UNIT

1

Internet of Things

Unit 1. Introduction of IoT

Dr. Kiran Wakchaure
Assistant Professor
Sanjivani University

Session Content

- Understand basics of IoT

Session Outcome

- Able to understand IoT and its advantages.

Course Outcomes

Course Outcomes: Upon completion of the course, students shall have the ability to		
CO1	Identify and describe the basic components and architecture of IoT systems.	U
CO2	Explain the various communication protocols used in IoT and their applications.	U
CO3	Develop simple IoT applications using Arduino and Raspberry Pi, integrating sensors and actuators.	AP
CO4	Analyze the data flow and network requirements for different IoT applications.	A
CO5	Evaluate the security and privacy challenges in IoT systems and propose potential solutions.	E

Teaching -Learning & Assessment Scheme

Teaching -Learning & Assessment Scheme								
Learning Scheme			Credits	Assessment Scheme			Summative Assessment	Total
				Formative Assessment			End Semester Exam/ Project Evaluation	
L	T	P		CIA-I	CIA-II	Practical Assessment		
3	0	0	3	25	25	20 (100 Scaled Down to 20)	30 (60 Scaled Down 30)	100

Course Contents

Course Contents		
UNIT I	**Introduction to IoT and Networking Basics**	**09 hrs**
Introduction to IoT, History and Evolution of IoT, IoT Architecture, Sensing, Types of Sensors, Sensor Characteristics, Actuation, Types of Actuators, Actuator Mechanisms, Basics of Networking, Network Models (OSI and TCP/IP), Network Topologies		
UNIT II	**Communication Protocols and Sensor Networks**	**9 hrs.**
Communication Protocols and Sensor Networks, Communication Protocols, IoT Communication Models Protocols (MQTT, CoAP, AMQP), Sensor Networks, Wireless Sensor Networks (WSN), Network Topologies in WSN, Energy Efficiency in WSN, Machine-to-Machine Communications, M2M Architecture, Applications of M2M		
UNIT III	**IoT Programming and Integration**	**09 hrs**
IoT Programming and Integration, Interoperability in IoT, Standards and Protocols for Interoperability, Challenges in Interoperability, Introduction to Arduino Programming Basics of Arduino, Programming Environment, Integration of Sensors and Actuators with Arduino Interfacing Techniques, Practical Applications, Introduction to Python Programming, Python Basics, Libraries for IoT, Introduction to Raspberry Pi, Raspberry Pi Hardware, Setting Up Raspberry Pi, Implementation of IoT with Raspberry Pi, IoT Projects with Raspberry Pi, Data Collection and Processing		

Course Contents

Course Contents		
UNIT IV	**Advanced IoT Concepts**	**09 hrs**
Introduction to SDN, SDN Architecture, Benefits of SDN, SDN for IoT, Integration of SDN and IoT, Use Cases, Data Handling and Analytics, Data Storage Solutions, Data Processing Techniques, Cloud Computing, Cloud Services for IoT, Cloud Platforms (AWS, Azure, Google Cloud), Sensor-Cloud, Concept of Sensor-Cloud, Applications and Benefits, Fog Computing, Fog vs. Cloud Computing, Edge Computing, Introduction to IoT Security and Privacy, Importance of security and privacy in IoT systems, Key challenges due to the distributed nature of IoT, Security vs. privacy in IoT: Definitions and differences, Security Solutions for IoT		
UNIT V	**IoT Applcations**	9 hrs.
IoT Applications and Case Studies, Smart Cities, IoT in Urban Planning, Smart Infrastructure Smart Homes, Home Automation Systems, Security Solutions, Connected Vehicles, Vehicle-to-Everything (V2X) Communication, Autonomous Vehicles, Smart Grid, IoT in Energy Management, Smart Metering, Industrial IoT, IoT in Manufacturing, Predictive Maintenance Case Study: Agriculture, Precision Farming, IoT in Crop Monitoring, Case Study: Healthcare, Remote Patient Monitoring, IoT in Medical Devices Case Study: Activity Monitoring, Wearable Devices, Health and Fitness Tracking		

Reference Books

Text Book:

1. Bahga, A., & Madisetti, V. (2014). Internet of things: A hands-on approach. VPT. ISBN: 978-0996025515
2. Buyya, R., & Dastjerdi, A. V. (2016). Internet of things: Principles and paradigms. Morgan Kaufmann. ISBN: 978-0128053959
3. Greengard, S. (2015). The internet of things. MIT Press. ISBN: 978-0262527736
4. Minoli, D. (2013). Building the internet of things with IPv6 and MIPv6: The evolving world of M2M communications. Wiley. ISBN: 978-1118473474
5. Hersent, O., Boswarthick, D., & Elloumi, O. (2012). The internet of things: Key applications and protocols (2nd ed.). Wiley. ISBN: 978-1119994350
6. Pethuru, R., & Anupama, C. R. (2017). The internet of things: Enabling technologies, platforms, and use cases. CRC Press. ISBN: 978-1498761284

Reference Book:

1. Fraden, J. (2010). Handbook of modern sensors: Physics, designs, and applications (4th ed.). Springer. ISBN: 978-1441964656
2. Kurose, J. F., & Ross, K. W. (2012). Computer networking: A top-down approach (6th ed.). Pearson. ISBN: 978-0132856201
3. Bahga, A., & Madisetti, V. (2015). Internet of things: A hands-on approach. Orient Blackswan Private Ltd. ISBN: 978-8173719547
4. Hanes, D., Salgueiro, G., Grossetete, P., Barton, R., & Henry, J. (2017). IoT fundamentals: Networking technologies, protocols, and use cases for the internet of things. Pearson. ISBN: 978-0134307084
5. Perros, H. G. (2021). An introduction to IoT analytics. CRC Press. ISBN: 978-0367686314
6. Relevant articles from journals related to IoT.

E-Content

Web References:

NPTEL

by Prof. Sudip Misra, IIT Kharagpur,

https://archive.nptel.ac.in/courses/106/105/106105166/

Coursera

An Introduction to Programming the Internet of Things (IOT) by University of California, Irvine

https://www.coursera.org/specializations/iot

Introduction to the Internet of Things and Embedded Systems by University of California, Irvine

https://www.coursera.org/learn/iot

MIT OpenCourseWare

The Internet of Things: A Quantum Leap Forward - Special Topics in Supply Chain Management

Industrial Internet of Things: From Theory to Applications - MIT Professional Education

https://professional.mit.edu/course-catalog/industrial-internet-things-theory-applications Stanford

- Stanford School of Engineering

https://online.stanford.edu/courses/xee100-introduction-internet-things

Online Resources:

1. Edge Impulse courseware https://github.com/edgeimpulse/courseware-embedded-machine-learning
2. AWS IoT Core resources https://aws.amazon.com/iot-core/resources/
3. Random nerds Tutorial https://randomnerdtutorials.com/?s=iot
4. Virtual Lab, IIT Kharagpur: http://vlabs.iitkgp.ac.in/rtes/

IoT

- Internet technology connecting devices, machines and tools to the internet by means of wireless technologies.
- Over 9 billion 'Things' connected to the Internet, as of now.
- 'Things' connected to the Internet are projected to cross 20 billion in the near future.
- Unification of technologies such as low-power embedded systems, cloud computing, big-data, machine learning, and networking.

IoT

•**IoT Device Growth:**

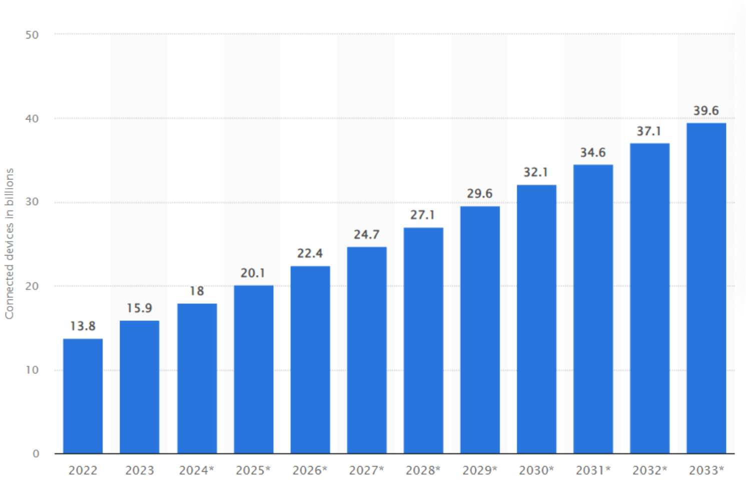
Expected to nearly double from **15.9 billion (2023)** to **32.1 billion (2030)**.

•**Regional Highlight:**

By 2033, **China** will lead with approximately **8 billion consumer devices**.

•**Consumer Market Share:**

Accounts for **60% of all IoT devices** in 2023, projected to remain steady for the next decade.



Major Industry Verticals with 100M+ Devices:

•Manufacturing

•Electricity, gas, steam & A/C

•Water supply & waste management

•Retail & wholesale

•Transportation & storage

•Government

Total Industry Growth:

IoT devices across all verticals forecasted to surpass **8 billion by 2033**.

In the 2000s, we are heading into a new era of ubiquity, where the “users” of the Internet will be counted in billions and where humans may become the minority as generators and receivers of traffic. Instead, most of the traffic will flow between devices and all kinds of “things”, thereby creating a much wider and more complex Internet of Things.

Origin of Terminology

(“The Internet of Things”, ITU Internet Report 2005)

IoT

- ✓ The title of the report was “Internet of Things”
- ✓ Discussed the possibility of internet connected M2M connectivity networks, extending to common household devices.
- ✓ Some areas identified as IoT enablers:
  - RFID,
  - Nanotechnology,
  - Sensors,
  - Smart Networks.

Reference: International Telecommunications Union (ITU). (2005). *The Internet of Things. Executive Summary [Online]*

Definition

Alternate Definition

The Internet of Things (IoT) is the network of physical objects that contain embedded technology to communicate and sense or interact with their internal states or the external environment.

Gartner Research

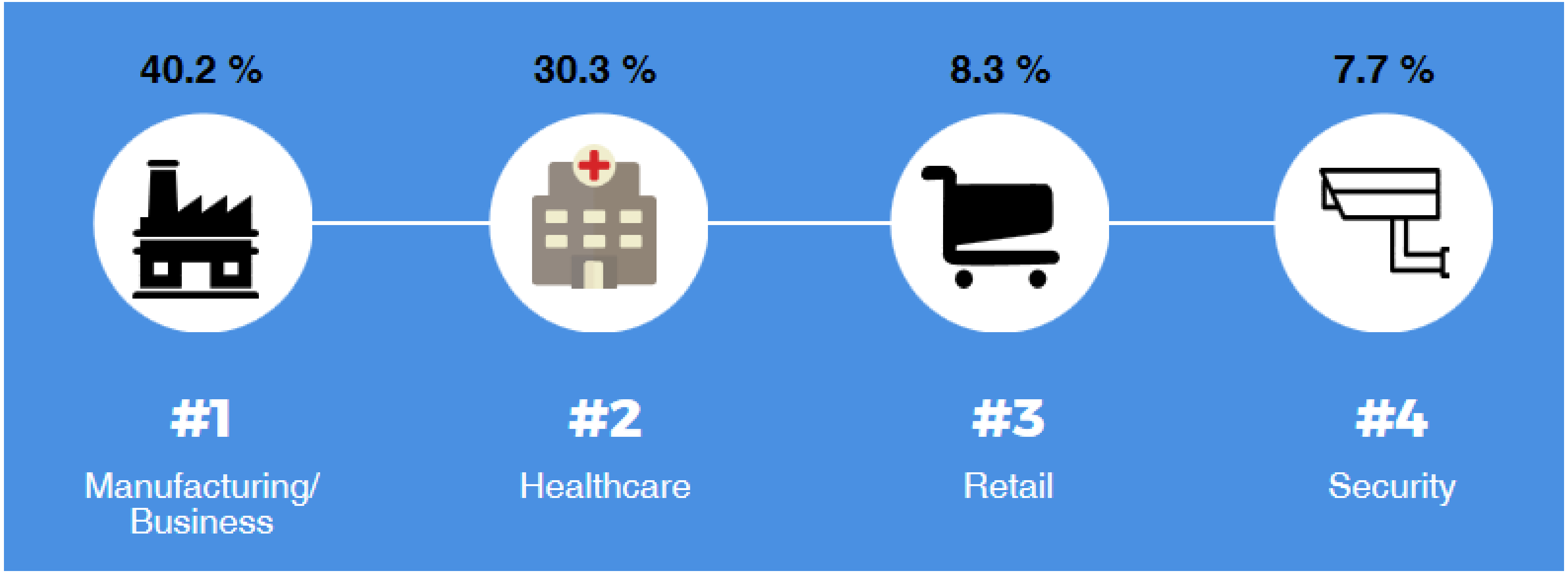
Reference: http://www.gartner.com/it-glossary/internet-of-things/

Characteristics

- ✓ Efficient, scalable and associated architecture
- ✓ Unambiguous naming and addressing
- ✓ Abundance of sleeping nodes, mobile and non-IP devices
- ✓ Intermittent connectivity

Reference: Teemu Savolainen, Jonne Soininen, and Bilhanan Silverajan,*"IPv6 Addressing Strategies for IoT",* IEEE SENSORS JOURNAL, VOL. 13, NO. 10, OCTOBER 2013

IoT Market Share



Source: Intel

- ✓ Business/Manufacturing
  - Real-time analytics of supply chains and equipment, robotic machinery.
- ✓ Healthcare
  - Portable health monitoring, electronic recordkeeping, pharmaceutical safeguards.
- ✓ Retail
  - Inventory tracking, smartphone purchasing, anonymous analytics of consumer choices.
- ✓ Security
  - Biometric and facial recognition locks, remote sensors.

Evolution of Connected Devices

- ✓ ATM
  - These ubiquitous money dispensers went online for the first time way back in 1974.
- ✓ WEB
  - World Wide Web made its debut in 1991 to revolutionize computing and communications.
- ✓ SMART METERS
  - The first power meters to communicate remotely with the grid were installed in the early 2000s.
- ✓ DIGITAL LOCKS
  - Smartphones can be used to lock and unlock doors remotely, and business owners can change key codes rapidly to grant or restrict access to employees and guests.

- ✓ SMART HEALTHCARE
  - Devices connect to hospitals, doctors and relatives to alert them of medical emergencies and take preventive measures.
- ✓ SMART VEHICLES
  - Vehicles self-diagnose themselves and alert owners about system failures.
- ✓ SMART CITIES
  - City-wide infrastructure communicating amongst themselves for unified and synchronized operations and information dissemination.
- ✓ SMART DUST
  - Computers smaller than a grain of sand can be sprayed or injected almost anywhere to measure chemicals in the soil or to diagnose problems in the human body.

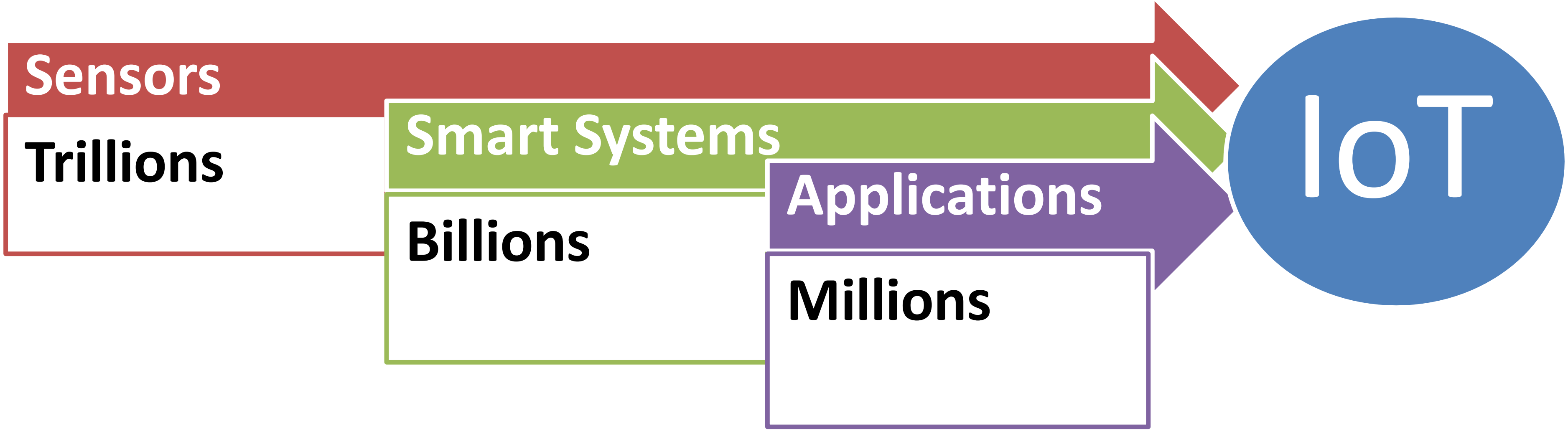
Modern Day IoT Applications

- ✓ Smart Parking
- ✓ Structural health
- ✓ Noise Urban Maps
- ✓ Smartphone Detection
- ✓ Traffic Congestion
- ✓ Smart Lighting
- ✓ Waste Management
- ✓ Smart Roads
- ✓ River Floods
- ✓ Smart Grid
- ✓ Tank level
- ✓ Photovoltaic Installations
- ✓ Water Flow
- ✓ Silos Stock Calculation
- ✓ Perimeter Access Control
- ✓ Liquid Presence

Modern Day IoT Applications

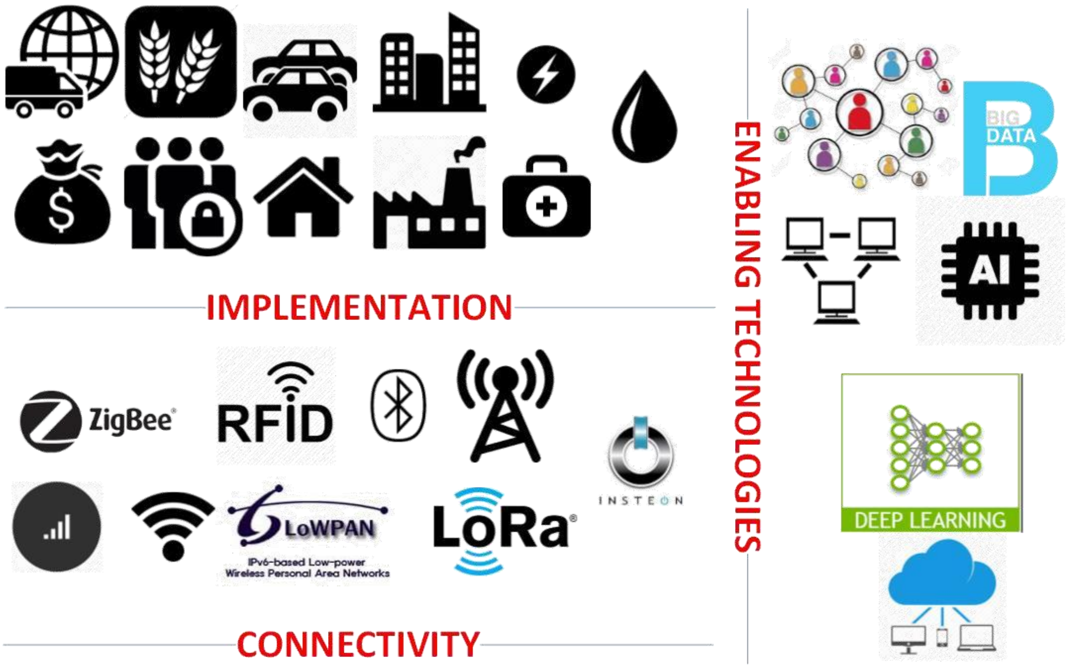
- Forest Fire Detection
- Air Pollution
- Snow Level Monitoring
- Landslide and Avalanche Prevention
- Earthquake Early Detection
- Water Leakages
- Radiation Levels
- Explosive and Hazardous Gases
- Supply Chain Control
- NFC Payment
- Intelligent Shopping Applications
- Smart Product Management

Expected!!

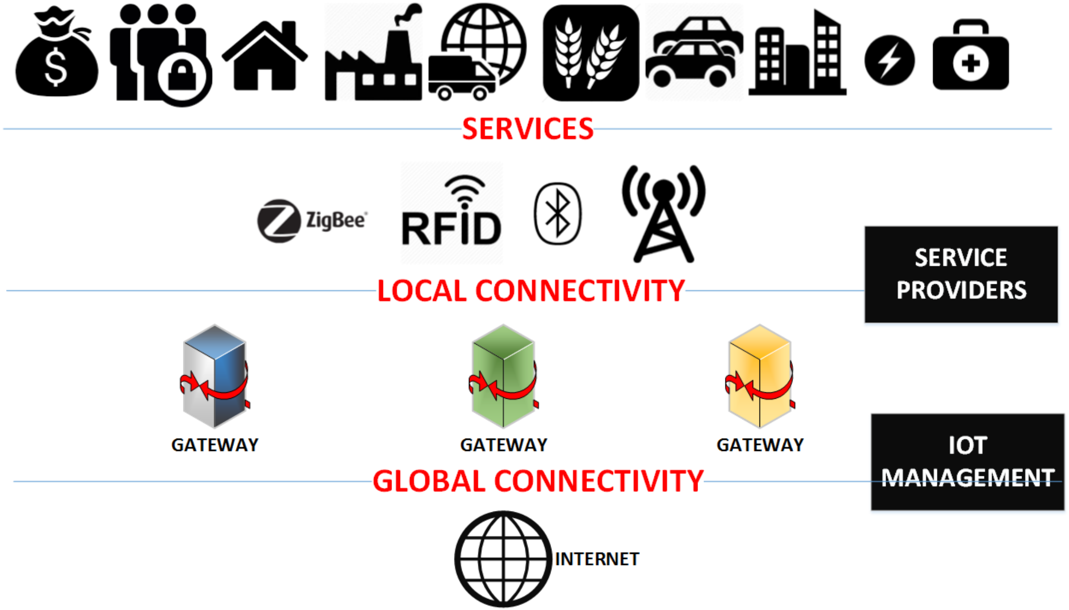

Sensors
Trillions
Smart Systems
Billions
Applications
Millions
IoT

IoT Enablers

IoT Enablers



Connectivity Layers



Baseline Technologies

- ✓ A number of technologies that are very closely related to IoT include
  - Machine-to-Machine (M2M) communications,
  - Cyber-Physical-Systems (CPS)
  - Web-of-Things (WoT).

IoT vs. M2M

- ✓ M2M refers to communications and interactions between machines and devices.
- ✓ Such interactions can occur via a cloud computing infrastructure (e.g., devices exchanging information through a cloud infrastructure).
- ✓ M2M offers the means for managing devices and devices interaction, while also collecting machine and/or sensor data.
- ✓ M2M is a term introduced by telecommunication services providers and, pays emphasis on machines interactions via one or more telcom/communication networks (e.g., 3G, 4G, 5G, satellite, public networks).

IoT vs. M2M

- ✓ **M2M is part of the IoT, while M2M standards have a prominent place in the IoT standards landscape.**
- ✓ However, IoT has a broader scope than M2M, since it comprises a broader range of interactions, including interactions between devices/things, things and people, things with applications and people with applications.
- ✓ It also enables the composition of workflows comprising all of the above interactions.
- ✓ IoT includes the notion of internet connectivity (which is provided in most of the networks outlined above), but is not necessarily focused on the use of telcom networks.

IoT vs. WoT

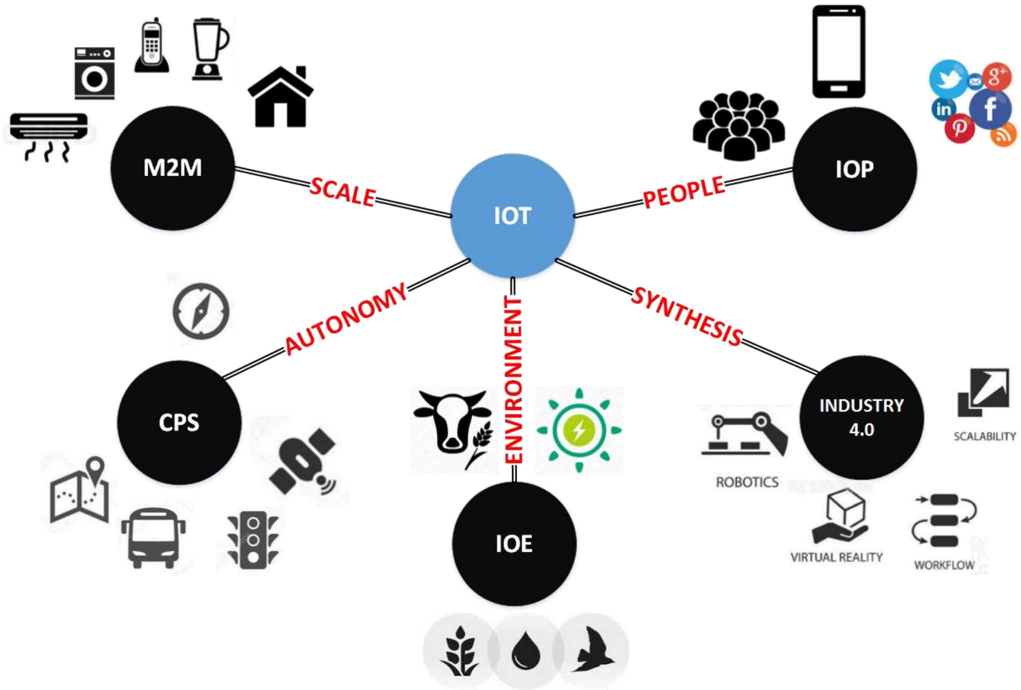
- ✓ From a developer's perspective, the WoT enables access and control over IoT resources and applications using mainstream web technologies (such as HTML 5.0, JavaScript, Ajax, PHP, Ruby n' Rails etc.).
  - The approach to building WoT is therefore based on RESTful principles and REST APIs, which enable both developers and deployers to benefit from the popularity and maturity of web technologies.
  - Still, building the WoT has various scalability, security etc. challenges, especially as part of a roadmap towards a global WoT.

IoT vs. WoT

- ✓ While IoT is about creating a network of objects, things, people,
- systems and applications, WoT tries to integrate them to the Web.

- ✓ Technically speaking, WoT can be thought as a flavour/option of an application layer added over the IoT's network layer. However, the scope of IoT applications is broader and includes systems that are not accessible through the web (e.g., conventional WSN and RFID systems).

Terminological Interdependence



References

https://www.geeksforgeeks.org/basics-computer-networking/

Definition

- ✓ A sensor detects (senses) changes in the ambient conditions or in the state of another device or a system, and forwards or processes this information in a certain manner [1].

"A device which detects or measures a physical property and records, indicates, or otherwise responds to it" [2].

- Oxford Dictionary

References:

1. http://www.businessdictionary.com/definition/sensor.html
2. https://en.oxforddictionaries.com/definition/sensor

Sensors

- ✓ They perform some input functions by sensing or feeling the physical changes in characteristics of a system in response to a stimuli.
- ✓ For example heat is converted to electrical signals in a temperature sensor, or atmospheric pressure is converted to electrical signals in a barometer.

Transducers

- ✓Transducers convert or transduce energy of one kind into another.
- ✓For example, in a sound system, a microphone (input device) converts sound waves into electrical signals for an amplifier to amplify (a process), and a loudspeaker (output device) converts these electrical signals back into sound waves.

Sensor vs. Transducer

✓The word "Transducer" is the collective term used for both **Sensors** which can be used to sense a wide range of different energy forms such as movement, electrical signals, radiant energy, thermal or magnetic energy etc., and **Actuators** which can be used to switch voltages or currents [1].

References:
1. http://www.electronics-tutorials.ws/io/io_1.html

Sensor Features

- ✓It is only sensitive to the measured property (e.g., A temperature sensor senses the ambient temperature of a room.)
- ✓It is insensitive to any other property likely to be encountered in its application (e.g., A temperature sensor does not bother about light or pressure while sensing the temperature.)
- ✓It does not influence the measured property (e.g., measuring the temperature does not reduce or increase the temperature).

Sensors

IoT sensors can be categorized based on their operational principles and data characteristics:

1. **Active vs. Passive Sensors** – Based on energy emission.
2. **Analog vs. Digital Sensors** – Based on signal type.
3. **Scalar vs. Vector Sensors** – Based on measurement dimensions

Active vs. Passive & Analog vs. Digital Sensors

1. **Active vs. Passive Sensors**

- Active Sensors: Emit energy (light, sound, radiation) and measure the response.
  - Examples: LiDAR, Radar, Sonar.
- Passive Sensors: Detect ambient energy without emitting any signals.
  - Examples: Thermometers, microphones, barometers.

2. **Analog vs. Digital Sensors**

- Analog Sensors: Provide continuous signals (voltage, current).
  - Examples: Temperature sensors (thermocouple), light-dependent resistors (LDR).
- Digital Sensors: Provide discrete signals (binary, digital output).
  - Examples: IR sensors, digital accelerometers.
  - Response in binary nature. Design to overcome the disadvantages of analog sensors. Along with the analog sensor, it also comprises extra electronics for bit conversion. Example – Passive infrared (PIR) sensor and digital temperature sensor(DS1620).

Sensors

1. Scalar Sensors

•Measure **single magnitude values** (no direction).

•*Examples*: Temperature, humidity, pressure, light intensity.

2. Vector Sensors

•Measure **both magnitude and direction**.

•*Examples*: Accelerometers (motion direction), magnetometers (earth's magnetic field), gyroscopes (angular velocity).

Sensor Types

1. Environmental Sensors

Sensor Type	Technical Name	Example Models
Temperature Sensor	Thermocouple, RTD (Resistance Temperature Detector), Thermistor	DHT11, LM35, DS18B20
Humidity Sensor	Capacitive Humidity Sensor	DHT22, HIH-4000
Air Quality Sensor	Gas Sensor (MQ series)	MQ-135, CCS811
Light Sensor	Photodiode, LDR (Light Dependent Resistor)	BH1750, TSL2561
Pressure Sensor	Piezoresistive Pressure Sensor	BMP180, MPX5700DP

Sensor Types

2. Motion & Position Sensors

Sensor Type	Technical Name	Example Models
Accelerometer	MEMS Accelerometer	ADXL345, MPU6050
Gyroscope	MEMS Gyroscope	MPU6050, L3G4200D
Magnetometer	Hall Effect Sensor, Fluxgate Magnetometer	HMC5883L, MLX90393
Proximity Sensor	Infrared (IR), Ultrasonic, Capacitive	HC-SR04, GP2Y0A02YK
Vibration Sensor	Piezoelectric Vibration Sensor	SW-420, ADXL335

Sensor Types

3. Optical Sensors

Sensor Type	Technical Name	Example Models
IR Sensor	Infrared Phototransistor	TCRT5000, KY-022
LIDAR Sensor	Time-of-Flight (ToF) Sensor	VL53L0X, LIDAR-Lite v3
Camera Module	CMOS Image Sensor, CCD Sensor	OV7670, Raspberry Pi Camera
Optical Encoder	Rotary Encoder, Incremental Encoder	KY-040, AS5048A

Sensor Types

4. Pressure & Flow Sensors

Sensor Type	Technical Name	Example Models
Barometric Sensor	Barometric Pressure Sensor	BMP180, BME280
Gas Flow Sensor	MEMS Flow Sensor	FS2012, FS300A
Liquid Flow Sensor	Hall Effect Flow Sensor	YF-S201, FS400

5. Acoustic & Sound Sensors

Sensor Type	Technical Name	Example Models
Microphone Sensor	Electret Condenser Microphone	KY-038, MAX9814
Ultrasonic Sensor	Ultrasonic Transducer	HC-SR04, PING)))
Piezoelectric Sensor	Piezoelectric Crystal	PIEZO Vibration Sensor

Sensor Types

6. Bio-Medical Sensors

Sensor Type	Technical Name	Example Models
Heart Rate Sensor	Photoplethysmography (PPG)	MAX30100, Pulse Sensor
ECG Sensor	Electrocardiogram Sensor	AD8232
Blood Pressure Sensor	Oscillometric Pressure Sensor	BMP280 (for pulse waves)
Glucose Sensor	Electrochemical Glucose Sensor	GY-MCU430

7. Smart Agriculture Sensors

Sensor Type	Technical Name	Example Models
Soil Moisture Sensor	Resistive Soil Moisture Sensor	YL-69, FC-28
Rain Sensor	Capacitive Rain Sensor	FC-37
pH Sensor	Electrochemical pH Sensor	SEN0161
CO2 Sensor	NDIR Gas Sensor	MH-Z19

Sensor Types

8. Industrial & Structural Sensors

Sensor Type	Technical Name	Example Models
Vibration Sensor	MEMS Accelerometer	ADXL345
Strain Gauge	Resistive Strain Gauge	BF350
Load Cell	Piezoelectric Load Cell	HX711

9. IoT & Smart Home Sensors

Sensor Type	Technical Name	Example Models
Gas Leakage Sensor	MQ Series Gas Sensor (LPG, CO, CH4)	MQ-2, MQ-7
Motion Detection	PIR (Passive Infrared Sensor)	HC-SR501
Smoke Detector	Photoelectric Smoke Sensor	MQ-135, SHARP GP2Y

Sensor Resolution

- ✓The <u>resolution</u> of a sensor is the smallest change it can detect in the quantity that it is measuring.
- ✓The resolution of a sensor with a digital output is usually the smallest resolution the digital output it is capable of processing.
- ✓The more is the resolution of a sensor, the more accurate is its precision.
- ✓A sensor's accuracy does not depend upon its resolution.

Analog Sensors

- ✓**Analog Sensors** produce a continuous output signal or voltage which is generally proportional to the quantity being measured.
- ✓Physical quantities such as Temperature, Speed, Pressure, Displacement, Strain etc. are all analog quantities as they tend to be continuous in nature.
- ✓For example, the temperature of a liquid can be measured using a thermometer or thermocouple (e.g. in geysers) which continuously responds to temperature changes as the liquid is heated up or cooled down.

Digital Sensors

- ✓**Digital Sensors** produce discrete digital output signals or voltages that are a digital representation of the quantity being measured.
- ✓Digital sensors produce a binary output signal in the form of a logic "1" or a logic "0", ("ON" or "OFF").
- ✓Digital signal only produces discrete (non-continuous) values, which may be output as a single "bit" (serial transmission), or by combining the bits to produce a single "byte" output (parallel transmission).

Scalar Sensors

- ✓**Scalar Sensors** produce output signal or voltage which is generally proportional to the magnitude of the quantity being measured.
- ✓Physical quantities such as temperature, color, pressure, strain, etc. are all scalar quantities as only their magnitude is sufficient to convey an information.
- ✓For example, the temperature of a room can be measured using a thermometer or thermocouple, which responds to temperature changes irrespective of the orientation of the sensor or its direction.

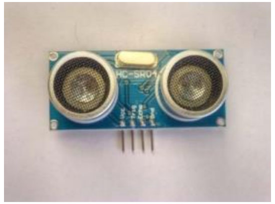
Vector Sensors

- ✓ **Vector Sensors** produce output signal or voltage which is generally proportional to the magnitude, direction, as well as the orientation of the quantity being measured.
- ✓ Physical quantities such as sound, image, velocity, acceleration, orientation, etc. are all vector quantities, as only their magnitude is not sufficient to convey the complete information.
- ✓ For example, the acceleration of a body can be measured using an accelerometer, which gives the components of acceleration of the body with respect to the x,y,z coordinate axes.

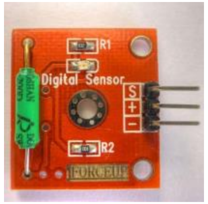
Sensor Types

Light	• Light Dependent resistor • Photo-diode
Temperature	• Thermocouple • Thermistor
Force	• Strain gauge • Pressure switch
Position	• Potentiometer, Encoders • Opto-coupler
Speed	• Reflective/ Opto-coupler • Doppler effect sensor
Sound	• Carbon Microphone • Piezoelectric Crystal
Chemical	• Liquid Chemical sensor • Gaseous chemical sensor

Pressure Sensor
Source: Wikimedia Commons



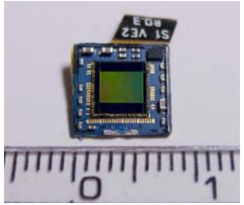
Ultrasonic Distance Sensor
Source: Wikimedia Commons



Tilt Sensor
Source: Wikimedia Commons

Infrared Motion Sensor
Source: Wikimedia Commons

Analog Temperature Sensor
Source: Wikimedia Commons



Camera Sensor
Source: Wikimedia Commons

54

Characteristics of Sensors

Sensors play a crucial role in IoT and automation by converting physical parameters into electrical signals. The key characteristics of sensors are:

1. Accuracy

- **Definition: The closeness of a sensor's measurement to the actual value.**
- **Example: A temperature sensor with ±0.1°C accuracy provides more precise readings than one with ±1°C.**

2. Precision & Repeatability

- **Precision: Ability to provide consistent results over multiple measurements.**
- **Repeatability: Consistency when measuring the same input multiple times under identical conditions.**

- **Example: A weight sensor giving 50.01g, 50.02g, and 50.00g for a 50g object is highly precise.**

Characteristics of Sensors

3. Sensitivity

- **Definition: The ratio of output signal change to input signal change.**
- **Example: A temperature sensor with high sensitivity detects small temperature variations accurately.**

4. Resolution

- **Definition: The smallest change a sensor can detect in the measured quantity.**
- **Example: A digital thermometer with a 0.01°C resolution can detect smaller changes than one with 0.1°C resolution.**

Characteristics of Sensors

3. Sensitivity

- **Definition: The ratio of output signal change to input signal change.**
- **Example: A temperature sensor with high sensitivity detects small temperature variations accurately.**

4. Resolution

- **Definition: The smallest change a sensor can detect in the measured quantity.**
- **Example: A digital thermometer with a 0.01°C resolution can detect smaller changes than one with 0.1°C resolution.**

Characteristics of Sensors

5. Response Time

- **Definition: The time taken by a sensor to reach a stable output after a change in input.**
- **Example: An IR sensor with a fast response time detects motion quickly in security systems.**

6. Linearity

- **Definition: How well the sensor output follows a straight-line relationship with the input.**
- **Example: A pressure sensor showing nonlinear behavior might provide incorrect readings under high pressure.**

7. Range

- **Definition: The minimum and maximum values a sensor can measure.**
- **Example: A humidity sensor with a 0–100% range is more versatile than one with a 20–80% range.**

Characteristics of Sensors

8. Drift

- **Definition: A slow change in sensor readings over time due to aging or environmental factors.**
- **Example: A gas sensor might show variations after prolonged use, requiring recalibration.**

9. Selectivity

- **Definition: The ability to respond to a specific input while ignoring others.**
- **Example: A CO_2 sensor should measure only CO_2 levels, not other gases.**

10. Power Consumption

- **Definition: The amount of power a sensor requires to operate.**
- **Example: IoT sensors with low power consumption are ideal for battery-powered devices.**

Characteristics of Sensors

11. Noise & Interference Resistance

- **Definition: The ability to provide stable readings in noisy environments.**
- **Example: A biomedical ECG sensor should resist external electrical noise for accurate heart rate monitoring.**

12. Cost & Reliability

- **Definition: The trade-off between sensor price and its durability and performance.**
- **Example: Industrial sensors are more expensive but highly durable, while consumer sensors may be cheaper but less reliable.**

Hysteresis Error

- ✓ A hysteresis error causes the sensor output value to vary depending on the sensor’s previous input values.
- ✓ If a sensor's output is different depending on whether a specific input value was reached by increasing or decreasing the input, then the sensor has a hysteresis error.
- ✓ The <u>present reading depends on the past input values</u>.
- ✓ Typically in analog sensors, magnetic sensors, heating of metal strips.

Reference: https://en.wikipedia.org/wiki/Sensor

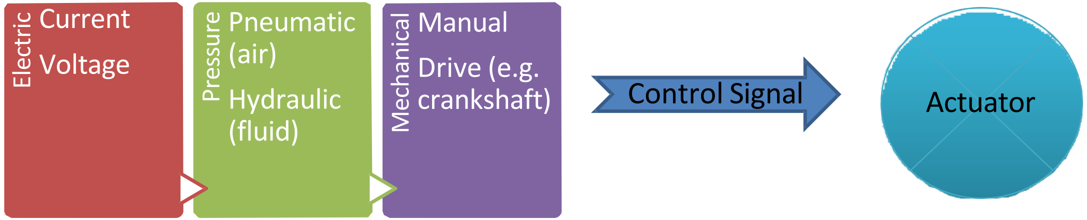
Thank You!!

Actuator

- ✓ An actuator is a component of a machine or system that moves or controls the mechanism or the system.
- ✓ An actuator is the mechanism by which a control system acts upon an environment
- ✓ An actuator requires a control signal and a source of energy.

Actuator

- ✓ Upon receiving a control signal is received, the actuator responds by converting the energy into mechanical motion.
- ✓ The control system can be simple (a fixed mechanical or electronic system), software-based (e.g. a printer driver, robot control system), a human, or any other input.



3

Actuator

Actuator Types

- Hydraulic
- Pneumatic
- Electrical
- Thermal/ Magnetic
- Mechanical

4

Actuator

Hydraulic Actuators

- ✓ A hydraulic actuator consists of a cylinder or fluid motor that uses hydraulic power to facilitate mechanical operation.
- ✓ The mechanical motion is converted to linear, rotary or oscillatory motion.
- ✓ Since liquids are nearly impossible to compress, a hydraulic actuator exerts considerable force.
- ✓ The actuator's limited acceleration restricts its usage.

Reference: https://en.wikipedia.org/wiki/Actuator

5

Actuator

Hydraulic Actuators

5

Actuator

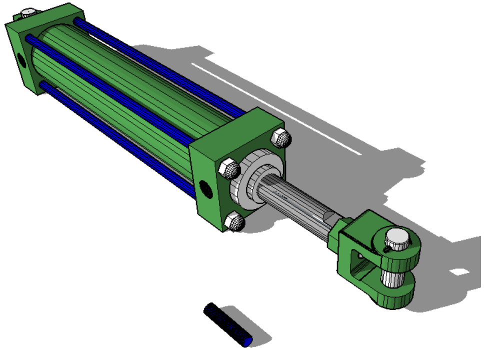


Fig: An oil based hydraulic actuator

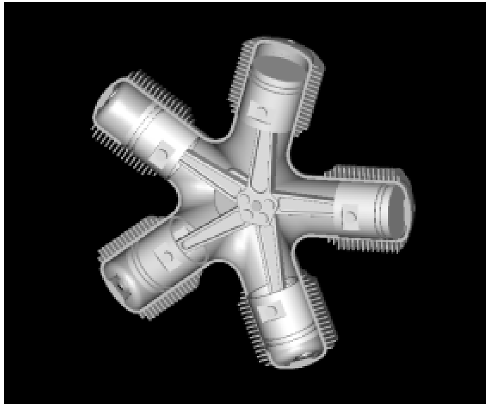


Fig: A radial engine acts as a hydraulic actuator
Source: Wikimedia Commons
File: Radial_engine.gif

6

Pneumatic Actuators

- ✓ A pneumatic actuator converts energy formed by vacuum or compressed air at high pressure into either linear or rotary motion.
- ✓ Pneumatic rack and pinion actuators are used for valve controls of water pipes.
- ✓ Pneumatic energy quickly responds to starting and stopping signals.
- ✓ The power source does not need to be stored in reserve for operation.

- ✓ Pneumatic actuators enable large forces to be produced from relatively small pressure changes (e.g., Pneumatic brakes can are very responsive to small changes in pressure applied by the driver).
- ✓ It is responsible for converting pressure into force.

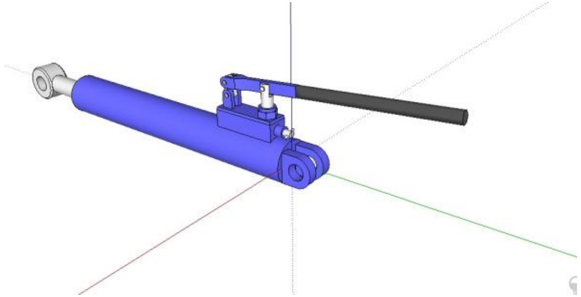


Fig: A manual linear pneumatic actuator

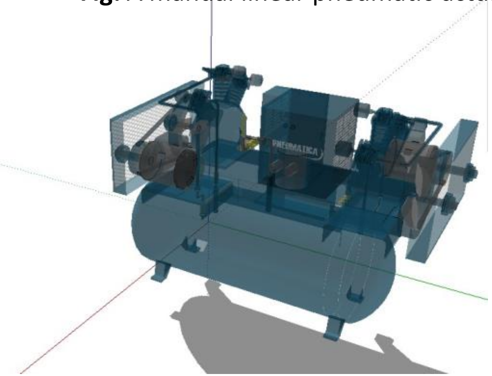


Fig: An air pump acts as a pneumatic actuator

9

Electric Actuators

- ✓An electric actuator is generally powered by a motor that converts electrical energy into mechanical torque.
- ✓The electrical energy is used to actuate equipment such as solenoid valves which control the flow of water in pipes in response to electrical signals.
- ✓Considered as one of the cheapest, cleanest and speedy actuator types available.

Reference: https://en.wikipedia.org/wiki/Actuator

Electric Actuators

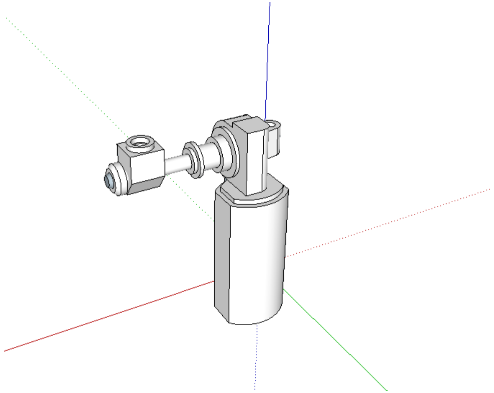


Fig: A motor drive-based rotary actuator

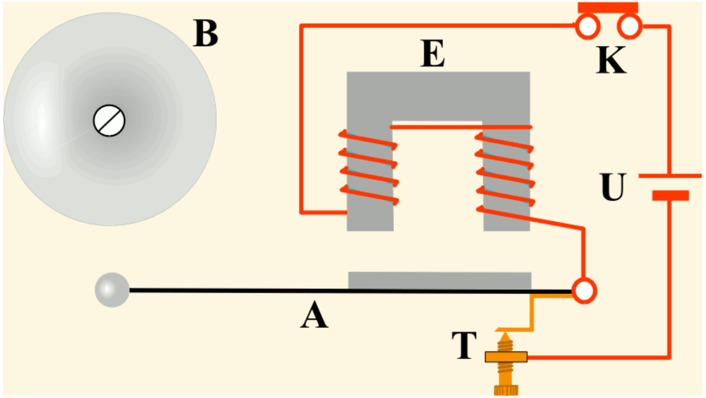


Fig: A solenoid based electric bell ringing mechanism

Source: Wikimedia Commons

File: Electric_Bell_animation.gif

11

Thermal or Magnetic Actuators

- ✓ These can be actuated by applying thermal or magnetic energy.
- ✓ They tend to be compact, lightweight, economical and with high power density.
- ✓ These actuators use shape memory materials (SMMs), such as shape memory alloys (SMAs) or magnetic shape-memory alloys (MSMAs).
- ✓ Some popular manufacturers of these devices are *Finnish Modti Inc.* and *American Dynalloy*.

Reference: https://en.wikipedia.org/wiki/Actuator

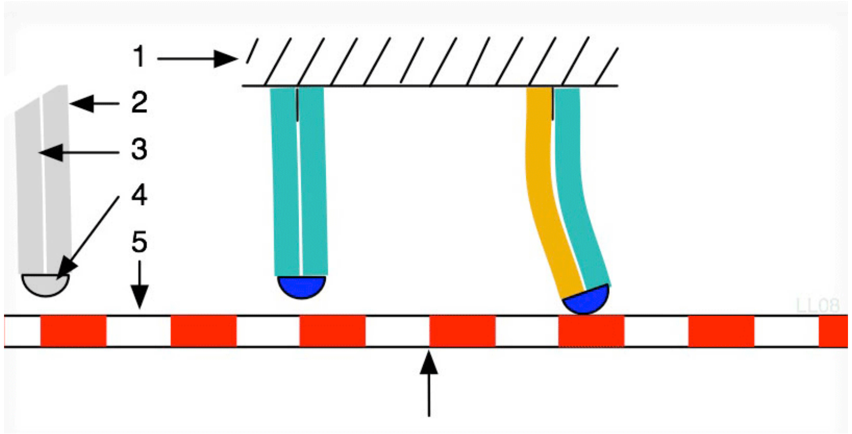


Fig: A piezo motor using SMA

Source: Wikimedia Commons
File: Piezomotor type bimorph.gif

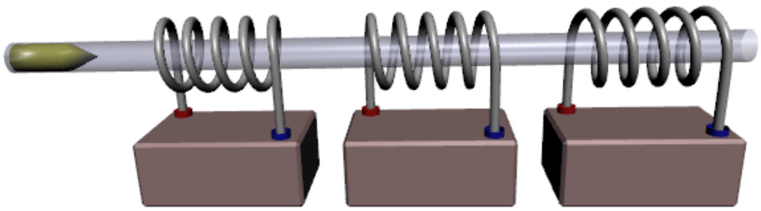


Fig: A coil gun works on the principle of magnetic actuation

Source: Wikimedia Commons
File: Coilgun animation.gif

Mechanical Actuators

- ✓ A mechanical actuator converts rotary motion into linear motion to execute some movement.
- ✓ It involves gears, rails, pulleys, chains and other devices to operate.
- ✓ Example: rack and pinion.

Reference: https://en.wikipedia.org/wiki/Actuator

Fig: A rack and pinion mechanism
Source: Wikimedia Commons[15]
File: Rack and pinion.png

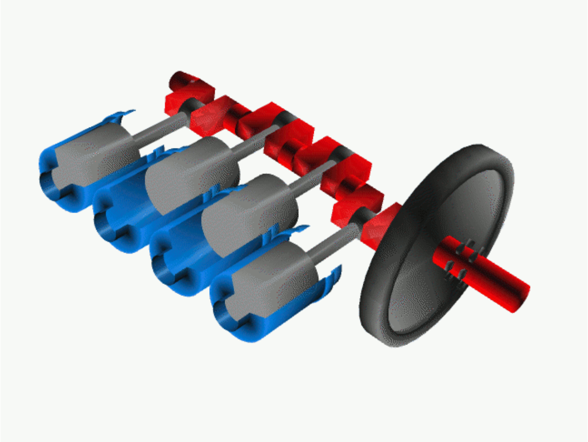


Fig: A crank shaft acting as a mechanical actuator

Source: Wikimedia Commons
File: Cshaft.gif

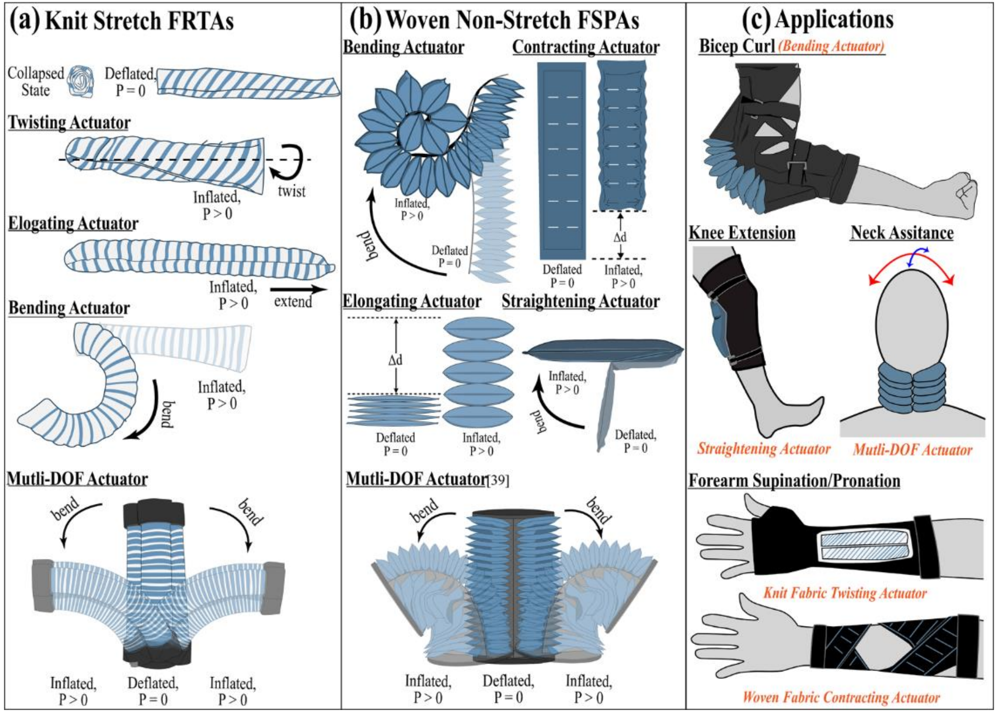
16

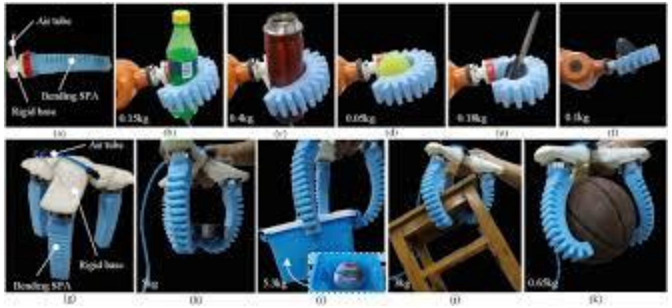
Soft Actuators

- ✓ Soft actuators (e.g. polymer based) are designed to handle fragile objects like fruit harvesting in agriculture or manipulating the internal organs in biomedicine.
- ✓ They typically address challenging tasks in robotics.
- ✓ Soft actuators produce flexible motion due to the integration of microscopic changes at the molecular level into a macroscopic deformation of the actuator materials.

Reference: https://en.wikipedia.org/wiki/Actuator

Soft Actuators

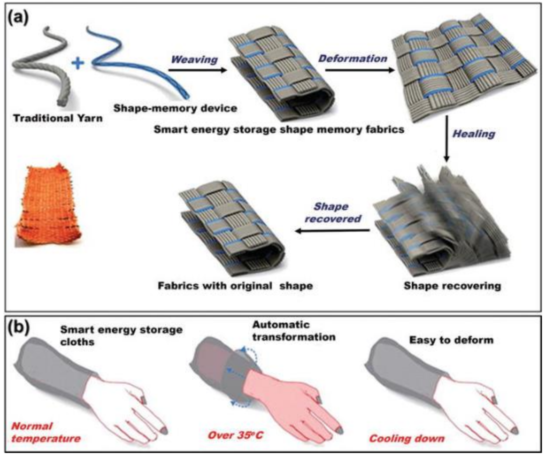




17

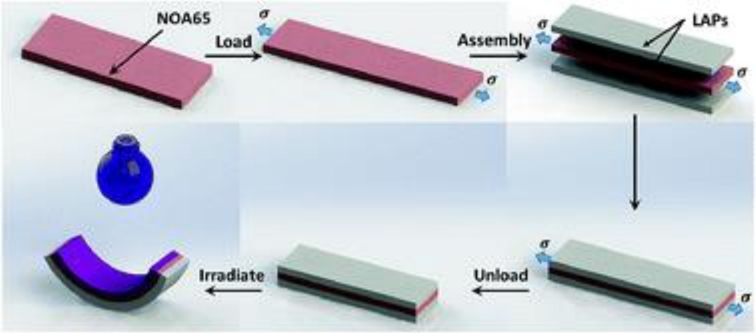
Shape Memory Polymers

- ✓ Shape memory polymer (SMP) actuators function similar to our muscles, even providing a response to a range of stimuli such as light, electrical, magnetic, heat, pH, and moisture changes.
- ✓ SMP exhibits surprising features such a low density, high strain recovery, biocompatibility, and biodegradability.



Light Activated Polymers

- ✓Photopolymer/light activated polymers (LAP) are a special type of SMP that are activated by light stimuli.
- ✓The LAP actuators have instant response.
- ✓They can be controlled remotely without any physical contact, only using the variation of light frequency or intensity.



19

Thank You!!

Internet

Key Elements of the Internet

Hardware Components

1.End Devices (Clients and Servers):

1. Examples: Computers, smartphones, tablets, and IoT devices.
2. Clients request information (e.g., accessing a website), and servers provide the requested data (e.g., hosting the website).

2.Routers: Direct data packets between networks and devices. They ensure data takes the most efficient path to its destination.

3.Switches: Operate within local networks, connecting multiple devices within a LAN (Local Area Network).

4.Modems: Convert digital data from a device into signals that can be transmitted over telephone or cable lines and vice versa.

5.Internet Backbone: Composed of undersea cables, satellites, data centers, and large-scale routers that form the core infrastructure of the internet.

6.Data Centers: Facilities housing servers that store and process vast amounts of information, such as websites, cloud services, and databases.

Internet

Software Components

1.Protocols:

1. Define how data is transmitted and received. Examples include:
    1. **TCP/IP**: Ensures reliable transmission.
    2. **DNS**: Converts domain names into IP addresses.
    3. **HTTP/HTTPS**: Used for accessing web pages.

2.Browsers and Applications:

1. Examples: Google Chrome, Firefox, and mobile apps.
2. Provide user interfaces for accessing internet resources.

3.Operating Systems and Middleware:

1. Manage network connections and facilitate communication between hardware and applications.

Internet

Software Components

1.Protocols:

1. Define how data is transmitted and received. Examples include:
    1. **TCP/IP**: Ensures reliable transmission.
    2. **DNS**: Converts domain names into IP addresses.
    3. **HTTP/HTTPS**: Used for accessing web pages.

2.Browsers and Applications:

1. Examples: Google Chrome, Firefox, and mobile apps.
2. Provide user interfaces for accessing internet resources.

3.Operating Systems and Middleware:

1. Manage network connections and facilitate communication between hardware and applications.

Internet

Steps Involved in Internet Communication

Step 1: User Initiates a Request

- Example: You type a website URL (e.g., www.google.com) in a browser.
- This request is sent using **HTTP/HTTPS** protocols.

Step 2: DNS Lookup

- The **Domain Name System (DNS)** converts the URL (www.google.com) into an IP address (e.g., 142.250.190.78).
- This IP address is necessary for locating the server hosting the website.

Step 3: Data Packets Creation

- Your device divides the request into smaller **data packets**.
- Each packet is assigned a sequence number and includes:
  - Destination IP address (server's address).
  - Source IP address (your device's address).

Step 4: Routing

- Packets are sent to the nearest **router**.
- Routers analyze the destination IP and forward the packets through the most efficient path across networks.

Internet

Step 5: Reaching the Server

•The data packets travel through multiple routers and network nodes until they reach the destination server hosting the requested website.

Step 6: Server Processes the Request

•The server receives the packets, processes the request (e.g., retrieving the webpage), and sends a response back.

Step 7: Returning Data to the User

•The server breaks its response into packets and sends them back through the network.

•These packets follow the reverse route (or a different route if necessary) to your device.

Step 8: Reassembly

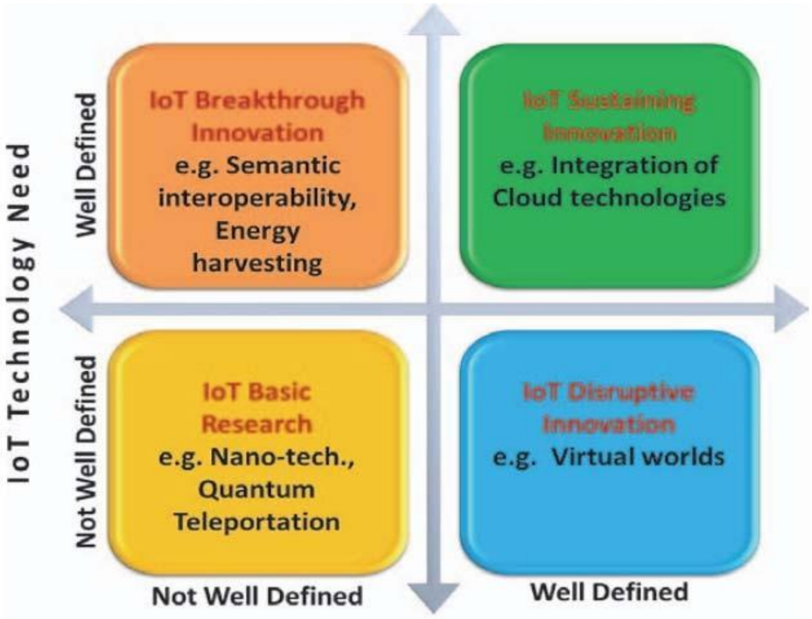
•Your device reassembles the data packets in the correct order using **TCP**.

•If any packets are lost, the server retransmits them.

Step 9: Rendering the Content

•The browser interprets the received data (HTML, CSS, JavaScript) and renders the website for you to view.

Convergence of Domains

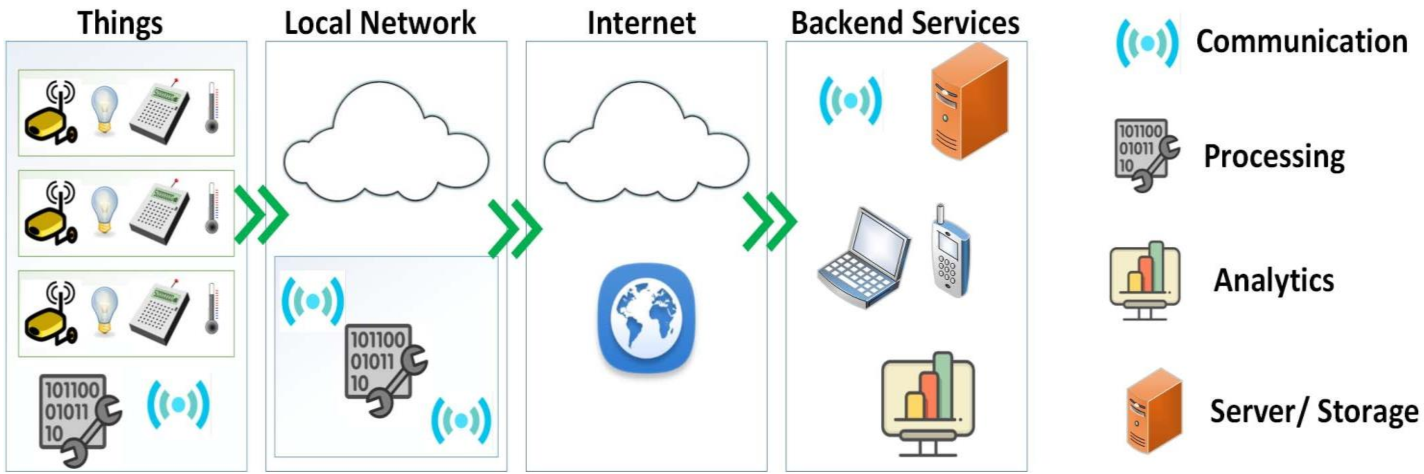


Source: O. Vermesan, P. Friess**,** "Internet of Things – Converging Technologies for Smart Environments and Integrated Ecosystems", River Publishers, Series in Communications, 2013

2

IoT Components

- Device (The Thing)
- Local Network
- Internet
- Backend Services
- Applications


Things
Local Network
Internet
Backend Services
101100
01011
10
Communication
Processing
Analytics
Server/ Storage

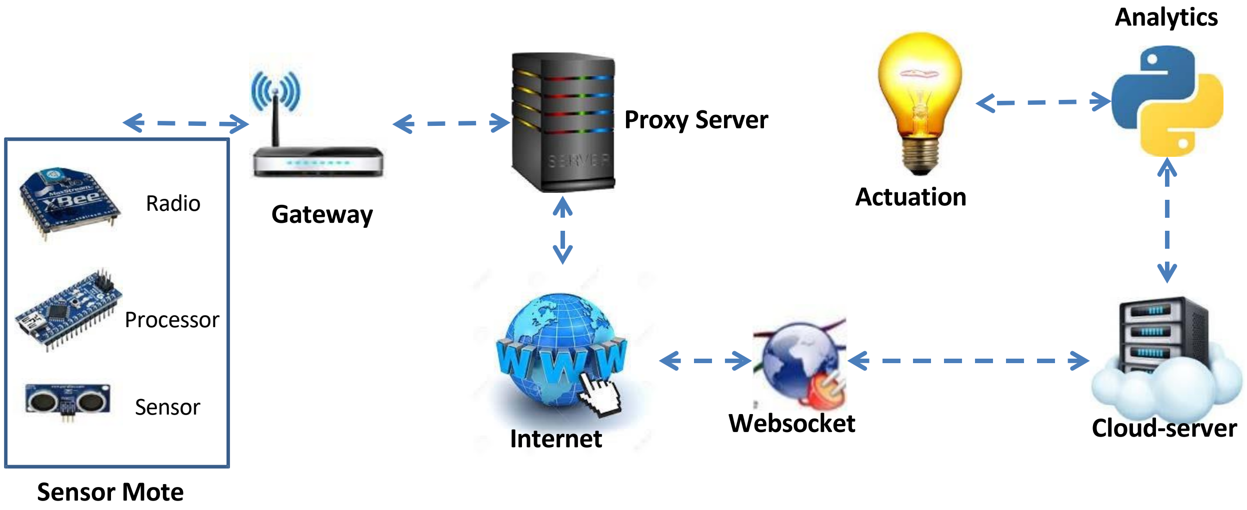
4

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

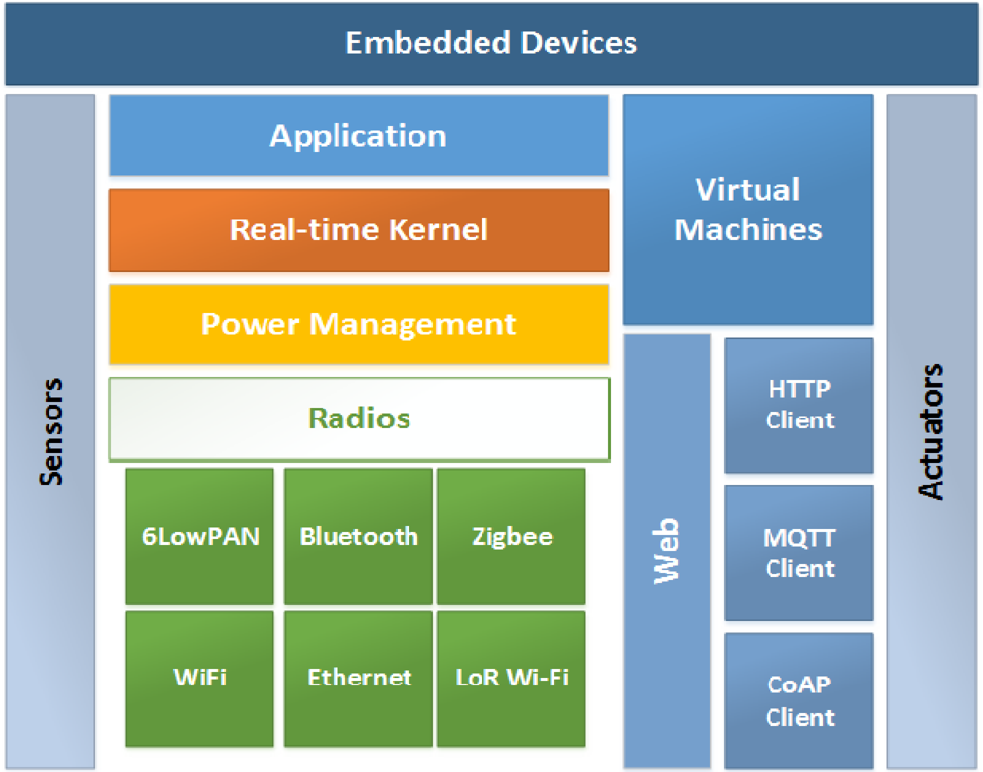
Source: O Vermesan, P. Friess, "Internet of Things – Converging Technologies for Smart Environments and Integrated Ecosystems", River Publishers, Series in Communications, 2013

An Example IoT Implementation



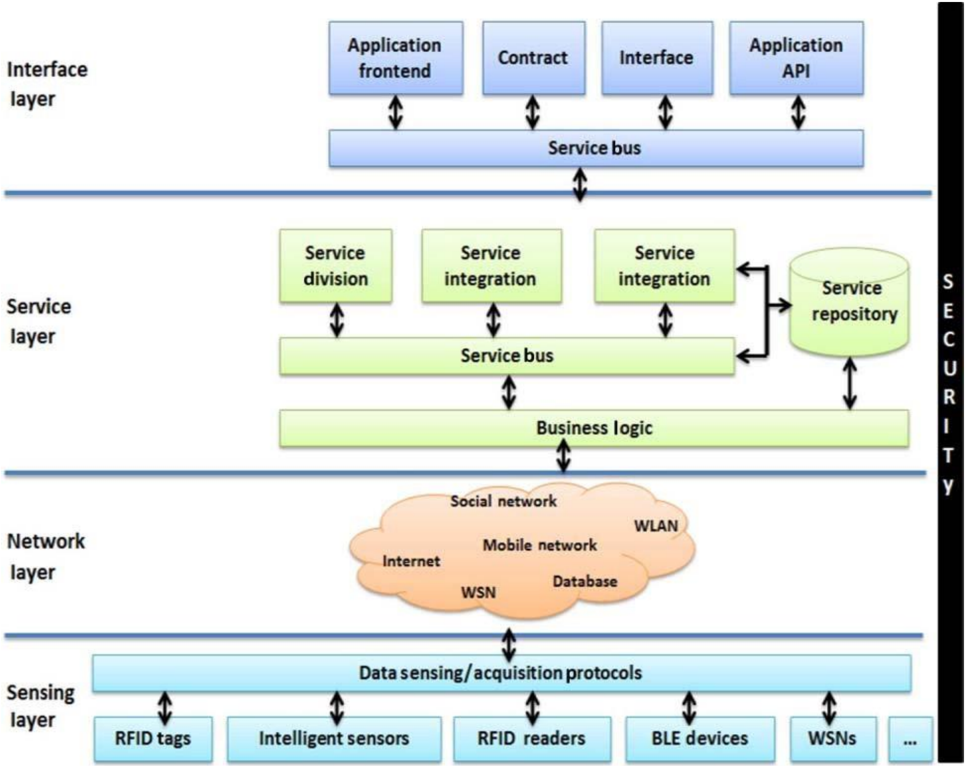
6

IoT Interdependencies



7

IoT Service Oriented Architecture



8

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

Networking

- Networking is the process of connecting multiple devices (computers, servers, etc.) to share resources such as files, internet, and communication services.
- It enables efficient communication and data exchange between devices.

Types of Networks

1. **LAN (Local Area Network)** – Covers a small geographic area, like a home, office, or building.
2. **MAN (Metropolitan Area Network)** – Covers a city or large campus.
3. **WAN (Wide Area Network)** – Covers large distances, like the internet.
4. **PAN (Personal Area Network)** – Used for personal devices like Bluetooth connections.
5. **SAN (Storage Area Network)** – Dedicated network for storage devices.

9

Networking

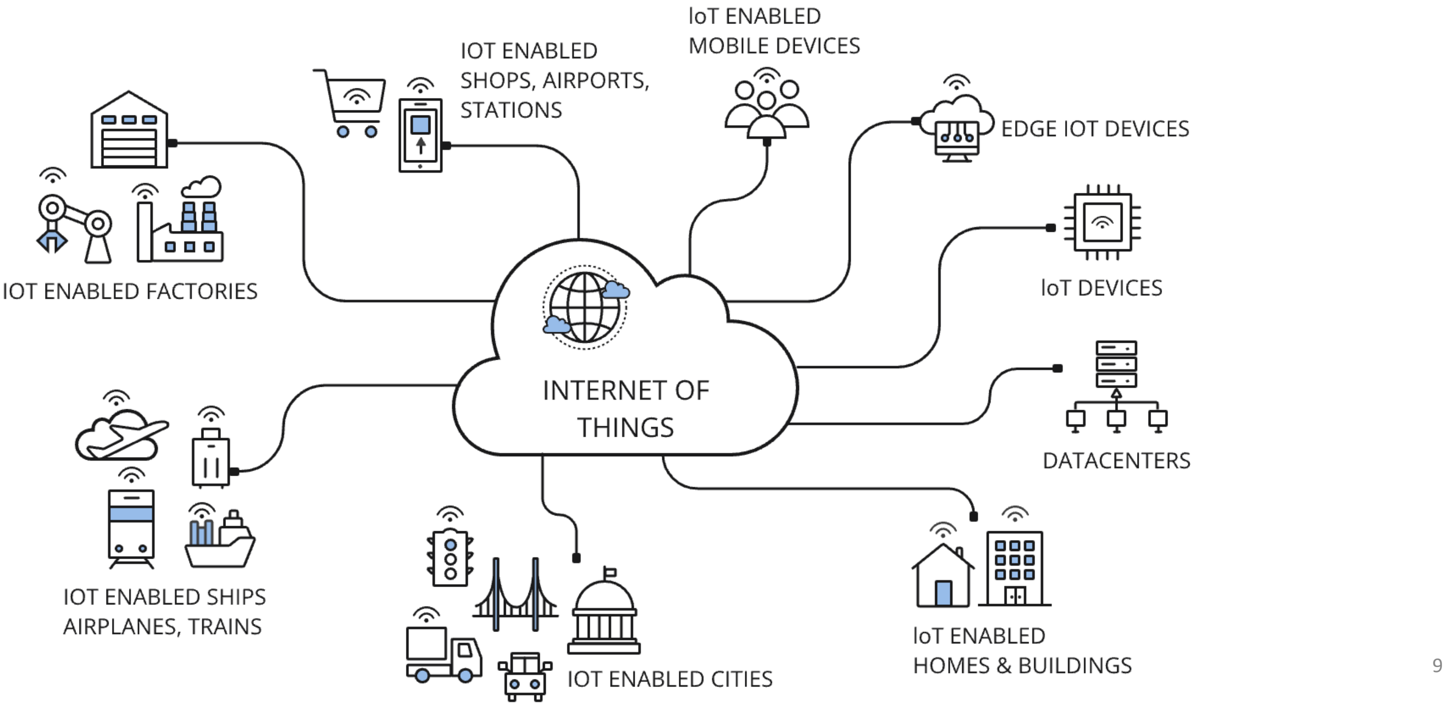
Networking Devices

- **Router** – Directs data packets between different networks.
- **Switch** – Connects devices within a network and efficiently manages traffic.
- **Hub** – Broadcasts data to all devices in a network.
- **Modem** – Converts digital data to analog for transmission over phone lines.
- **Repeater** – Amplifies signals to extend network range.

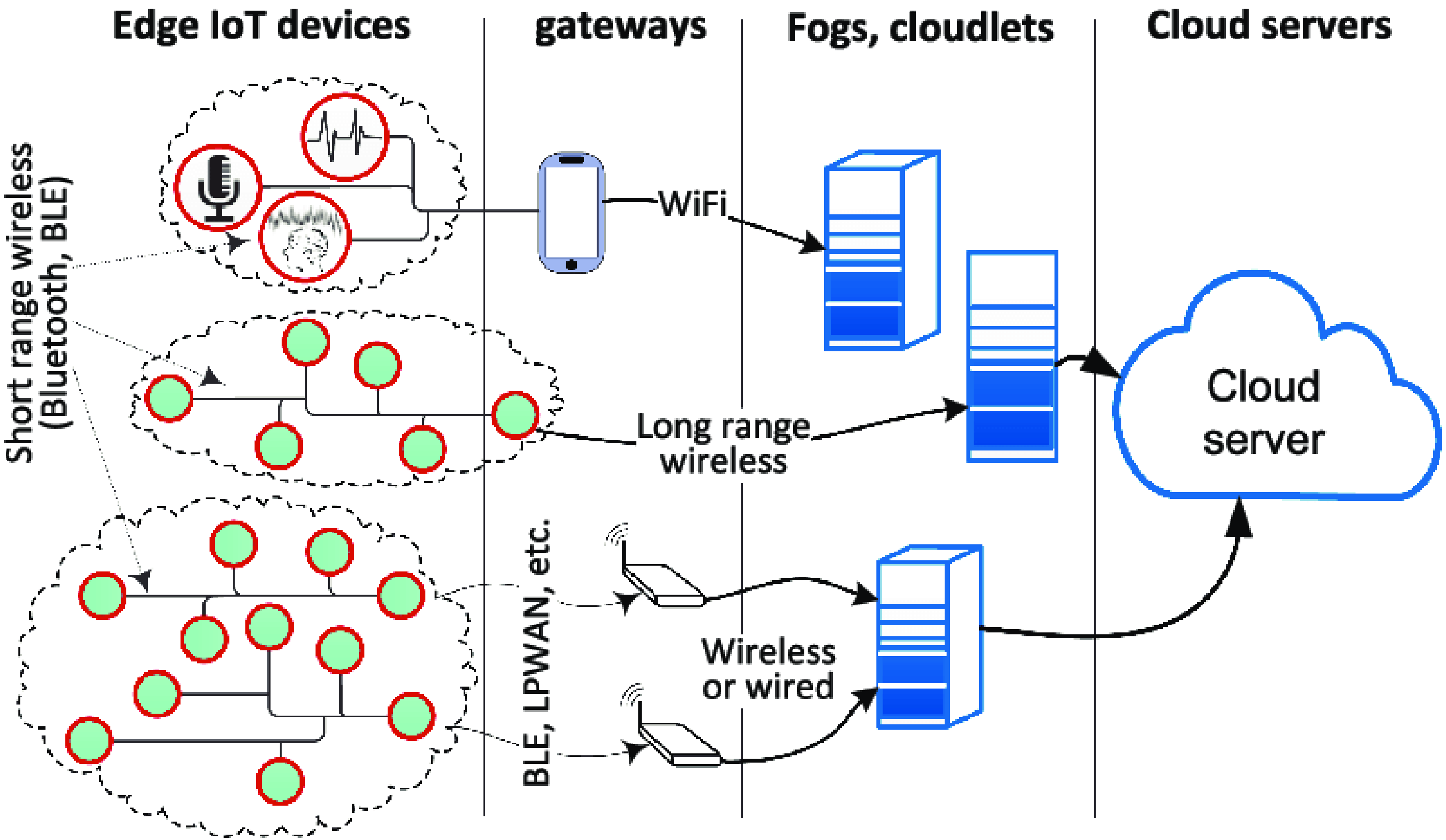
Access Point – Provides wireless connectivity to devices

9

Networking


IOT ENABLED FACTORIES
IOT ENABLED
SHOPS, AIRPORTS,
STATIONS
IoT ENABLED
MOBILE DEVICES
EDGE IOT DEVICES
IoT DEVICES
DATACENTERS
INTERNET OF
THINGS
IOT ENABLED SHIPS
AIRPLANES, TRAINS
IOT ENABLED CITIES
IoT ENABLED
HOMES & BUILDINGS
9

Networking



Edge IoT devices
gateways
Fogs, cloudlets
Cloud servers
Short range wireless (Bluetooth, BLE)
WiFi
Long range wireless
Cloud server
BLE, LPWAN, etc.
Wireless or wired

Network Topology

- **Network topology** defines the arrangement of nodes and connections in a network.

1. Bus Topology

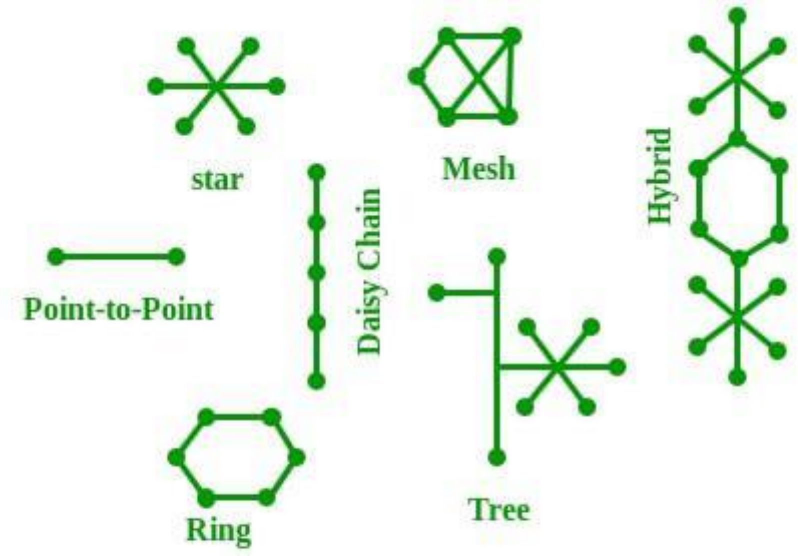
- Single central cable (backbone) connects all devices.
- **Pros**: Cheap, easy to implement.
- **Cons**: If the backbone fails, the entire network goes down.

2. Star Topology

- All devices are connected to a central hub/switch.
- **Pros**: Easy to manage, scalable.
- **Cons**: If the central hub fails, the network stops working.

3. Ring Topology

- Devices are connected in a circular loop.
- **Pros**: Predictable network performance.
- **Cons**: A single failure can disrupt the entire network.



Network Topology

Mesh Topology

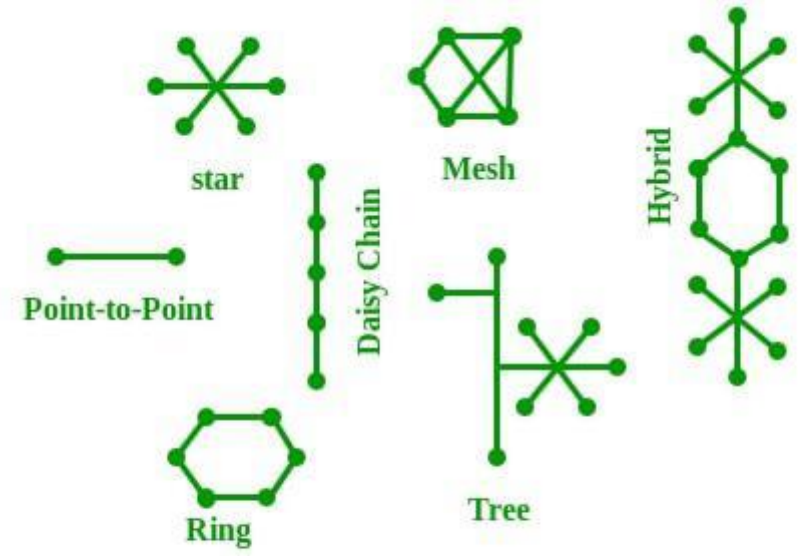
- Every node is connected to every other node.
- **Pros**: Highly reliable, multiple redundant paths.
- **Cons**: Expensive and complex.

5. Hybrid Topology

- Combination of two or more topologies (e.g., Star + Bus).
- **Pros**: Flexible and scalable.
- **Cons**: Complex to implement.

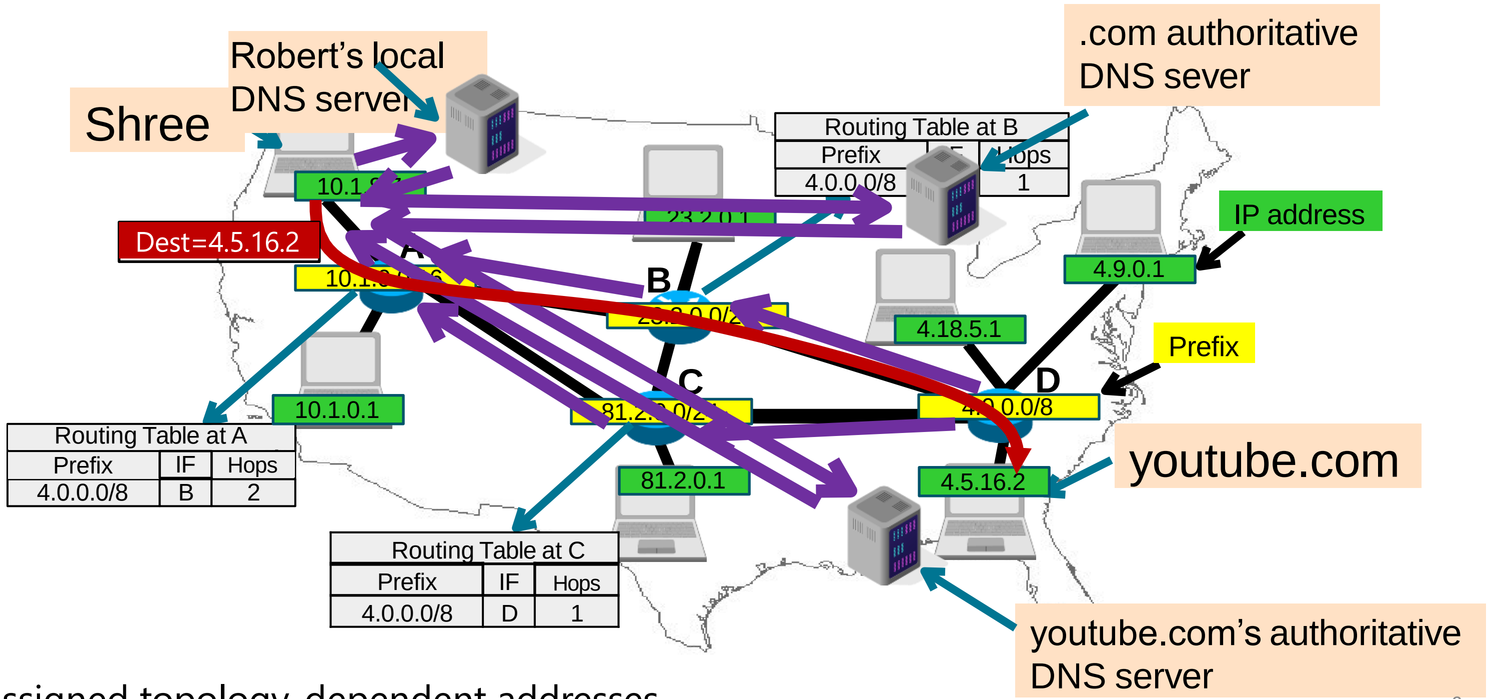
6. Tree Topology

- A hierarchical structure with parent-child nodes.
- **Pros**: Ideal for large networks.
- **Cons**: If a root node fails, sub-networks disconnect.



9

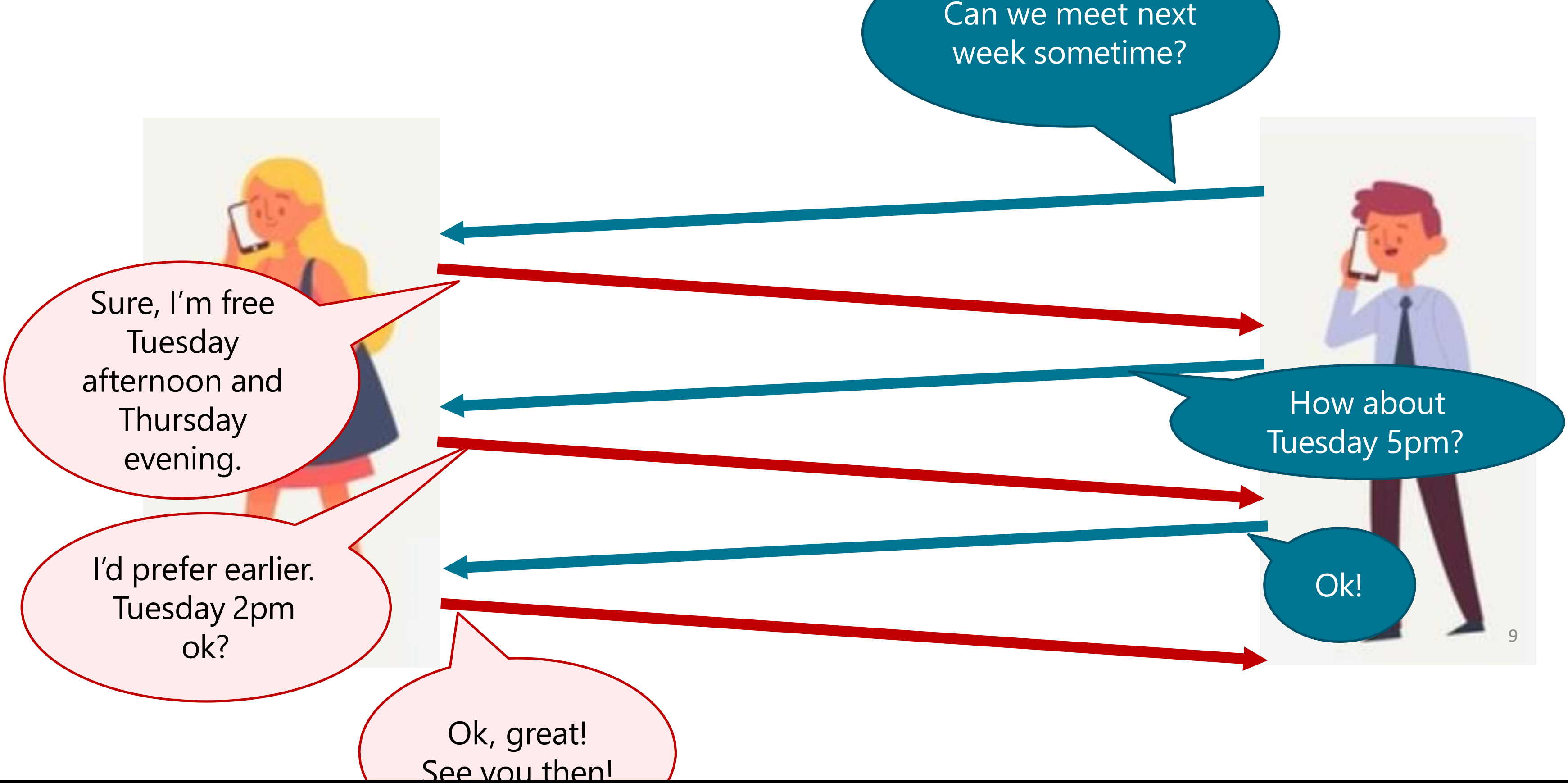
Communication



- Hosts assigned topology-dependent addresses
- Routers advertise address blocks ("prefixes")
- Routers compute "shortest" paths to prefixes
- Map IP addresses to names with DNS

Protocols

What is a Protocol?



Protocol

What is a Protocol?

- Sequence of communications used to conduct some activity in a distributed system
- Protocols are widely used in networks
  - Figure out how fast to send data, discover paths to destinations, replicate data, encode data into transmittable patterns, etc.
- Protocols often organized into “suites” or “stacks”
  - Handle collection of activities associated with particular environment 9
  - Examples: TCP/IP (Internet), Infiniband (Data Center), Bluetooth (IoT)

Networks Have Protocols To….

Compute paths through networks	Routing protocols
Figure out how fast to send data	Transport protocols
Encrypting messages so others can't read them	Encryption protocols
Figure out who has an address	Address resolution protocols
Figure out what kinds of things the network can do	Service discovery protocols [9]

Protocol

The TCP/IP Protocol Stack

	OSI Model (1984, International Standards Organization)	TCP/IP Model (1973-1989, Cerf/Kahn/DARPA)	***Data Protocols***	***Control Protocols***
Layer 7	Application	Application	IMAP, HTTP, H.323, RDP, SSH, BitTorrent, Bitcoin JSON/XML	
Layer 6	Presentation			
Layer 5	Session			
Layer 4	Transport	Transport	TCP, UDP, QUIC, SCTP	
Layer 3	Networking	Networking	IPv4, IPv6	BGP, OSPF, IS-IS BFD, ICMP
Layer 2	Datalink	Datalink & Physical	Ethernet, Wireless Ethernet (WiFi, 802.11)	ARP, DHCP
Layer 1	Physical			

9

OSI Model

- **Stack Structure:** Protocols are often arranged in a layered **stack**.
- In the 1980s, researchers developed **architectural models** to guide network design.
- One key model is the **OSI (Open Systems Interconnection) Model**, which consists of **seven layers**:
  1. **Application Layer** – Implements network applications.
  2. **Presentation Layer** – Handles data representation and encoding.
  3. **Session Layer** – Manages sessions between endpoints.
  4. **Transport Layer** – Ensures reliable data transmission.
  5. **Network Layer** – Handles IP addressing and routing.
  6. **Data Link Layer (MAC Layer)** – Manages local network communication.
  7. **Physical Layer** – Converts data into physical signals.

Data Flow in OSI Model

1. Application Layer (Layer 7)

- **Purpose**: User interaction and protocols (e.g., HTTP, FTP).
- **Example**: Browser sends an HTTP request for www.example.com.

2. Presentation Layer (Layer 6)

- **Purpose**: Data formatting, encryption, decryption.
- **Example**: Encrypts HTTP data using SSL/TLS for secure transmission.

3. Session Layer (Layer 5)

- **Purpose**: Manages sessions between devices.
- **Example**: Establishes and maintains a session with the web server.

4. Transport Layer (Layer 4)

- **Purpose**: Ensures reliable data delivery (segmentation, error checking).
- **Example**: TCP divides data into segments, assigns sequence numbers.

5. Network Layer (Layer 3)

- **Purpose**: Logical addressing and routing.
- **Example**: Adds source and destination IP addresses to form packets.

6. Data Link Layer (Layer 2)

- **Purpose**: Physical addressing and error detection.
- **Example**: Frames are created with MAC addresses of devices.

7. Physical Layer (Layer 1)

- **Purpose**: Transmission of raw data (bits) via physical medium.
- **Example**: Converts frames into electrical, optical, or radio signals.

Example of Data Flow in TCP/IP

When you load a webpage, the following happens:

1. **Application Layer**:
   1. Your browser sends an HTTP request to the server for the webpage.
2. **Transport Layer**:
   1. The data (HTTP request) is segmented into packets.
   2. TCP assigns port numbers (e.g., port 80 for HTTP).
3. **Internet Layer**:
   1. IP assigns source and destination IP addresses.
   2. Routers forward packets to the server.
4. **Network Access Layer**:
   1. Data is converted into electrical or optical signals for transmission over Ethernet or Wi-Fi.

Key Differences: IPv4 vs IPv6

Feature	**IPv4**	**IPv6**
Address Size	32-bit (4 octets)	128-bit (16 octets)
Address Format	Decimal (e.g., 192.168.1.1)	Hexadecimal (e.g., 2001:db8::1)
Address Space	~4.3 billion addresses	Virtually unlimited
Security	Limited, optional IPSec	Built-in IPSec
Header Size	20 bytes	40 bytes
Routing	Complex	Simplified, more efficient
Compatibility	Widely used	Gradually replacing IPv4
Configuration	Requires manual or DHCP setup	Supports auto-configuration

Key Differences: IPv4 vs IPv6

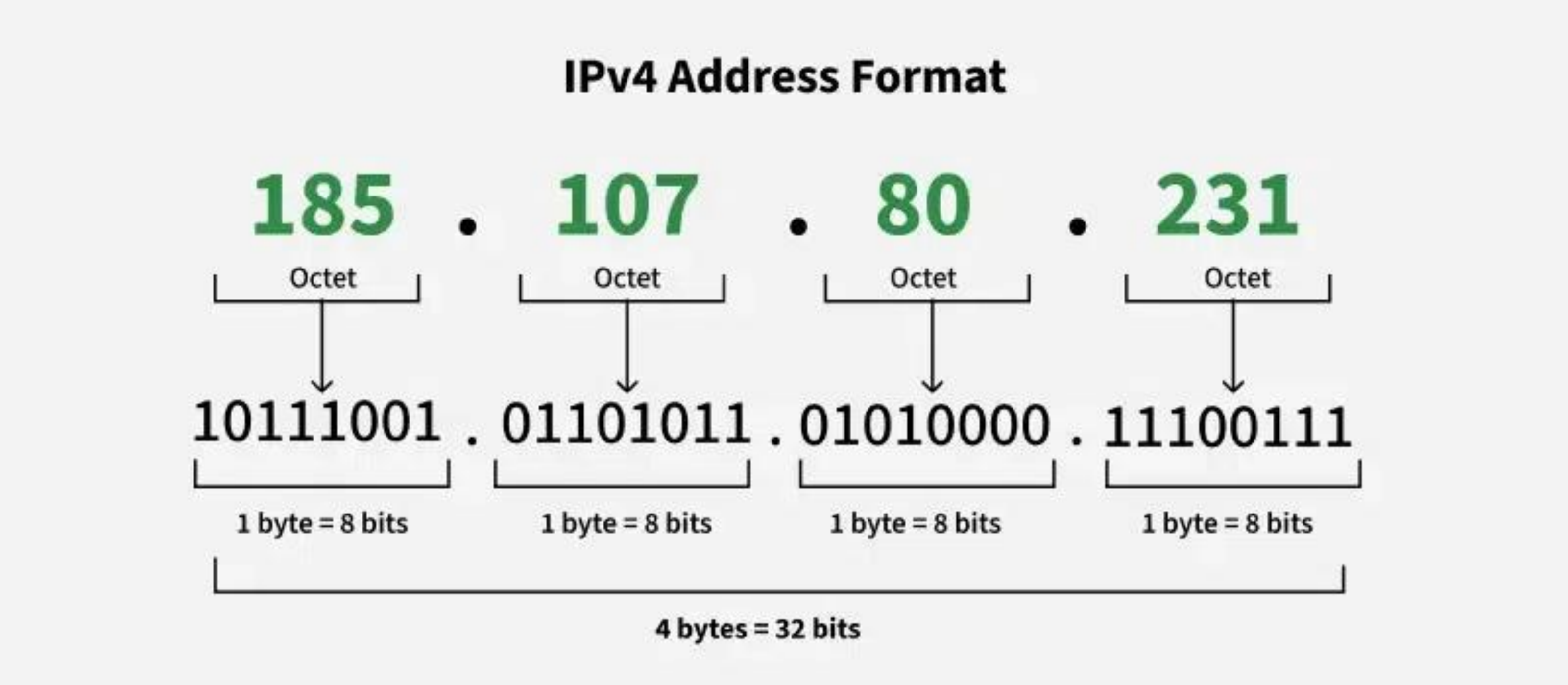
Why IPv6 is Important

1. **Device Explosion**: With IoT and increasing devices, IPv4 cannot meet the demand.
2. **Future-Ready**: IPv6 is designed to handle future internet expansion.
3. **Better Performance**: IPv6 reduces latency with simplified routing and larger data packets.

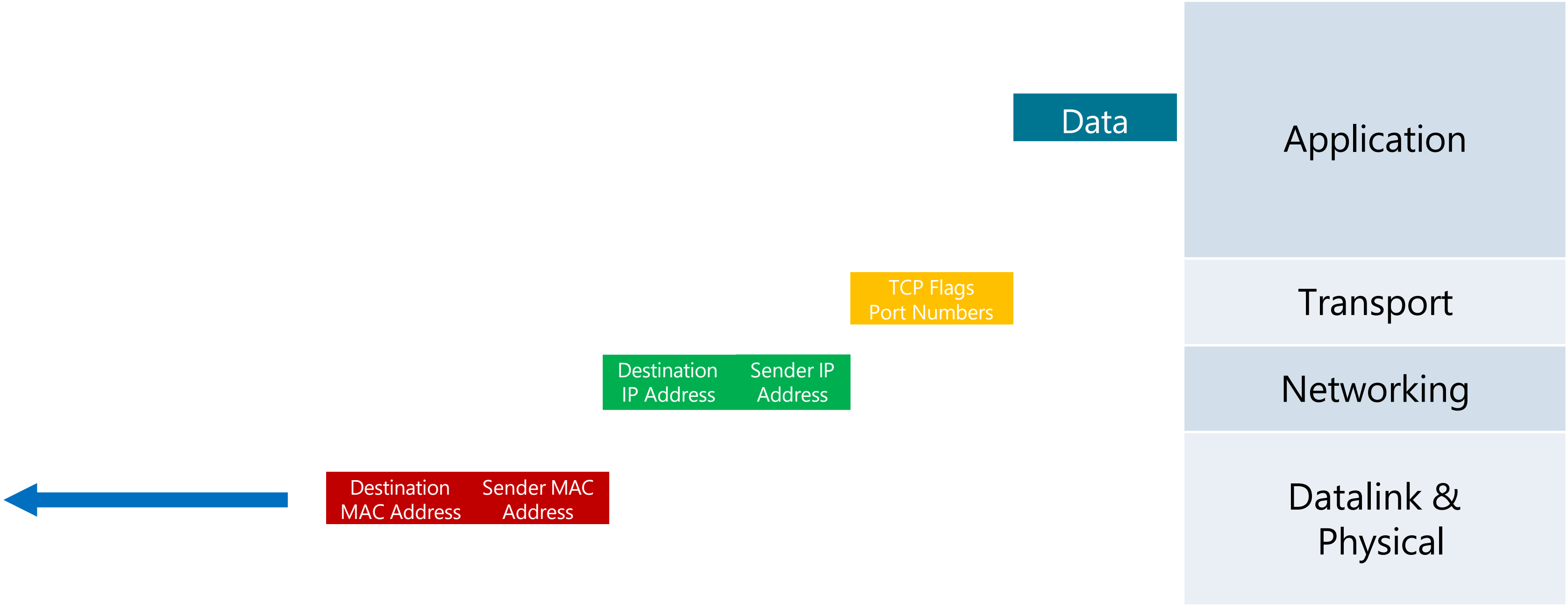
Example

- **IPv4 Address**: 192.168.1.1
- **IPv6 Address**: 2001:0db8:85a3:0000:0000:8a2e:0370:7334 (or compressed: 2001:db8::8a2e:370:7334)

IPv4


IPv4 Address Format
185 . 107 . 80 . 231
Octet
Octet
Octet
Octet
10111001 . 01101011 . 01010000 . 11100111
1 byte = 8 bits
1 byte = 8 bits
1 byte = 8 bits
1 byte = 8 bits
4 bytes = 32 bits

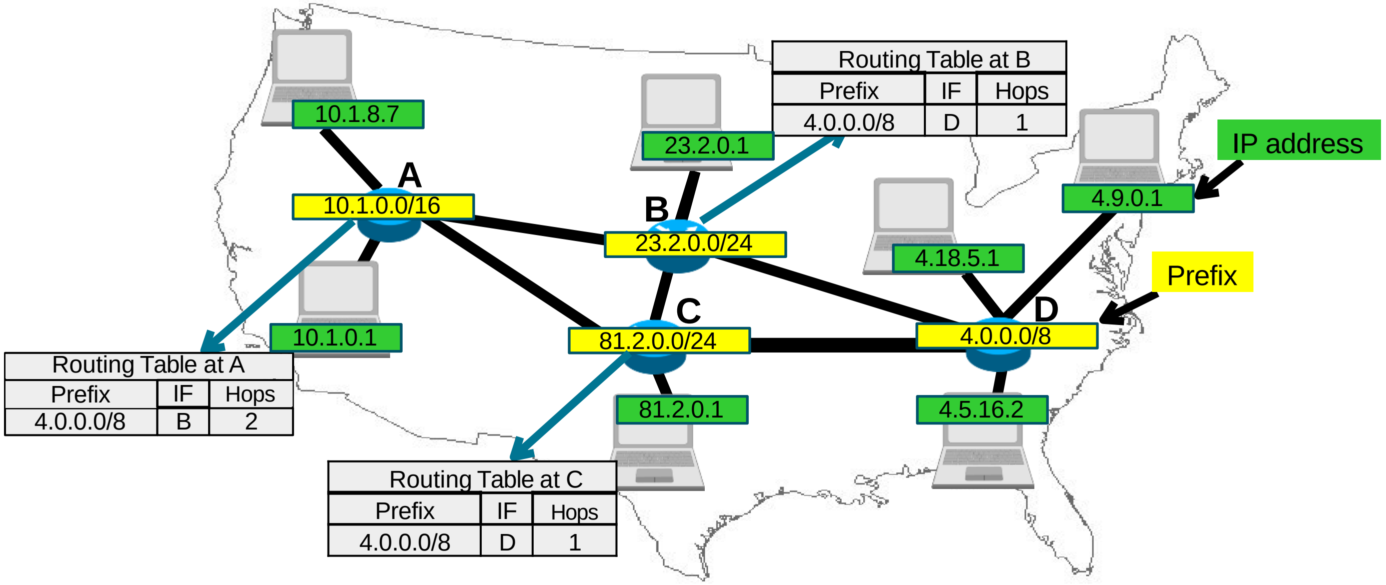
Protocol Encapsulation



9

- Each layer of protocol stack encapsulates data passed to it
- Each forwarding layer inspects data only at that encapsulation layer
  - Switching only looks at Ethernet header, Routing only looks at IP header, etc.
  - Terminology: “Layer-3 switch”, “Layer-4 load balancer”, “Layer-7 load balancer”

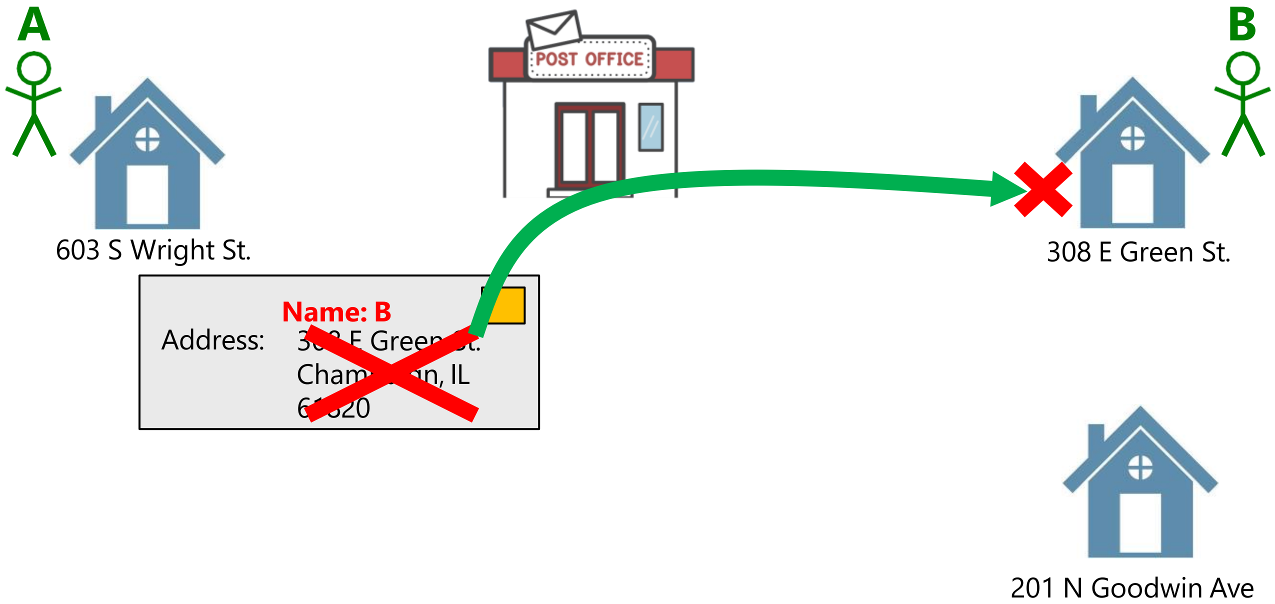
How Can Many Hosts Communicate?



9

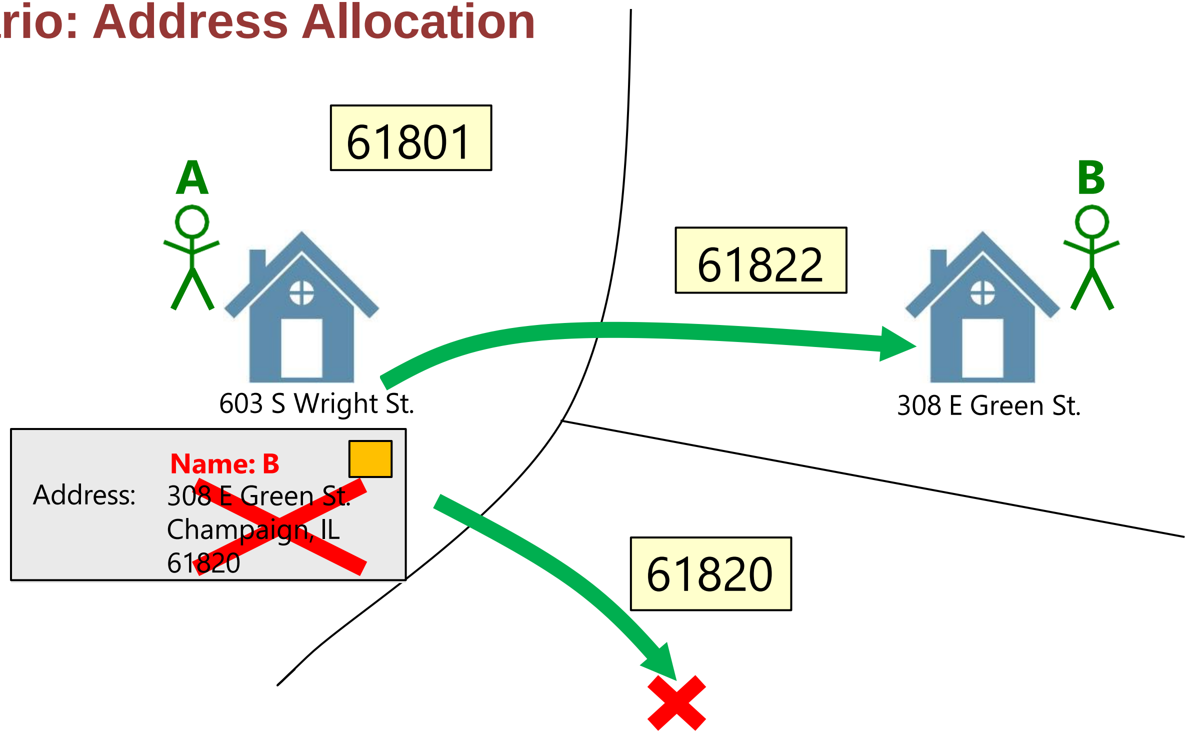
- Hosts assigned topology-dependent addresses
- Routers advertise address blocks ("prefixes")
- Routers compute "shortest" paths to prefixes
- Map IP addresses to names with DNS

Scenario: Sending a Letter

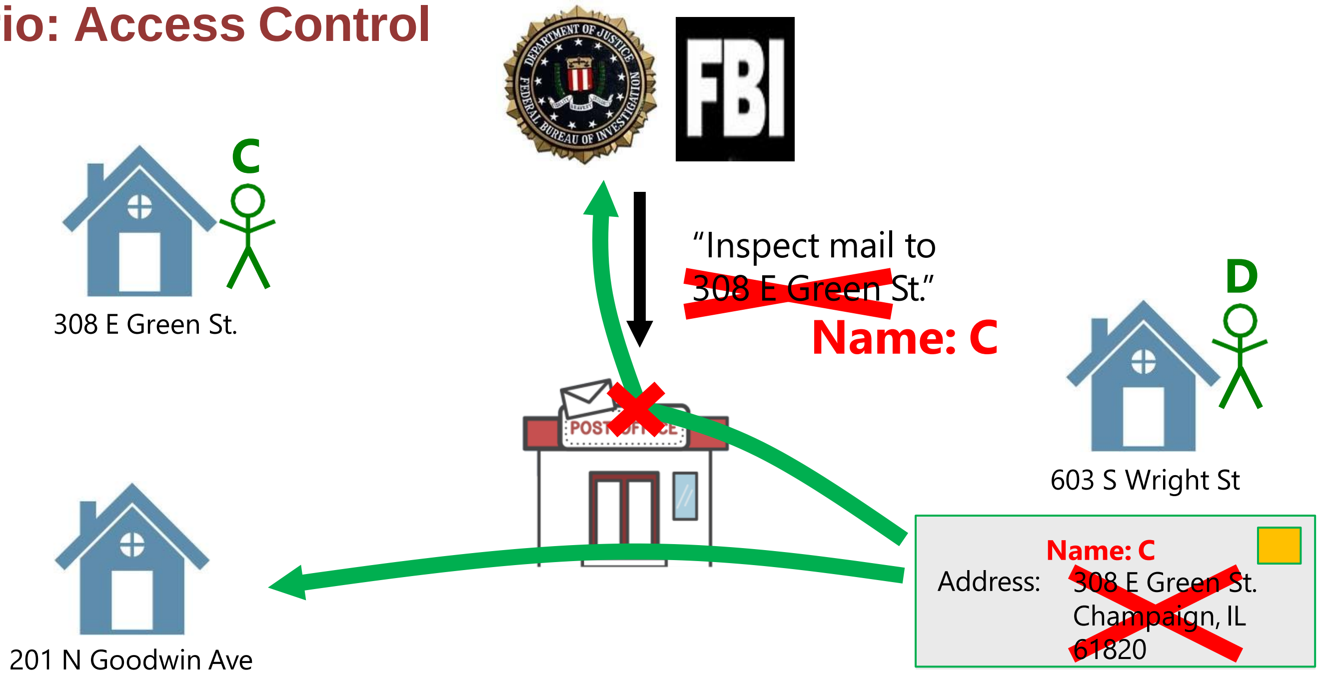


9

Scenario: Address Allocation



Scenario: Access Control



9

Internet Addressing:

Different Layers Use Different Addresses

Application	**URLs** (e.g., http://www.cs.Illinois.edu/index.html) • Identifies "resources" – files, content **Domain names** (e.g., illinois.edu) • Identifies groups of computers under single administrative control
Transport	**Port numbers** (e.g., 143 for IMAP, 22 for SSH) • Identifies application running on a computer
Networking	**IP Addresses** (e.g., 18.220.149.166) • Identifies location where a computer's interface attaches to the Internet
Datalink & Physical	**MAC Addresses** (e.g., 9C-EF-D5-FE-27-72) • Identifies specific computer interface

9

→ *All these addresses are used for end-to-end communication*

Internet Addressing:

MAC Address	IP Address
Datalink layer	Network layer
Flat (location-independent) identifier Like a social security number Usually hard-coded, **requires no configuration**	Hierarchically-assigned, **location-dependent** identifier Like a postal address **Needs to be manually configured**, assigned by DHCP
Portable; can stay the same as the host moves	Not portable; must be changed if host changes networks
Used to get packet to destination on same LAN	Used to get packet to destination IP subnet
Example: 9C-EF-D5-FE-27-72	Example: 18.220.149.166

Thank you