Overview of IoT Architectures, Applications OS and M2M Communication

Dr. R. R. Rout,
Associate Professor,
NIT Warangal

Note: All the materials shown in this presentations are only for academic discussions, some of the contents are taken from the Internet

What is IoT?

- The **Internet of Things** (**IoT**) is the network of physical objects—devices, vehicles, buildings and other items embedded with electronics, software, sensors, and network connectivity—that enables these objects to collect and exchange data.

- A Thing can be defined as an embedded device based on a micro controller that can transmit and receive information.

- Interconnection of Heterogeneous Objects (Things)

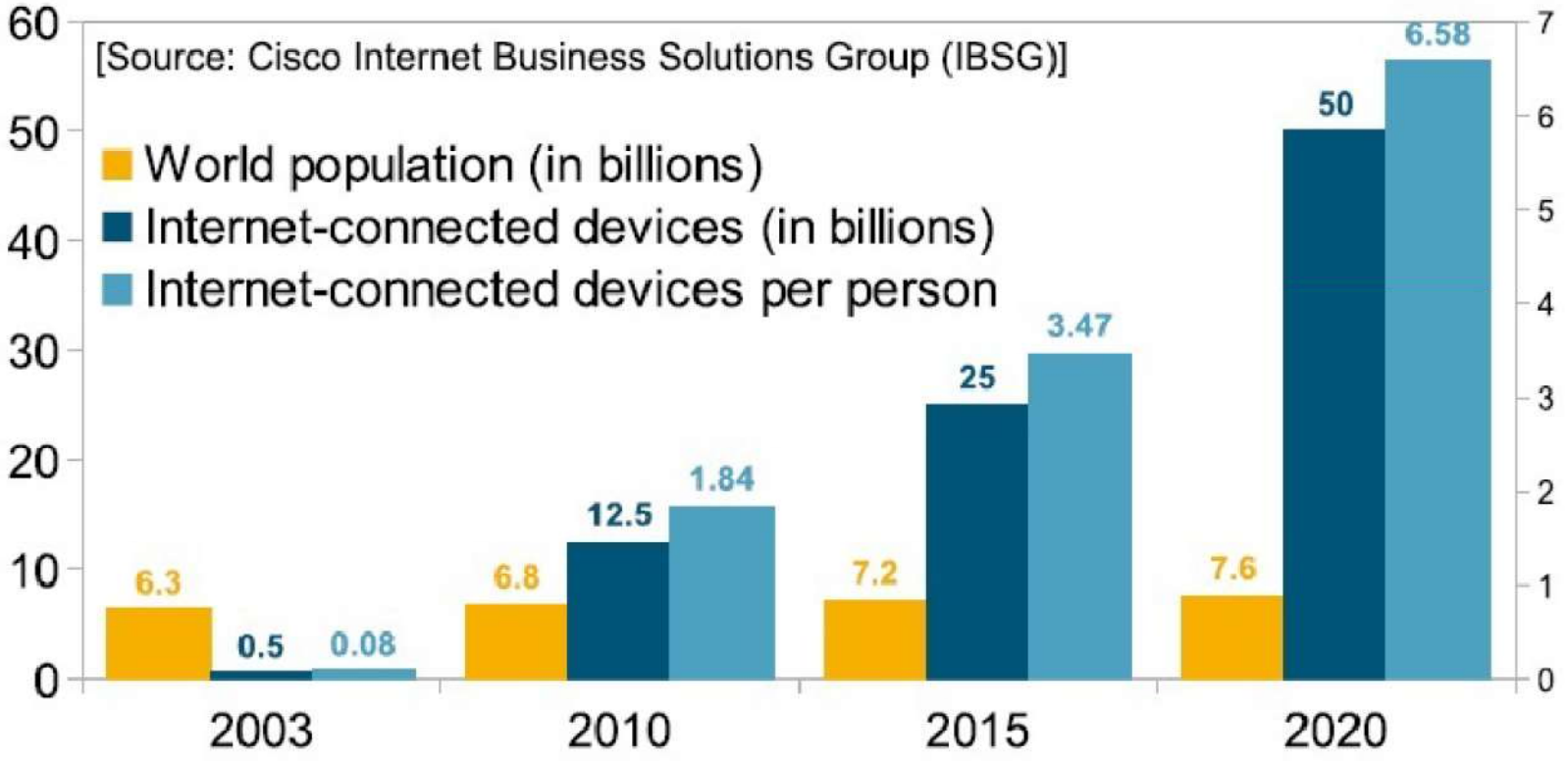
The term “Internet of Things”

Kevin Ashton (born on 1968) coined ***"Internet of Things"*** phrase to describe a system where the Internet is connected to the physical world via ubiquitous sensors

Internet of Things

- Kevin Ashton a British technology pioneer who cofounded the Auto-ID Center at the Massachusetts Institute of Technology (MIT)
  - which created a global standard system for RFID and other sensors

Growth of Interconnected devices



Network Structures

Traditional Network with nodes and links:
(LAN and Internet)

Overlay Network (P2P Network)
(Chord, BitTorrent)

Virtual Tier Networks with Personalized Nodes (Online social Networks)

Importance of IoT

- More things are being connected to address a growing range of business needs. In fact, by 2020, more than 50 billion things will connect to the Internet—seven times our human population.

- Wearable *health* and performance monitors, *connected vehicles*, smart grids, *smart homes* and connected manufacturing.

- This Internet of Things (IoT) will revolutionize the way we work, live, play, and learn.

What Is the Internet of Things?

- The Internet of Things is the intelligent connectivity of physical devices driving massive gains in efficiency, business growth, and quality of life.

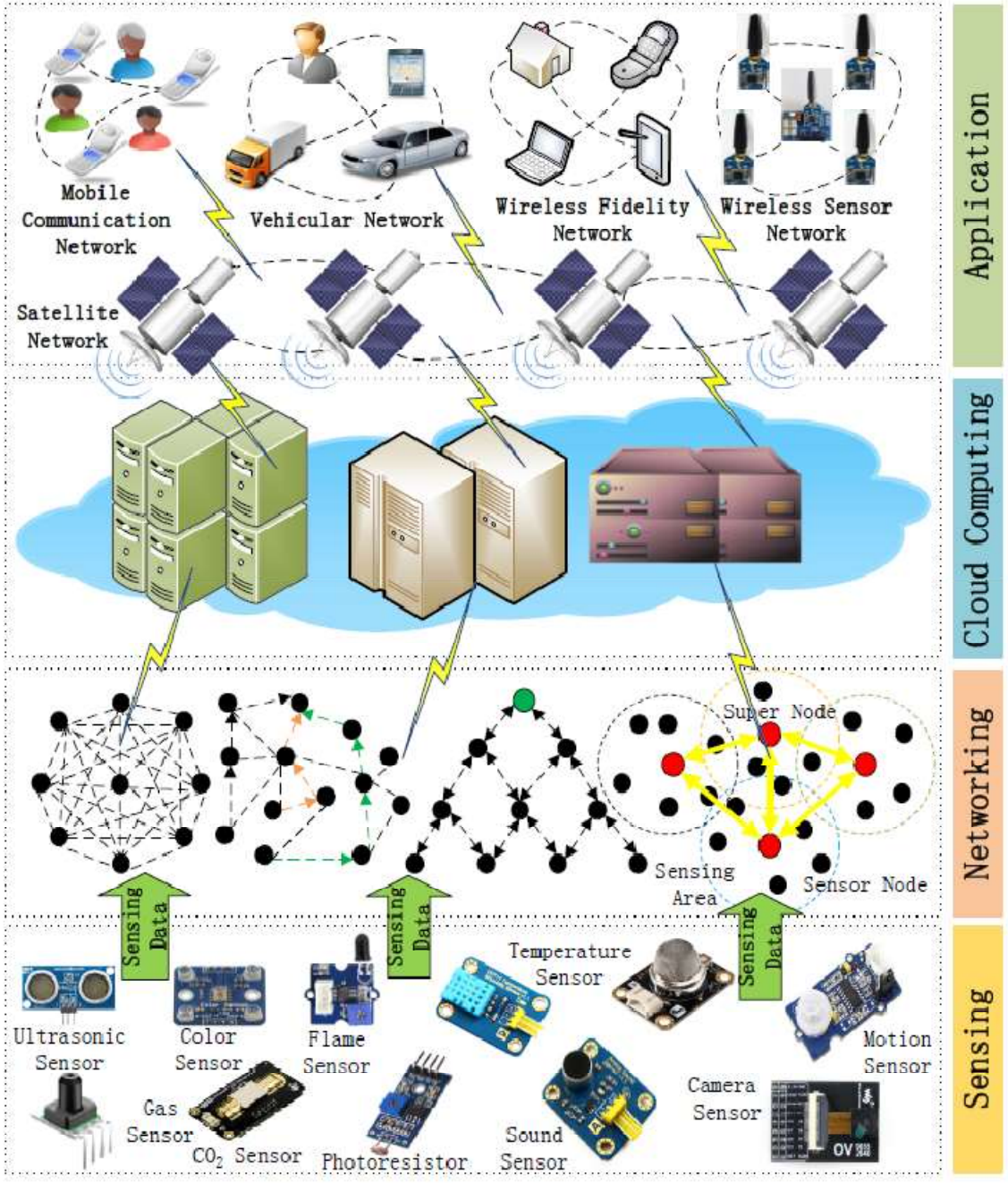
How Ubiquitous?

Gartner: "IoT Installed Base Will Grow to ***26 Billion Units*** By 2020." *That number might be too low.*

- Every mobile
- Every auto
- Every door
- Every room
- Every sensor in every device … in every bed, chair or bracelet ... in every home, office, building or hospital room … in every city and village ... on Earth ...

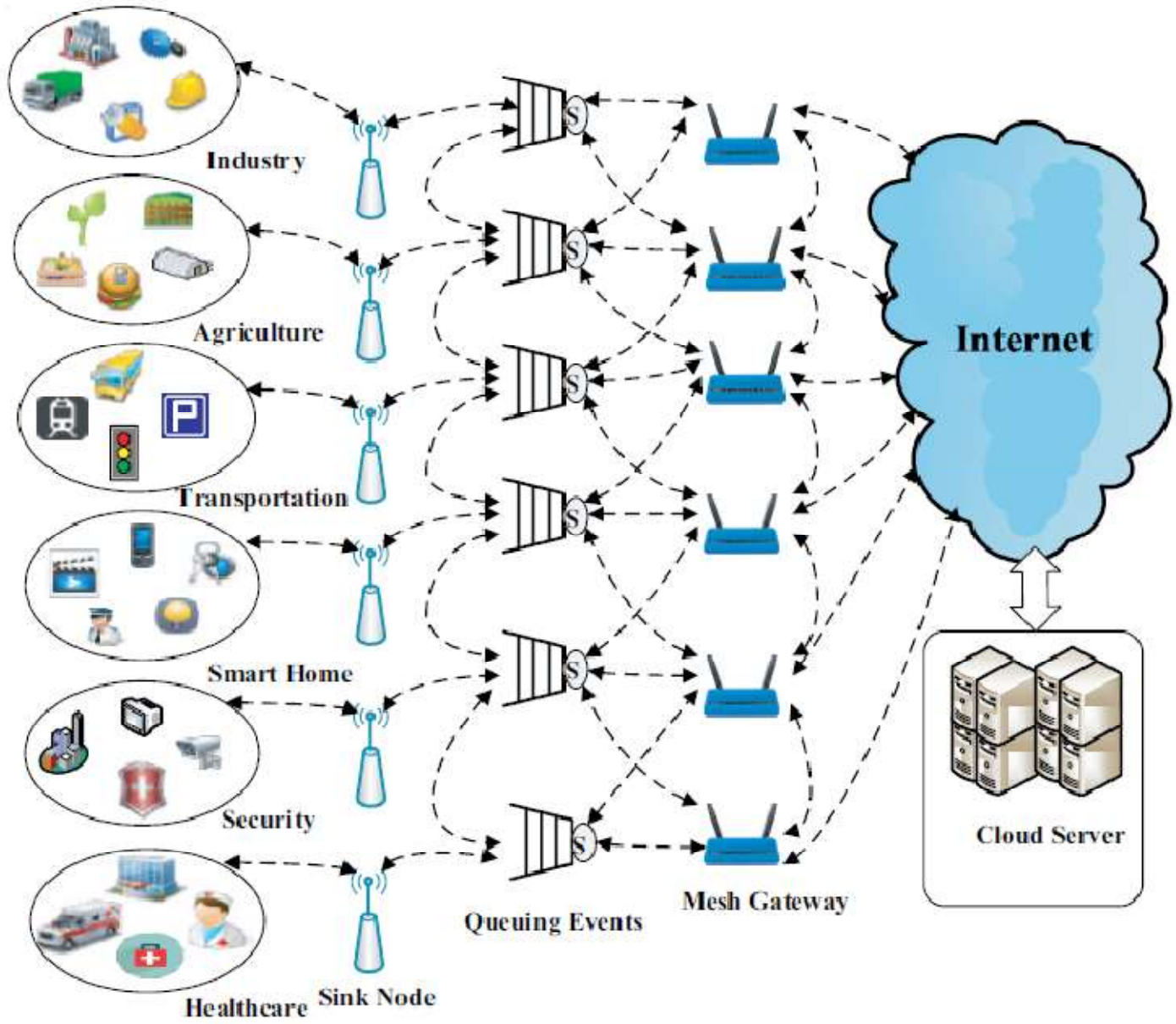
An End-to-End IoT System

- Researchers have designed and developed various hardware and software platforms to support IoT Applications
  - Application oriented research and development
- Heterogeneous IoT (HetIoT) has involved various network architectures
  - WSN
  - Wi-Fi
  - Cellular Wireless (3G/4G/LTE/5G)
  - WMN
  - Vehicular Network.
- Heterogeneous network units employ RFID, sensors and other smart terminals to get the sensing information anytime, anywhere


Application
Mobile Communication Network
Vehicular Network
Wireless Fidelity Network
Wireless Sensor Network
Satellite Network
Cloud Computing
Networking
Super Node
Sensing Area
Sensor Node
Sensing Data
Sensing Data
Sensing Data
Sensing
Ultrasonic Sensor
Color Sensor
Flame Sensor
Temperature Sensor
Motion Sensor
Gas Sensor
CO_2 Sensor
Photoresistor
Sound Sensor
Camera Sensor

IoT System….

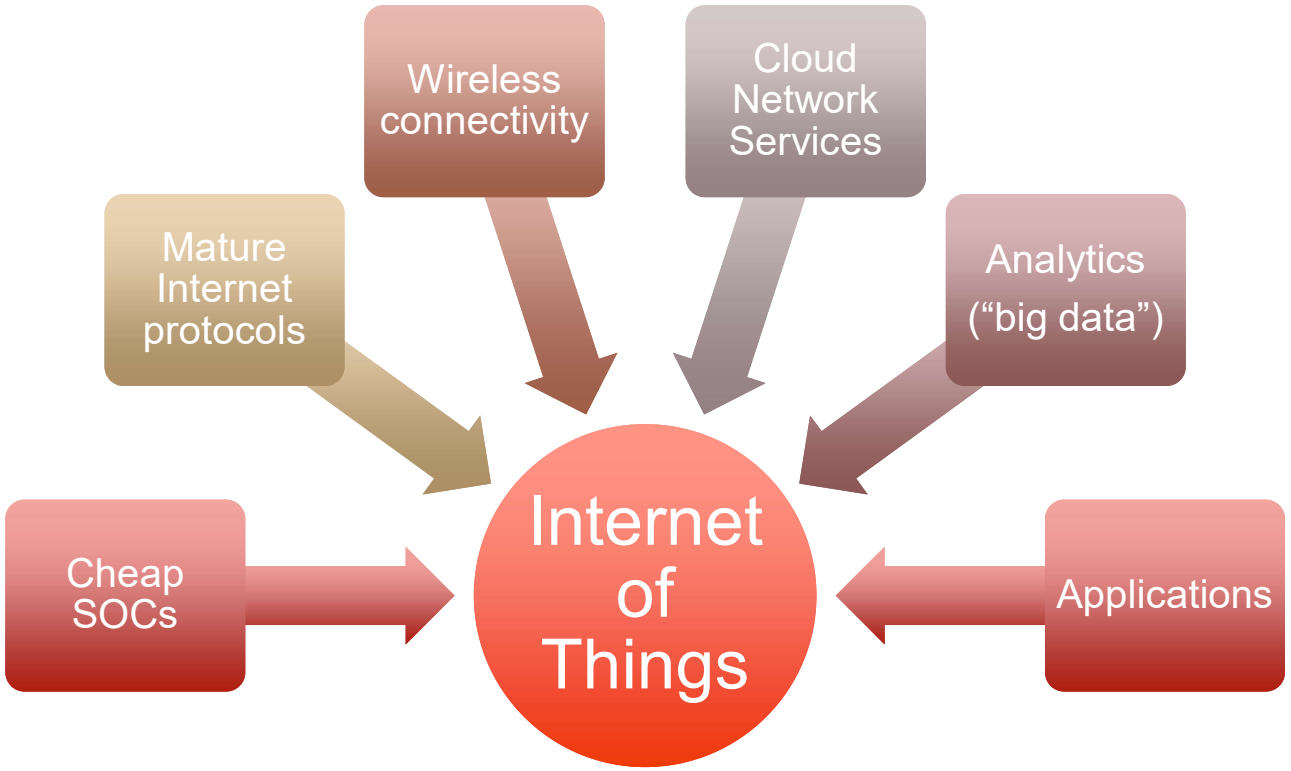
- IoT Devices can be connected to cloud server by Internet or satellites
- IoT devices reliably transmit urgent events and data in real time to a remote monitoring center for processing
- The monitor at the central server intelligently processes and analyzes a large amount of data to achieve the smart control of objects.


Industry
Agriculture
Transportation
Smart Home
Security
Healthcare
Sink Node
Queuing Events
Mesh Gateway
Internet
Cloud Server

Wireless Standards in IoT

- IEEE 802.11
- IEEE 802.15.4
- Cellular Networks (3G/4G)
- IEEE 802.3
- Name of IoT hardware platforms
  - Aurdrino
  - Raspberry
  - Intel
  - Zolertia Z1

Technology Domain for IoT Systems



Emerging IoT Applications

Connected Rail (High Speed Trains)

- Passenger Comfort and web services
- Route Optimization (efficient route finding)
- Critical Sensing (Object detection)

Vehicular Traffic Management

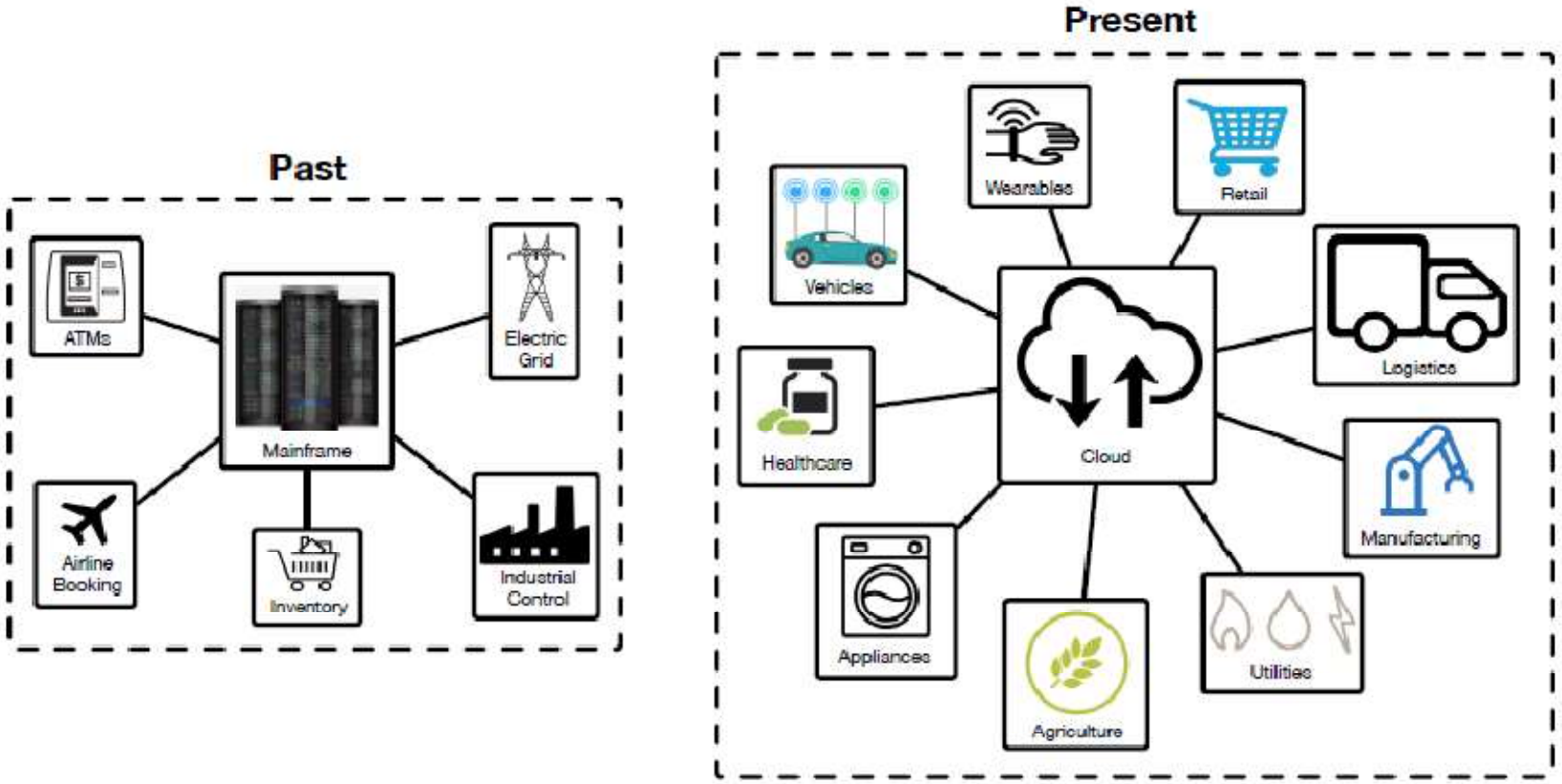
- Connected Traffic Signals
- Vehicle Parking (parking lot management)
- Multimodal City Services (cost/delay…)

IoT in Agriculture

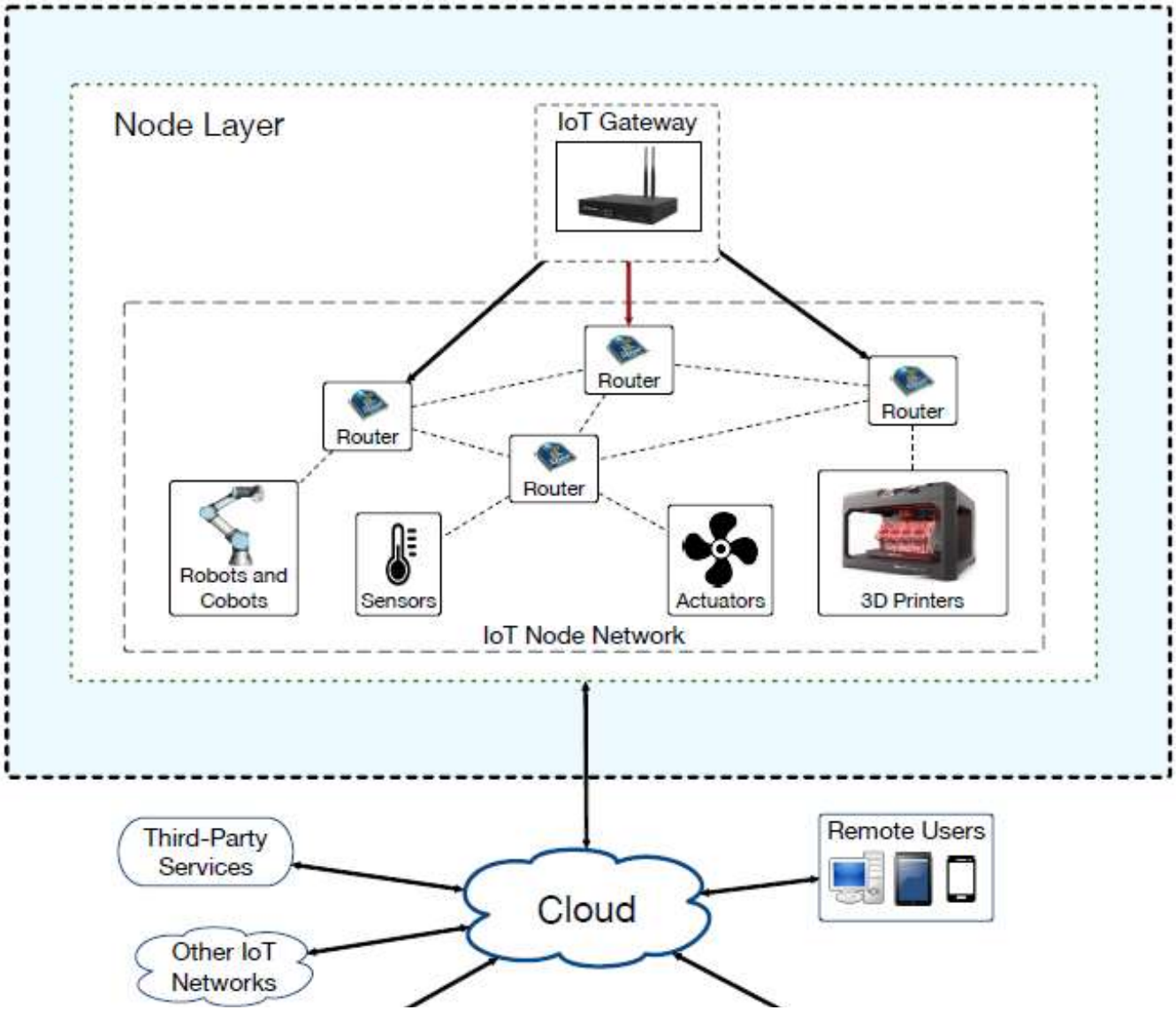
- Precision Farming – Plant & Soil Monitoring
- Agricultural Drones
- Smart Greenhouses (Self regulating, micro-climate controlled environment for optimal plant growth)
- Autonomous & Robotic Labour




IoT Architectures



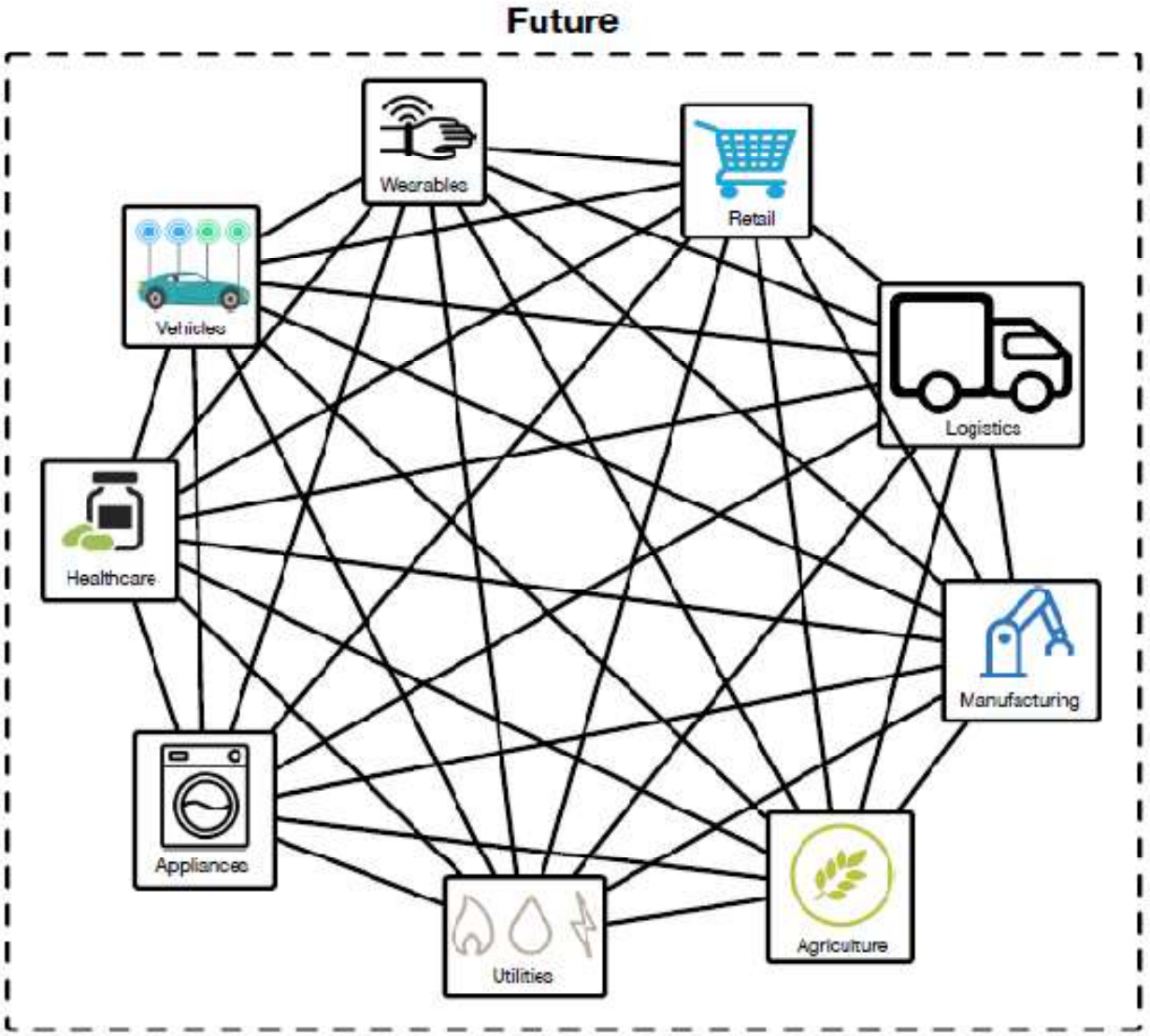
Cloud-Based Architecture



Application specific traffic

- Applications
  - Event Centric
  - Continuous monitoring
  - Priority levels
  - Bandwidth Guarantee
  - Two way/single way communication
  - Local decision vs Global decisions

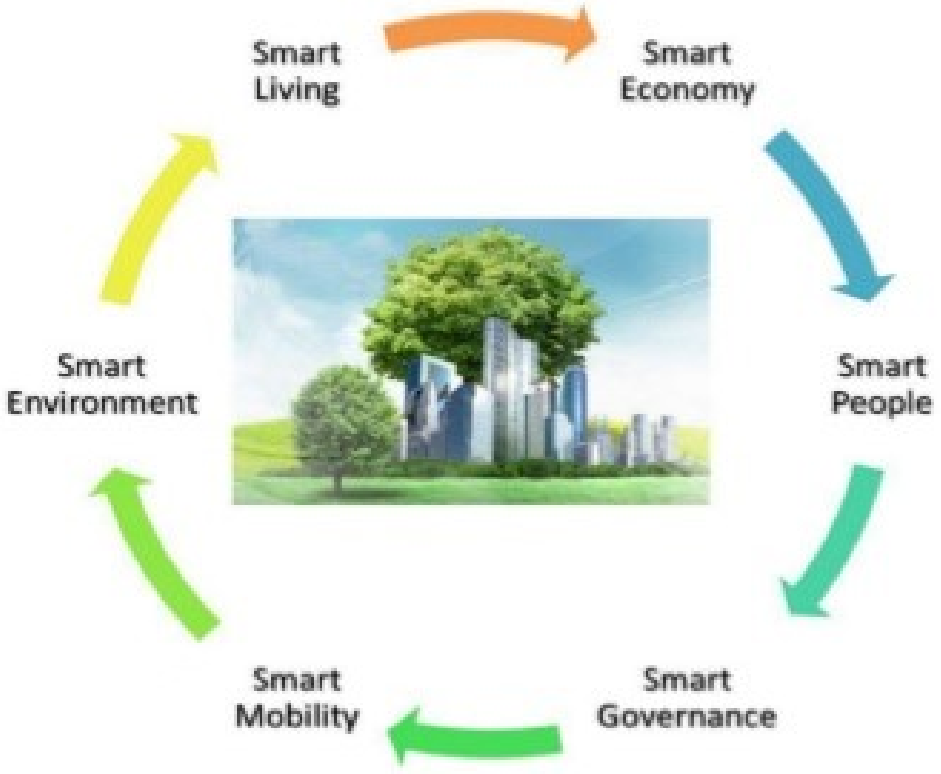
Futuristic IoT Architecture



Defining SDN

- **Software Defined Networking (SDN):**
  - Centralizes command and control in the network
  - Delegates the network flow control decision making to a device with network omniscience
  - Separates the Control plane from the Data plane

Smart city components



Smart City IoT System Requirments

- Monitoring water contamination
- Waste management
- Traffic Congestion monitoring and decision support system
- Air Pollution monitoring
- Smart Grid and metering
- Smart healthcare (Urban/rural)
- Smart parking lot management
- Multi-modal transport
- E-governance systems (MIS/GIS,....)
- Gas filling and management.....many more

OS in IoT Systems

- IoT devices are extremely low in power, memory, and resources.

- An adequate or customized OS with a kernel, networking, real-time capability (memory and processing) and more can make these devices flexible.

OS in IoT Systems

- OSs designed for IoTs devices vary based on
  - Architecture
  - Scheduling methods
  - Networking technologies
  - Programming models
  - Power and memory management methods
  - Other features required for IoTs applications

Contiki

- Contiki was an early development of WSNs, later with a few improvements now it is used on a platform of IoTs
- Since its publication in 2004, it has come far because of its clever features.
- Contiki is licensed under Berkeley software distribution (BSD)

Contiki OS

- Contiki has been developed with a modular architecture style
  - Preemtive Multi-threading
  - C Language.
- Contiki implements Rime as its networking stack and aims for high efficiency in power and memory management.
- Its supports for IPv6 over low power wireless personal area networks (6LoWPAN) applications
- Contiki has already been declared a winner, and at this point in time it is the most frequently used OS for IoTs.

Tiny OS

- TinyOS is called the defacto OS for WSNs.

- TinyOS supports monolithic architecture and an event-driven programming model.
- It has various scheduling techniques and multiple algorithms.
- TinyOS has its own programming language, NesC, originally derived from the C language.
- Efficiency and management techniques for both power and memory
- Open source design

RIOT OS

- RIOT has multiple features, such as its support for 6LoWPAN and real-time scheduling, which make it suitable for IoTs.
- RIOT has come into the spotlight in recent years because it came out in 2013 (only for IoT devices) and was licensed under a less general public license (LGPLv2).
- RIOT has micro-kernel architecture and supports real-time scheduling because of its multi-threading model and its networking support.
- It has been developed in C and C++ programming languages

Nano-RK OS

- Nano-RK is a fully preemptive reservation-based real-time operating system (RTOS) from Carnegie Mellon University with multi-hop networking support for use in WSNs.
- Nano-RK is also famous for its unique method of resource reservation.
- Nano-RK came under the spotlight around 2005.
- Efficient methods of energy and memory management.
- Nano-RK is much like TinyOS because of its monolithic architecture and event-driven system.
- It supports real-time applications and is written in C, making it easier for developers to use.

Lite OS

- LiteOS was developed in 2008 and since then has been fighting for popularity in this area because it has a unique modular architecture and kernel; it uses a programming language known as LiteC++.
- It implements both event and threads in its programming model, however, the scheduling of tasks is “run to completion”.
- It is an open source OS for IoTs and can be considered for various applications in the IoTs field

Mantis OS

- MantisOS is another OS released in 2005 and licensed under BSD
- It is written in C language. It has a method called “comm” for networking support in IoTs.
- It implements resource sharing as semaphores.

Programming Model in Oss in IoT

OSs	Concurrency Support	Programming Model	Programming Language
Contiki	Yes	Hybrid	C
TinyOS	Yes	Event based component model	NesC
RIOT	Yes	Hybrid	LiteC++
Nano-RK	Yes	Multi-Threading	C
LiteOS	No	Event driven	C
MantisOS	Yes	Multi-Threading	C
SOS	No	Multi-Threading	C
RETOS	Yes	Event driven	C

Machine-to-Machine (M2M) Communication

M2M Communication

- Machine to machine (M2M) refers to technologies that allow both wireless and wired systems to communicate with each other devices autonomously.

- M2M communications acts as an enabling technology for the practical realization of Internet-of-Things (IoT).

- No single dominant application, but thousands of embedded applications

  → need low cost to develop & deploy

M2M Communication

- Machine-to-machine (M2M) communications provide ubiquitous connectivity between devices
- Devices have the ability to communicate autonomously requiring no human intervention
- IoT represents a future where billions of everyday objects (including machines) and surrounding environments will be connected and managed through a range of communication networks and cloud-based servers.

Communication Pattern

- Various types of communication may exist within the IoT, and these include D2D, device to human and vice versa, and device to distributed storage.

- Communication could be within the same network (intradomain) or across heterogeneous networks (interdomain)

- In addition, D2D communication can be with or without human intervention.

Communication Pattern

- Communication could be also single hop or multiple hop.
- In the single-hop communication, devices communicate with each other via a network infrastructure, which could be an access point or a base station.
- For multi-hop communication, devices relay information for each other to achieve end-to-end communication between any source and destination device.

Applications

- Some of the most prominent M2M application areas include security and public safety (surveillance systems, object/human tracking, security alarms, etc.)
  - Smart grids (grid control, industrial metering, demand response), vehicular telematics (with enhanced navigation)
  - Healthcare (telemedicine, remote diagnosis, etc.),
  - Manufacturing (production chain monitoring), and remote maintenance (industrial automation, vending machine control, etc.)

Classification of M2M

- M2M networks can be divided into two broad domains:

  1) *Capillary M2M and* 2) *Cellular M2M networks*.

- *In capillary M2M networks*, *M2M* devices form a device area network wherein connectivity is provided through short-range communication technologies (such as ZigBee and Wi-Fi).

- Wide area connectivity is provided through a gateway.

Classification of M2M

- Capillary M2M networks are generally characterized by a huge number of low-cost and low-complexity devices
  - Requirements of high energy efficiency and reliability

- In cellular M2M networks, M2M devices are equipped with embedded SIM cards and have the ability of communicating autonomously with the cellular network like a normal user equipment.

- Cellular M2M has unique characteristics of small data transmissions, mostly mobile-originated (uplink) traffic, little or no mobility of devices, service requirements of high energy efficiency, etc.

IoT Research and Development

- Lightweight protocols for devices to work together, communicate
- Unique and extensible identifiers for all those billions of devices
- Demand for API access and interoperability
- Cybersecurity
- Privacy and Policy

Important References

- *F. Javed, et al.* "Internet of Things Operating Systems support, Networking Technologies, Applications and Challenges: A comparative Review" IEEE Communication Surveys and Tutorials": DOI 10.1109/COMST.2018.2817685, early access article, 2018
- *Tie Qui et al.*, "How Can Heterogeneous Internet of Things Build our Future: A Survey" DOI 10.1109/COMST.2018.2803740, early access article, 2018

Thank You!