously sends Sensor data to the broker.
- Building control application receives sensor data from the broker and decides to activate the blinds.
- Application sends a blind activation message to the blind actor node through the broker.



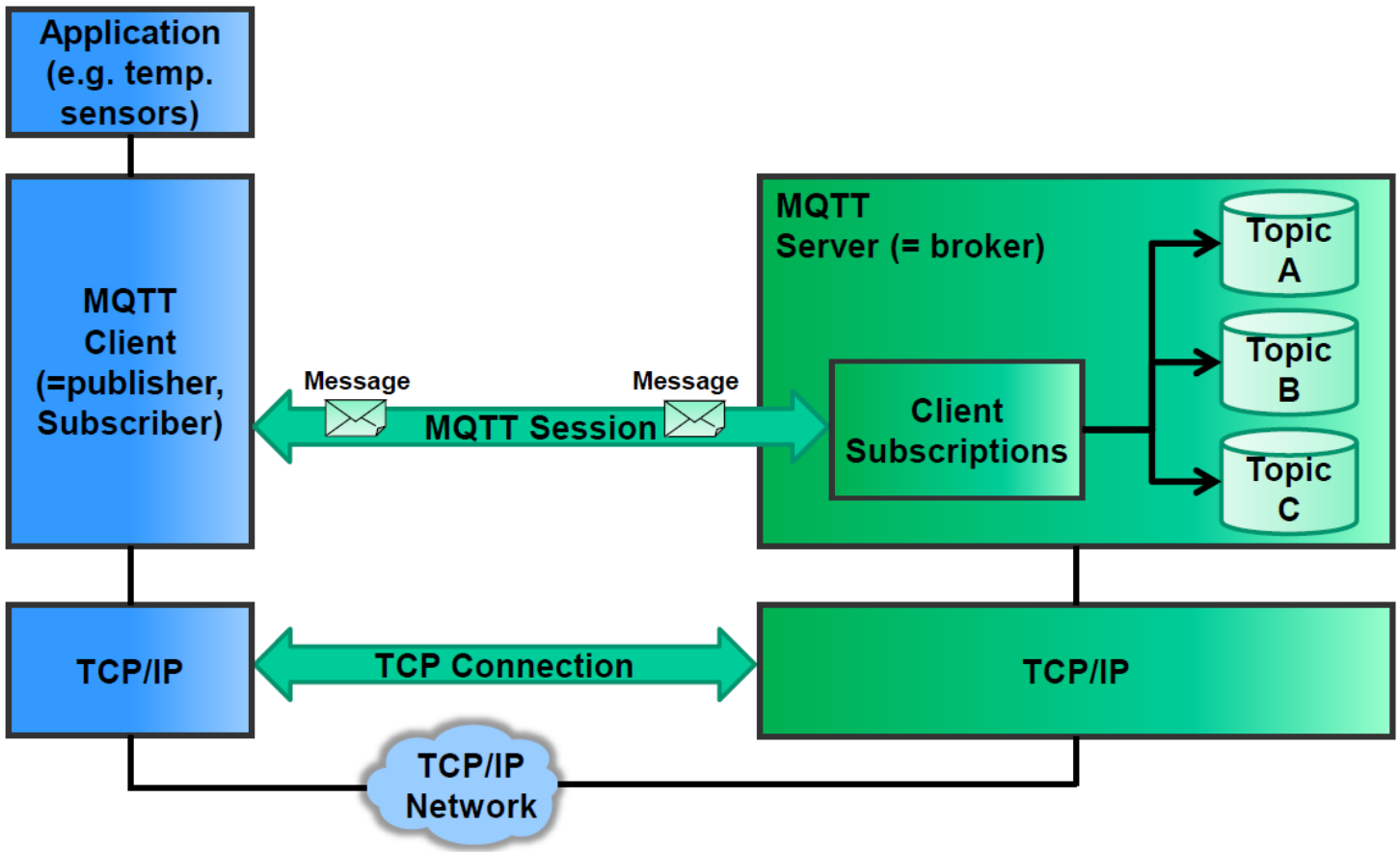
MQTT Protocol Characteristics

- Lightweight message queueing and transport protocol
- Asynchronous communication model with messages (events)
- Low overhead (2 Bytes header) for low network bandwidth applications
- Publish / Subscribe model
- Decoupling of data producer (publisher) and date consumers (subscriber) through topics (message queues)
- Runs on connection-oriented transport (TCP) to be used in conjunction with 6LoWPAN (TCP header compression)



MQTT Model

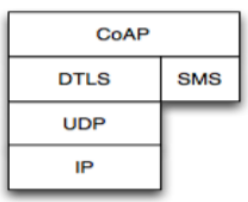
MQTT Core elements: Clients, Servers (=Brokers), Sessions, Subscriptions and Topics

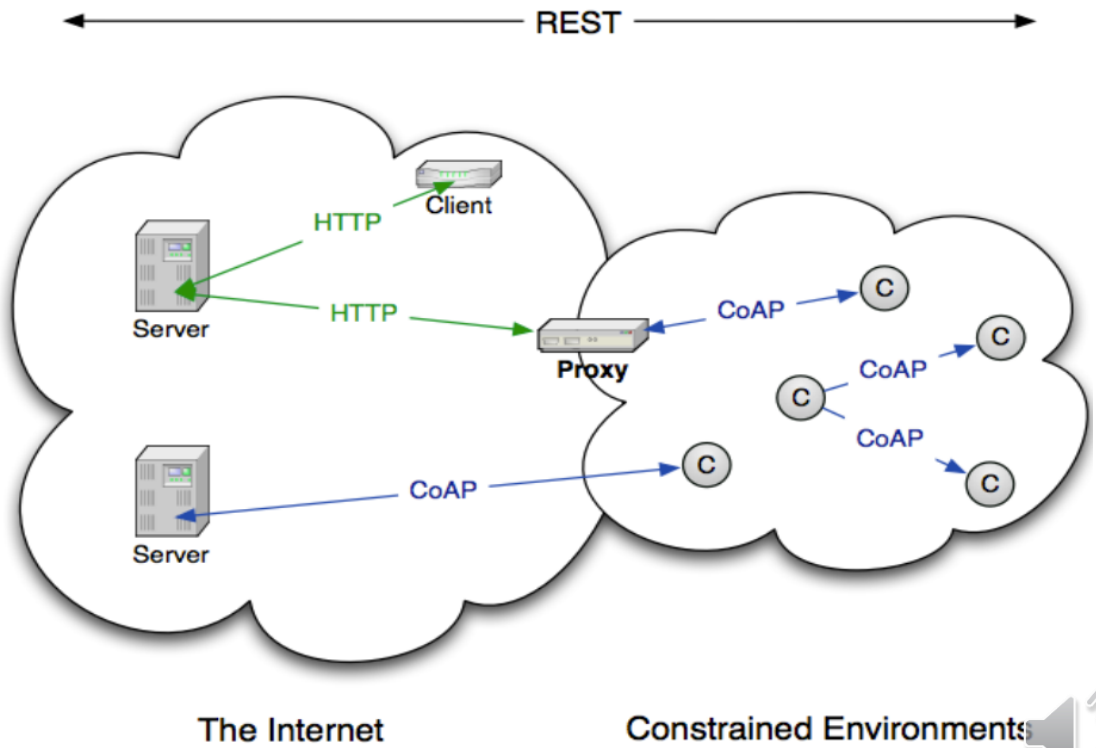


CoAP Protocol Definition

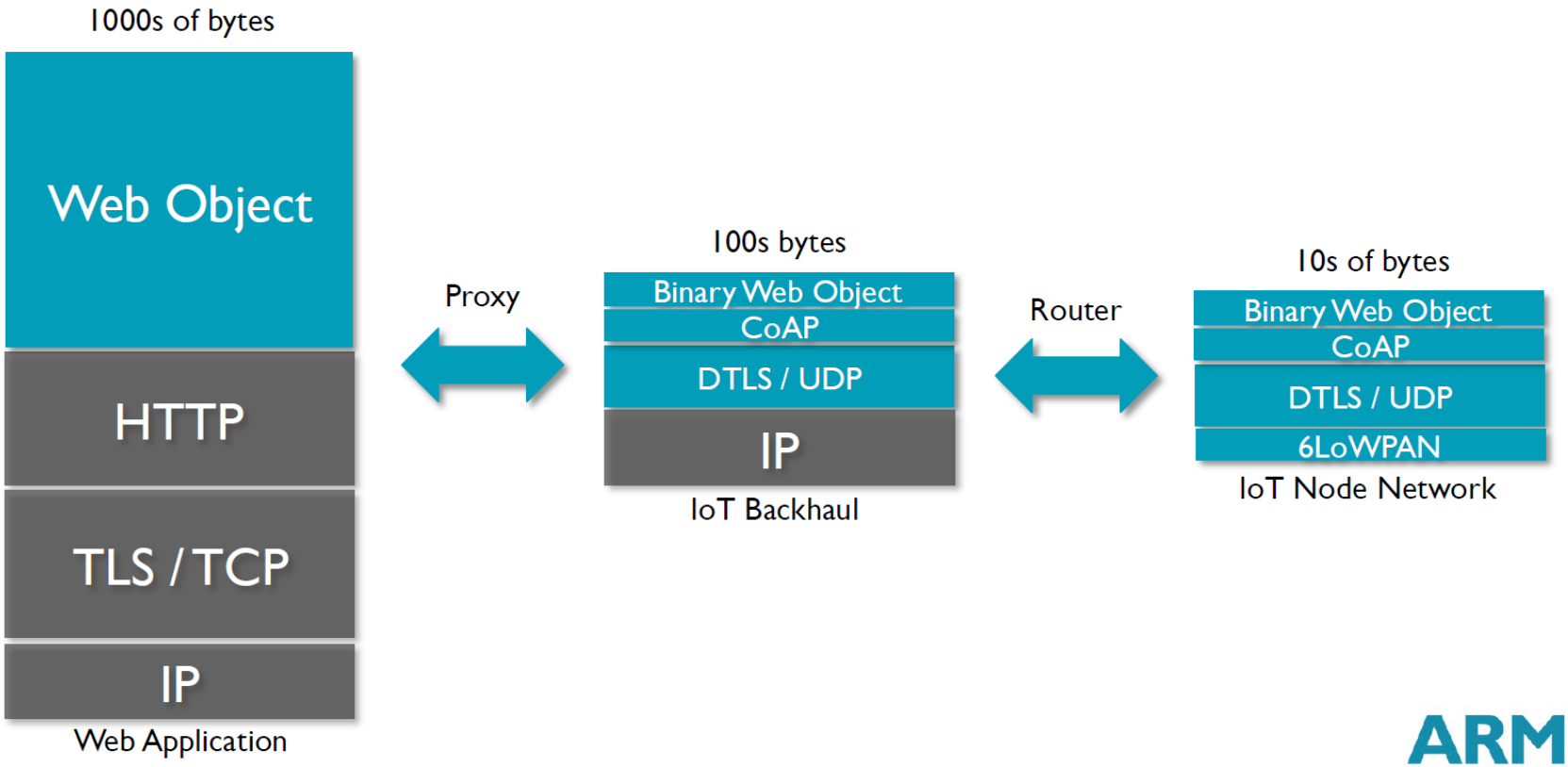
- CoAP (**Co**nstrained **A**pplication **P**rotocol) is developed by the CoRE (Constrained Resource Environments) IETF group.
- CoAP is a document transfer protocol designed for the needs of constrained devices.

- Open IETF Standard
- Compact 4-byte Header
- UDP, SMS, (TCP) Support
- Strong DTLS Security
- Asynchronous Subscription
- Built-in Discovery





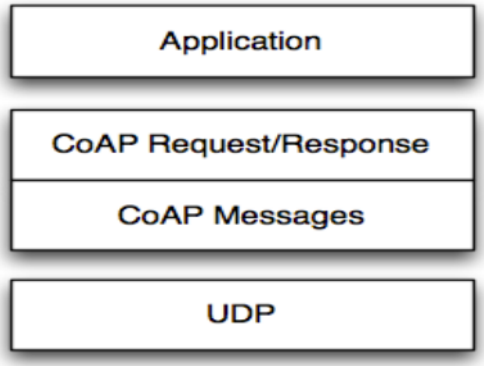
Data Transfer: Nodes - Web Application



ARM

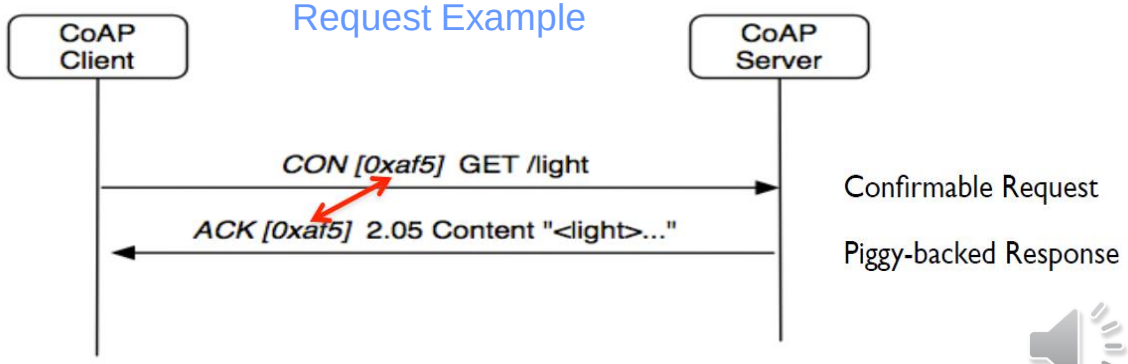
CoAP Features

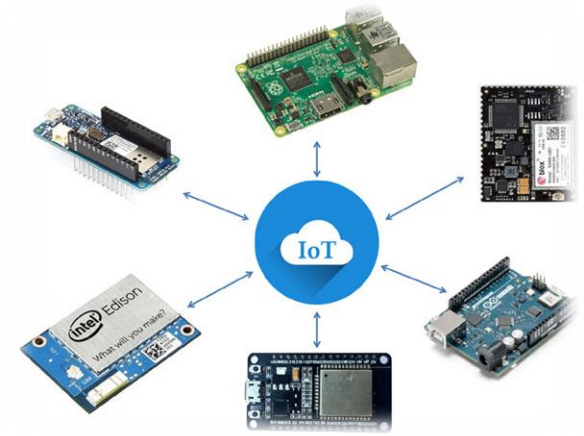
- Embedded web transfer protocol (coap://)
- Asynchronous transaction model
- UDP binding with reliability and multicast support
- GET, POST, PUT, DELETE methods
- URI support
- Small, simple 4 byte header
- DTLS based PSK, RPK and Certificate security
- Subset of MIME types and HTTP response codes
- Built-in discovery
- Optional observation and block transfer



Transaction Model

Request Example

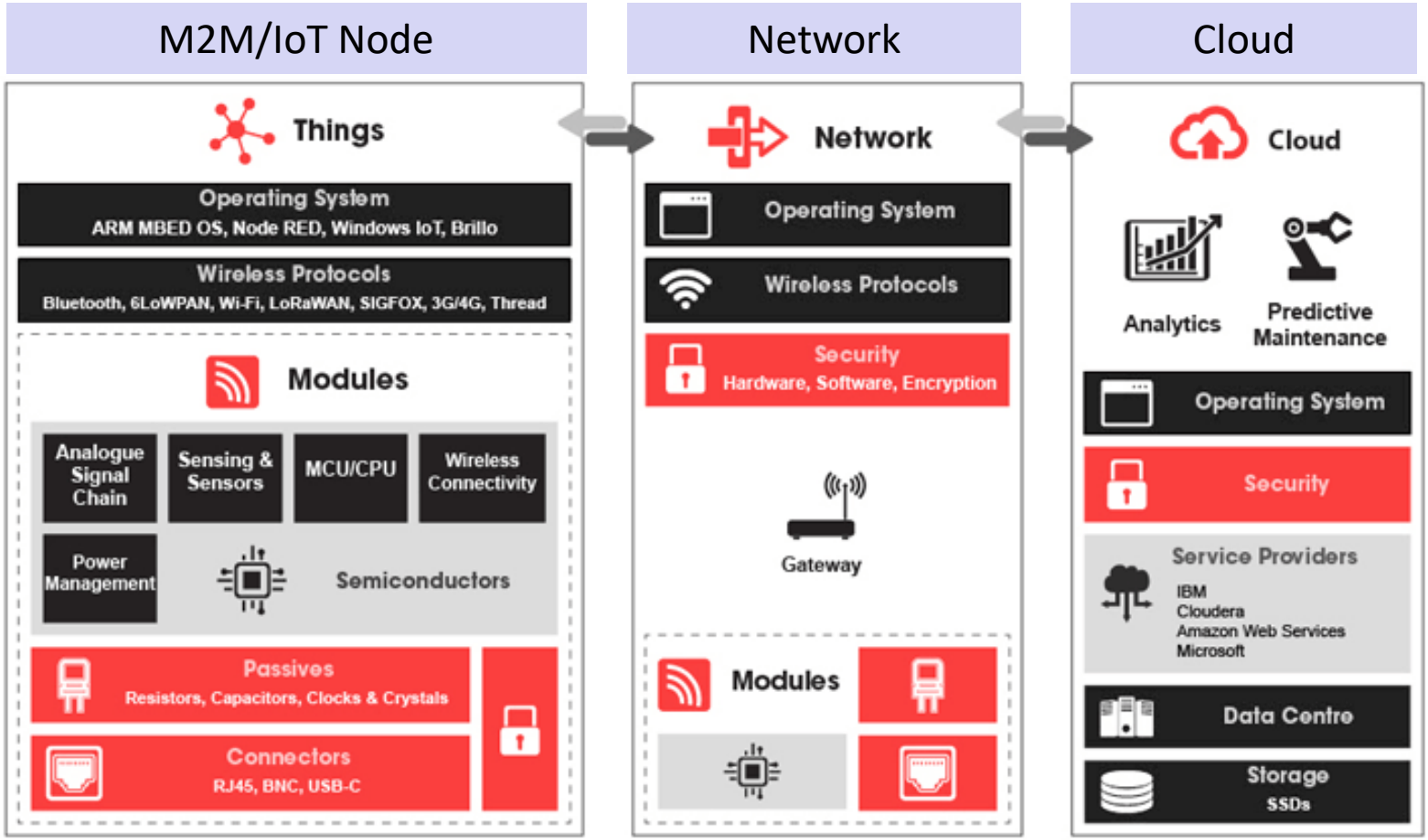




Part I-4

IoT Solution Components & Technologies

IoT Solution Components



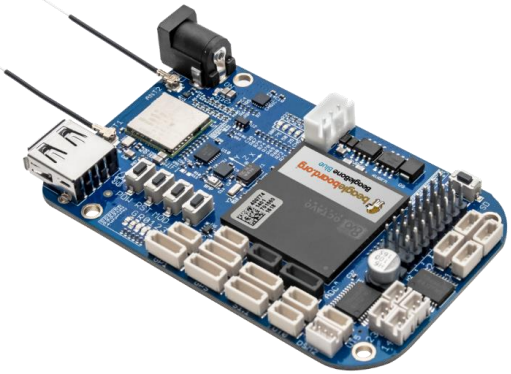
Source: Texas Instruments

IoT Devices Platforms Classification

Open Hardware

Arduino

Raspberry PI



BeagleBone Black

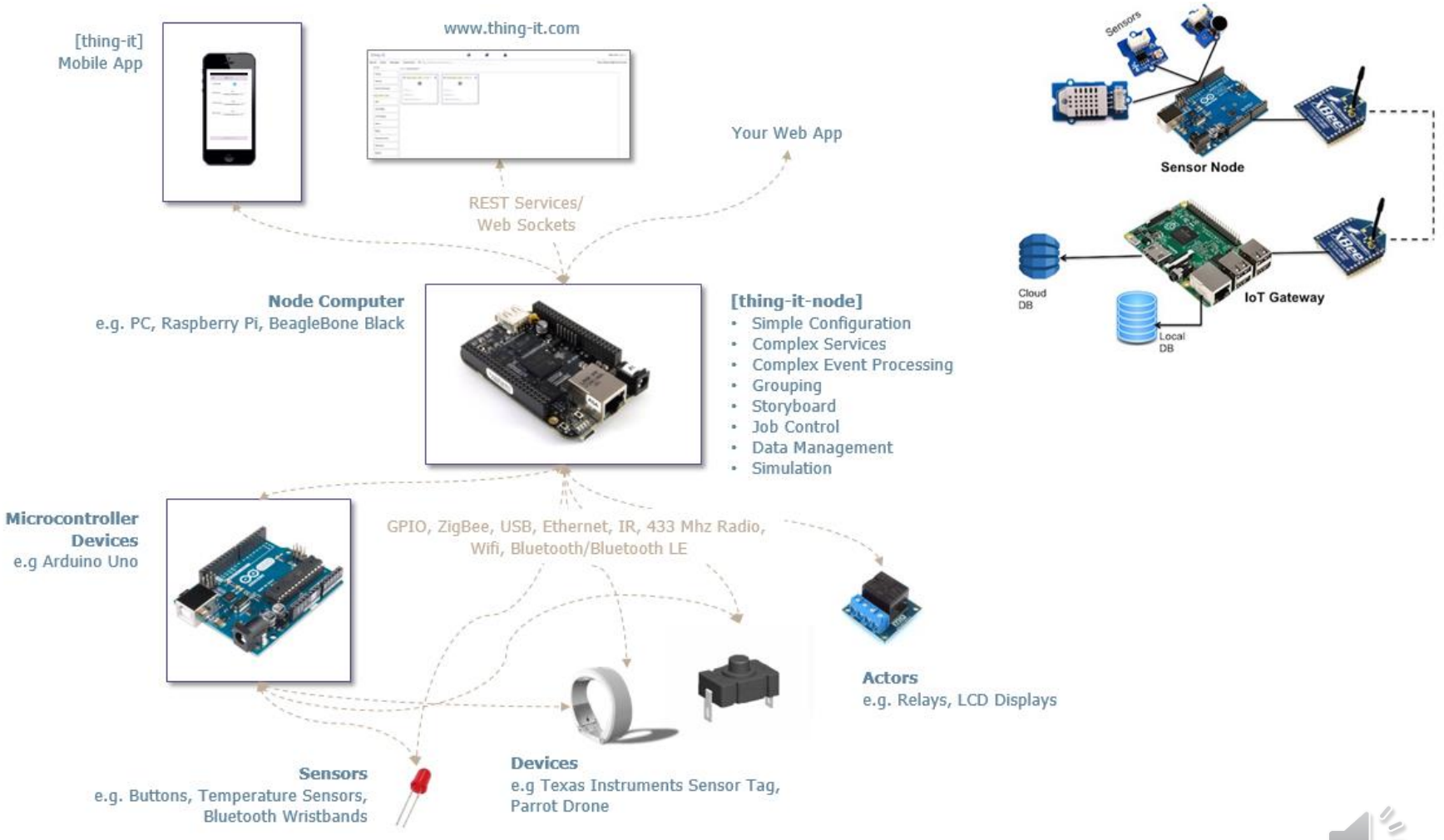
Reference Designs

TI SimpleLink

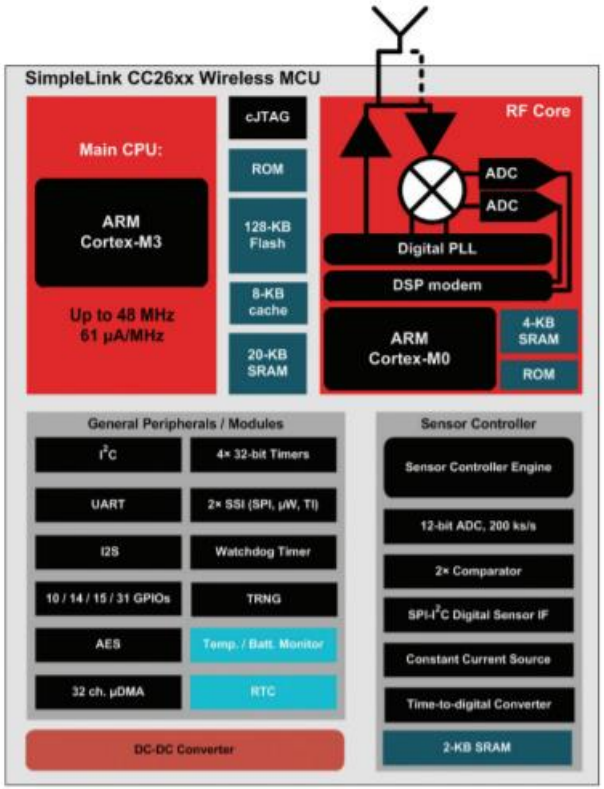
STM32L475 Discovery

Renesas Synergy

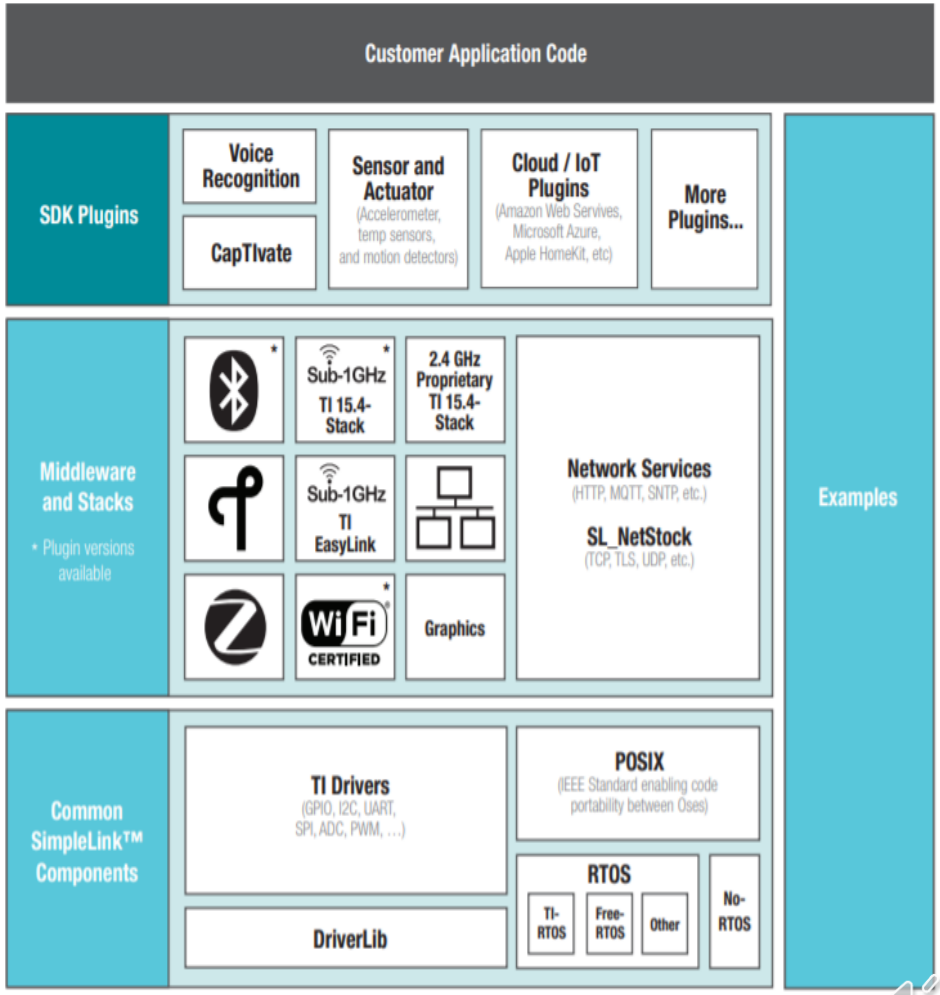
IoT Solution Implementation Example



TI SimpleLink IoT Platform Example

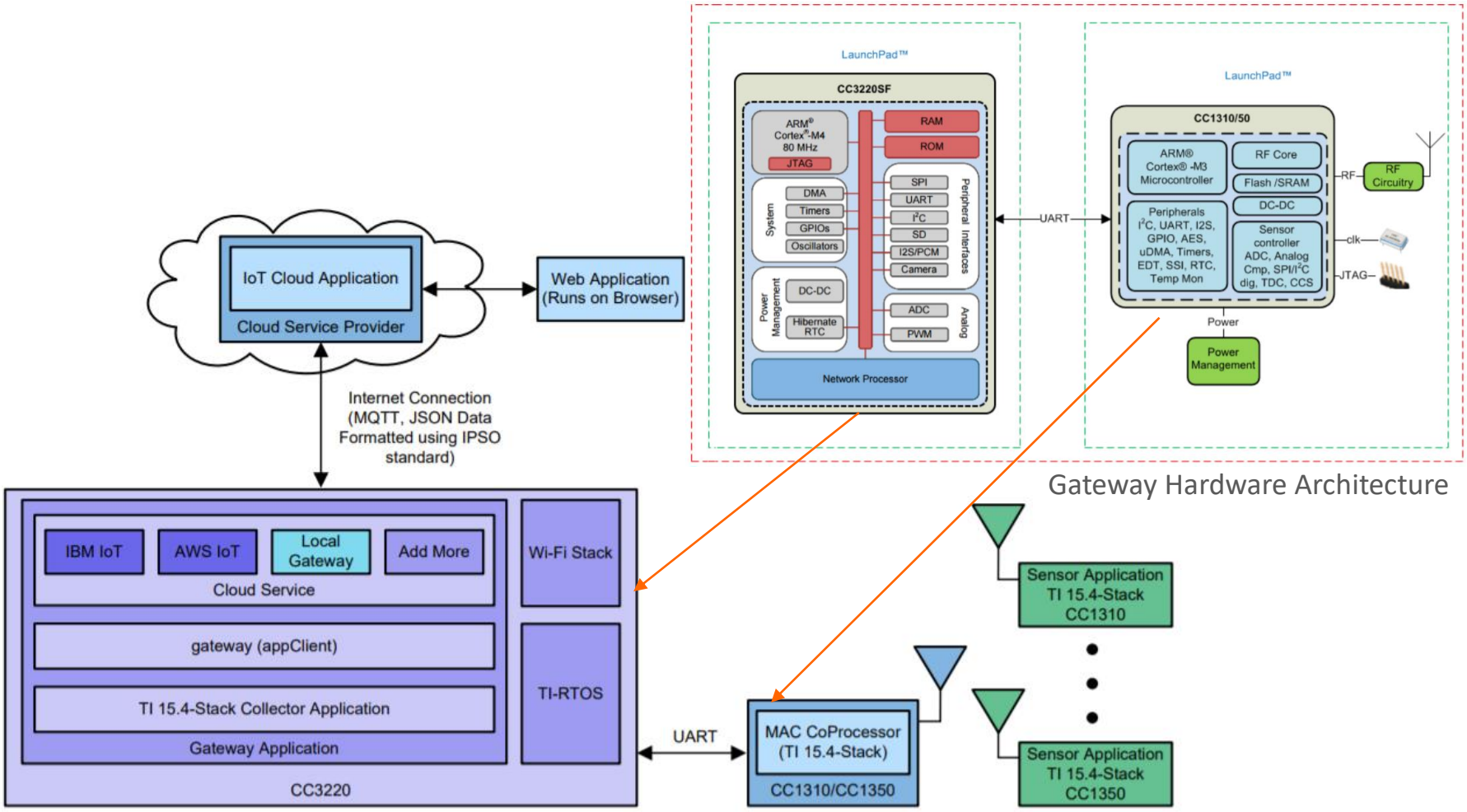


SimpleLink CC26xx wireless MCU



SimpleLink SDK

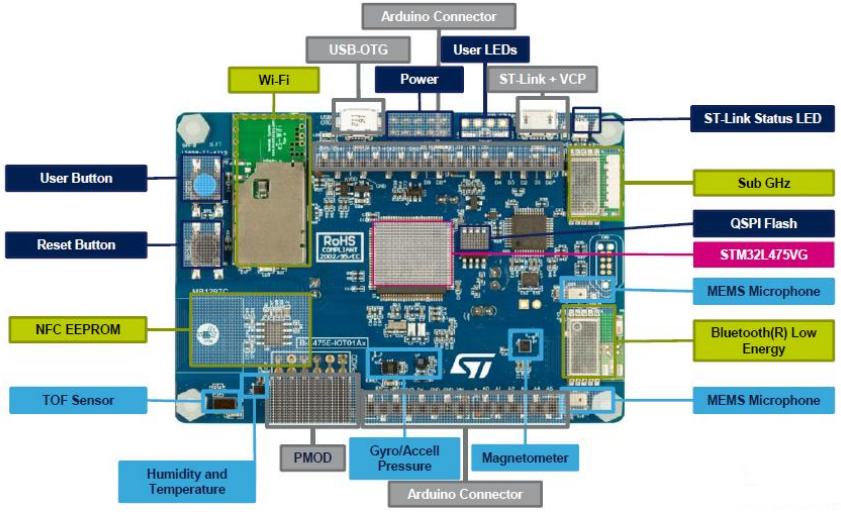
TI Gateway Reference Design

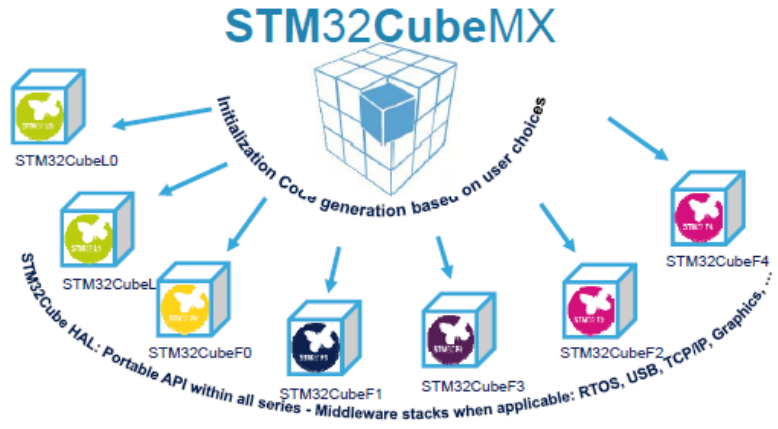


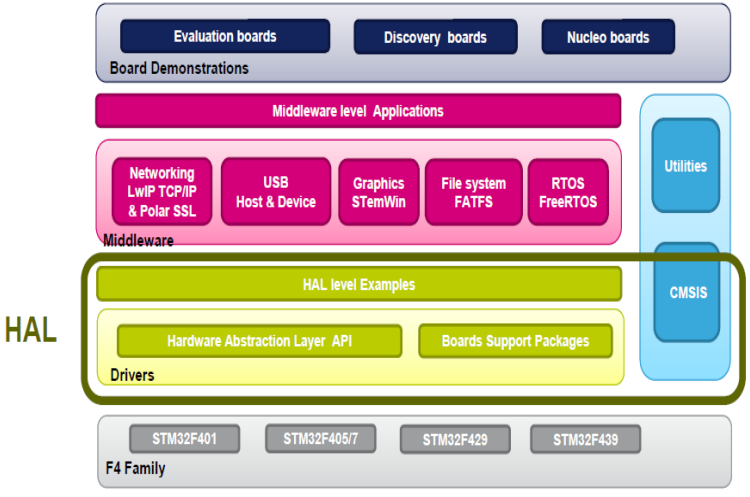
Gateway Hardware Architecture

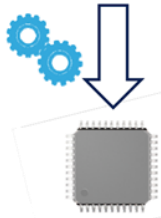
Gateway Software Architecture

STM32L475 Discovery IoT Node Platform





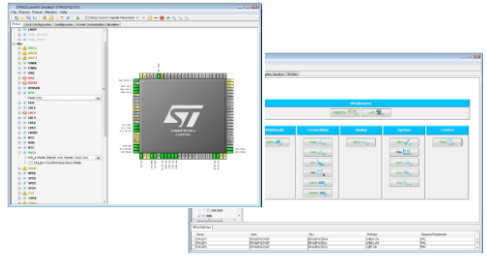




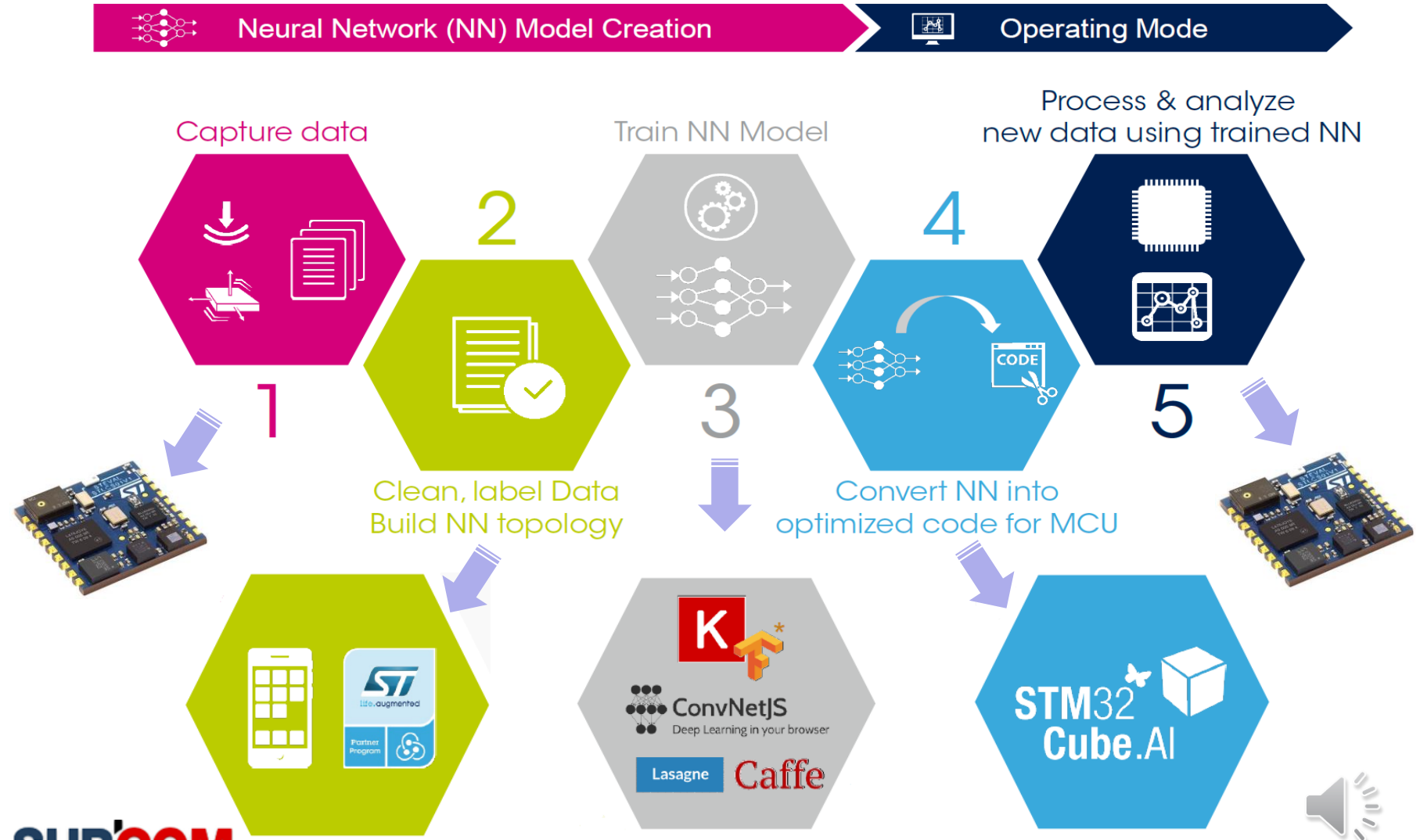
Generates Initialization C Code based on user choices

Abstraction Layer : STM32Cube HAL

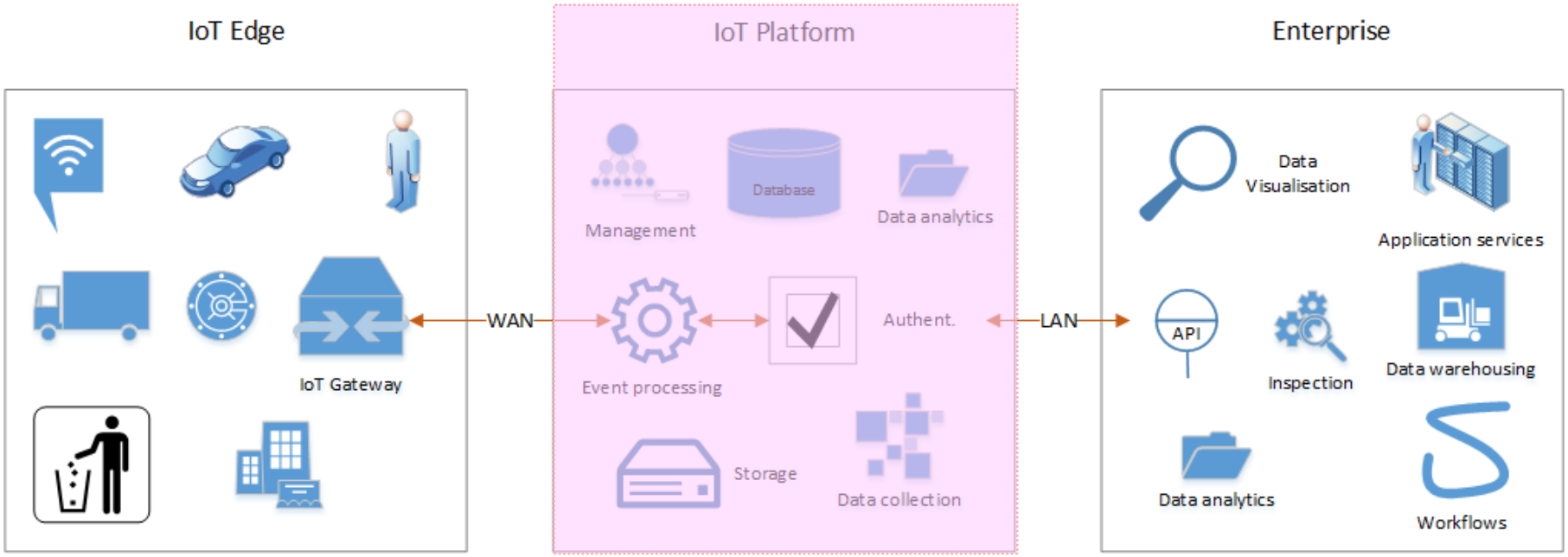
Middleware: RTOS, USB, TCP/IP, Graphics, …



STM32 Cube.AI for Embedded Intelligence



IoT Cloud Platform Definition



Efficient, scalable and affordable way to manage objects and their data.

Typical features:

- Connectivity and network management,
- Device management,
- Data acquisition,
- Analysis, visualization, data processing, storage.

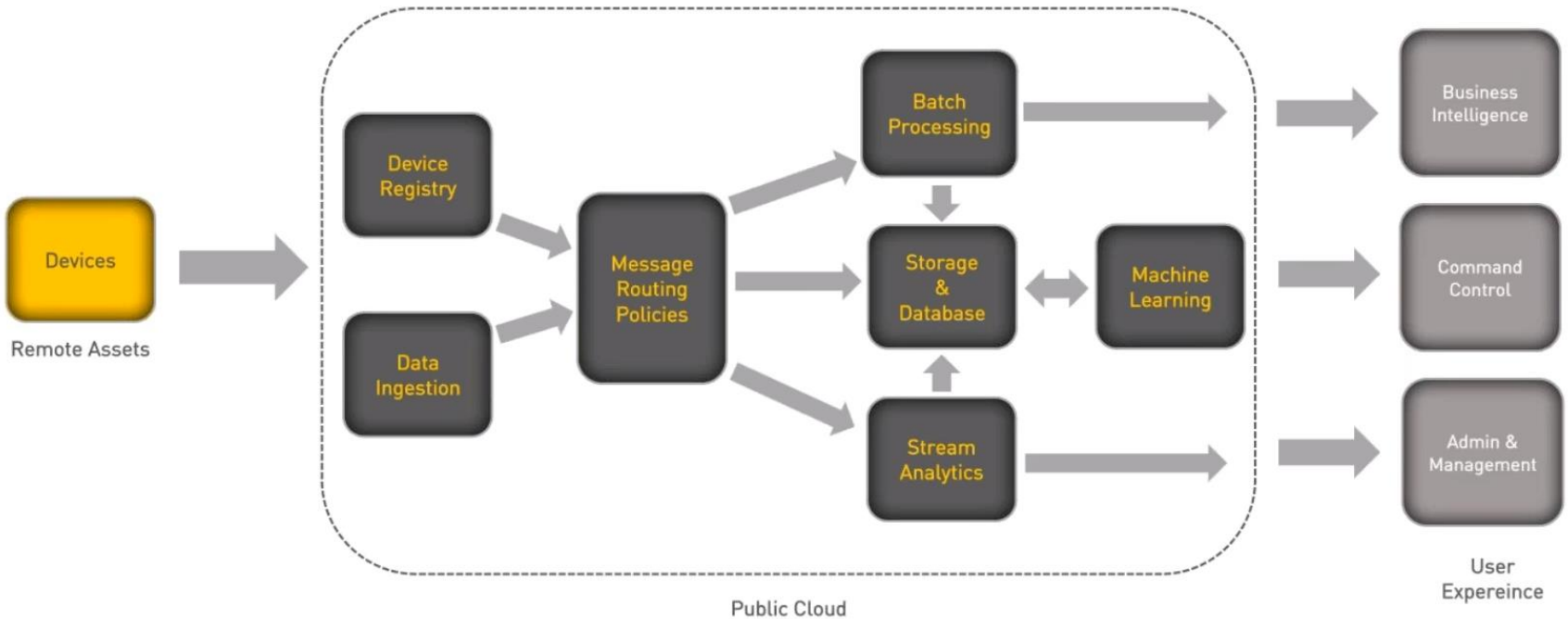
Three types of Cloud IoT:

- **IaaS**: Infrastructure as a Service
- **PaaS**: Plate-form as a Service
- **SaaS**: Software as a Service

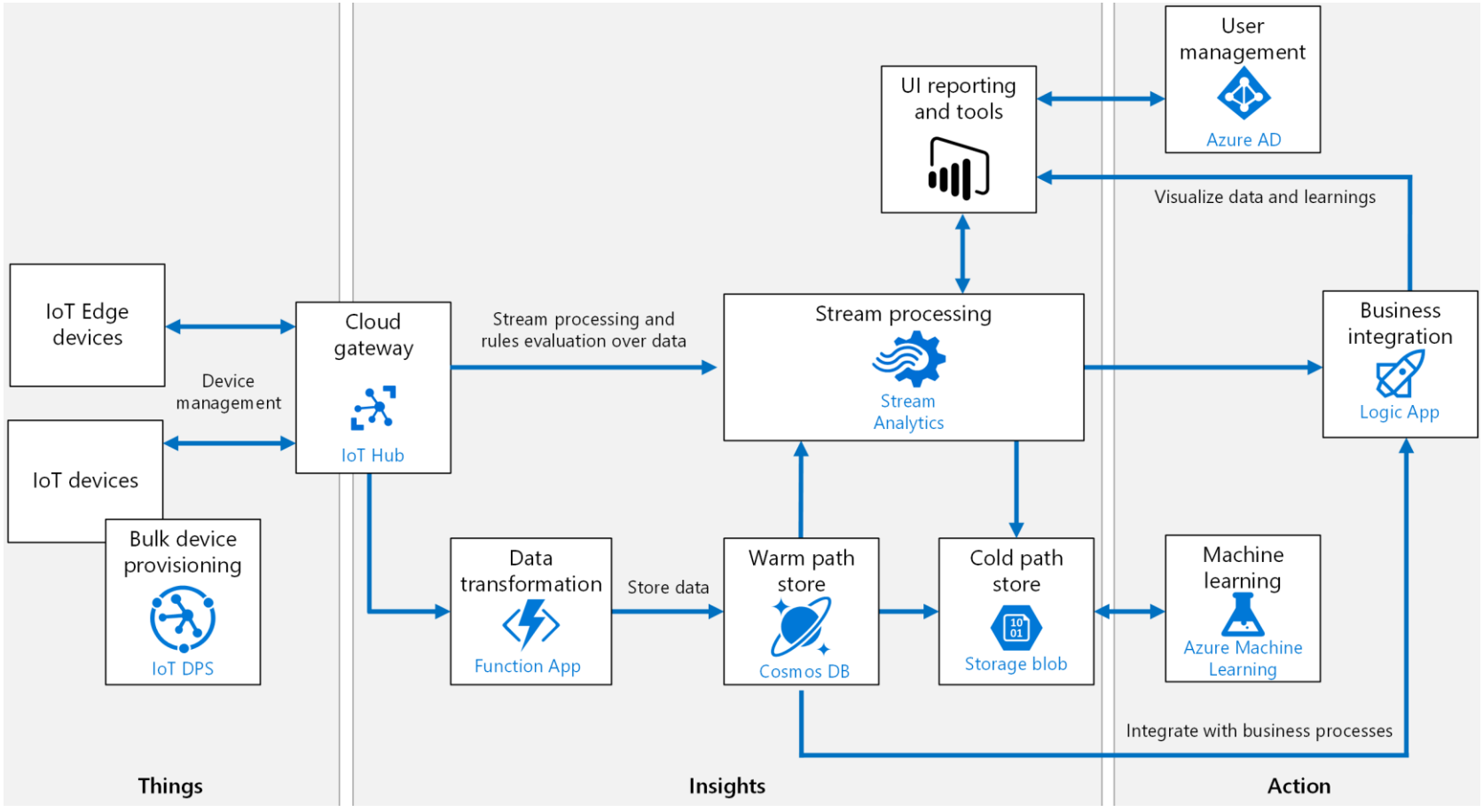
Main Cloud IoT Platforms

- Amazon Web Services IoT
- IBM Watson IoT Platform
- Microsoft Azure IoT Hub
- Google Cloud IoT
- Oracle Integrated Cloud for IoT
- SAP Cloud Platform for the Internet of Things
- Cisco Jasper Control Center
- PTC ThingWorx Industrial IoT Platform
- Salesforce IoT
- Xively
- Carriots

Typical Platform Architecture for Cloud IoT



Microsoft Azure Cloud IoT



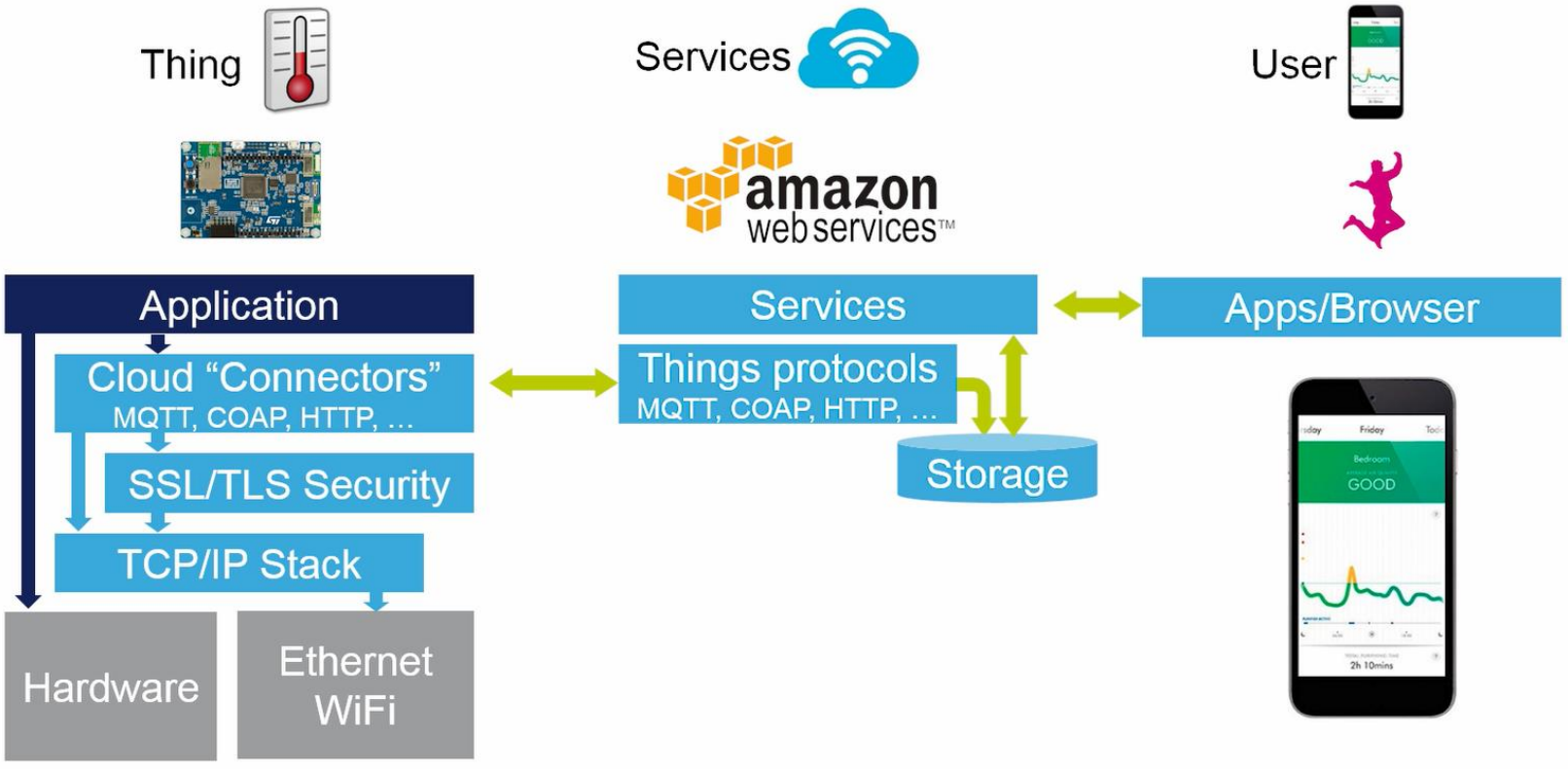
IoT Services of Cloud Azure (1/2)

- IoT Central: It is a SaaS solution that helps you connect, monitor, and manage your IoT devices. To start, you select a template for your device type and create and test a basic IoT Central application that the operators of the device will use. The IoT Central application will also enable you to monitor the devices and provision new devices. This service is for straightforward solutions that don't require deep service customization.
- IoT solution accelerators: It is a collection of PaaS solutions you can use to accelerate your development of an IoT solution. You start with a provided IoT solution and then fully customize that solution to your requirements. You need Java or .NET skills to customize the back-end, and JavaScript skills to customize the visualization.
- IoT Hub: This service allows you to connect from your devices to an IoT hub, and monitor and control billions of IoT devices. This is especially useful if you need bi-directional communication between your IoT devices and your back end. This is the underlying service for IoT Central and IoT solution accelerators.
- IoT Hub Device Provisioning Service: This is a helper service for IoT Hub that you can use to provision devices to your IoT hub securely. With this service, you can easily provision millions of devices rapidly, rather than provisioning them one by one.

IoT Services of Cloud Azure (2/2)

- IoT Edge: This service builds on top of IoT Hub. It can be used to analyze data on the IoT devices rather than in the cloud. By moving parts of your workload to the edge, fewer messages need to be sent to the cloud.
- Azure Digital Twins: This service enables you to create comprehensive models of the physical environment. You can model the relationships and interactions between people, spaces, and devices. For example, you can predict maintenance needs for a factory, analyze real-time energy requirements for an electrical grid, or optimize the use of available space for an office.
- Time Series Insights: This service enables you to store, visualize, and query large amounts of time series data generated by IoT devices. You can use this service with IoT Hub.
- Azure Maps: This service provides geographic information to web and mobile applications. There is a full set of REST APIs as well as a web-based JavaScript control that can be used to create flexible applications that work on desktop or mobile applications for both Apple and Windows devices.

IoT Solution Implementation Example



STM32L475 IoT Node

SUP'COM
Higher School of Communication of Tunis

Thank you for your attention

Send your questions to

adel.ghazel@supcom.tn