INTRODUCTION TO IoT

1.1 Introduction to IoT

IoT comprises things that have unique identities and are connected to internet. By 2020 there will be a total of 50 billion devices /things connected to internet. IoT is not limited to just connecting things to the internet but also allow things to communicate and exchange data.

Definition-

The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction.

A thing in the internet of things can be a person with a heart monitor implant, a farm animal with a biochip transponder, an automobile that has built-in sensors to alert the driver when tire pressure is low or any other natural or man-made object that can be assigned an Internet Protocol (IP) address and is able to transfer data over a network.

Increasingly, organizations in a variety of industries are using IoT to operate more efficiently, better understand customers to deliver enhanced customer service, improve decision-making and increase the value of the business.

1.2 Characteristics of IoT

1. Intelligence

IoT comes with the combination of algorithms and computation, software & hardware that makes it smart. Ambient intelligence in IoT enhances its capabilities which facilitate the things to respond in an intelligent way to a particular situation and supports them in carrying out specific tasks. In spite of all the popularity of smart technologies, intelligence in IoT is only concerned as means of interaction between devices, while user and device interaction is achieved by standard input methods and graphical user interface.

2. Connectivity

Connectivity empowers Internet of Things by bringing together everyday objects. Connectivity of these objects is pivotal because simple object level interactions contribute towards collective intelligence in IoT network. It enables network accessibility and compatibility in the things. With this connectivity, new market opportunities for Internet of things can be created by the networking of smart things and applications.

3. Dynamic Nature

The primary activity of Internet of Things is to collect data from its environment; this is achieved with the dynamic changes that take place around the devices. The state of these devices change dynamically, example sleeping and waking up, connected and/or disconnected as well as the context of devices including temperature, location and speed. In addition to the state of the device, the number of devices also changes dynamically with a person, place and time.

4. Enormous scale

The number of devices that need to be managed and that communicate with each other will be much larger than the devices connected to the current Internet. The management of data generated from these devices and their interpretation for application purposes becomes more critical. Gartner (2015) confirms the enormous scale of IoT in the estimated report where it stated that 5.5 million new things will get connected every day and 6.4 billion connected things will be in use worldwide in 2016, which is up by 30 percent from 2015. The report also forecasts that the number of connected devices will reach 20.8 billion by 2020.

5. Sensing

IoT wouldn't be possible without sensors which will detect or measure any changes in the environment to generate data that can report on their status or even interact with the environment. Sensing technologies provide the means to create capabilities that reflect a true awareness of the physical world and the people in it. The sensing information is simply the analogue input from the physical world, but it can provide the rich understanding of our complex world.

6. Heterogeneity

Heterogeneity in Internet of Things as one of the key characteristics. Devices in IoT are based on different hardware platforms and networks and can interact with other devices or service platforms through different networks. IoT architecture should support direct network connectivity between heterogeneous networks. The key design requirements for heterogeneous things and their environments in IoT are scalabilities, modularity, extensibility and interoperability.

7. Security

IoT devices are naturally vulnerable to security threats. As we gain efficiencies, novel experiences, and other benefits from the IoT, it would be a mistake to forget about security concerns associated with it. There is a high level of transparency and privacy issues with IoT. It

is important to secure the endpoints, the networks, and the data that is transferred across all of it means creating a security paradigm.

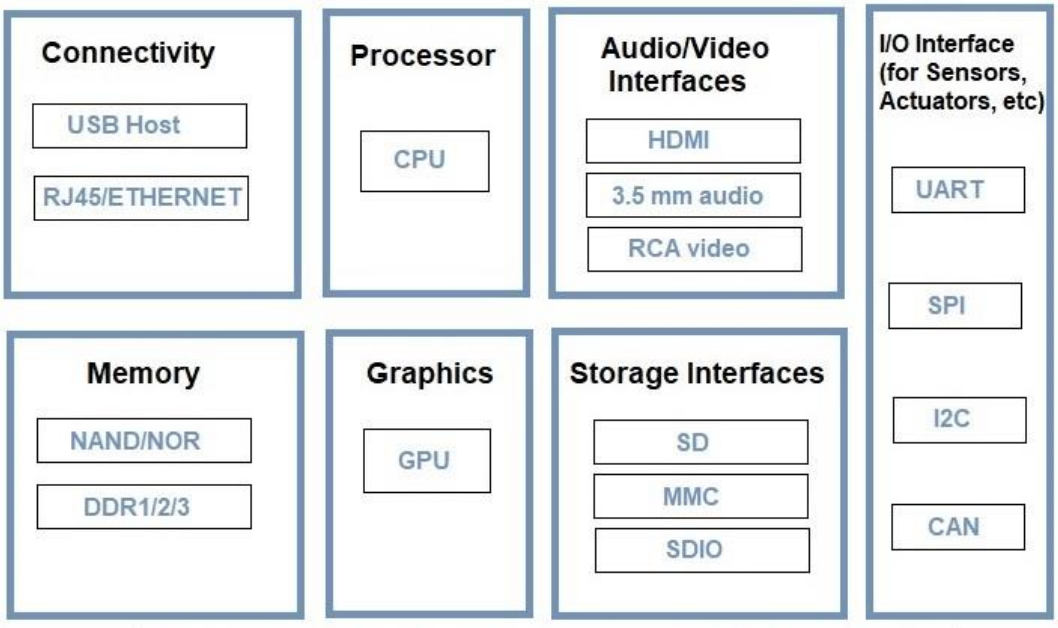
There are a wide variety of technologies that are associated with Internet of Things that facilitate in its successful functioning. IoT technologies possess the above-mentioned characteristics which create value and support human activities; they further enhance the capabilities of the IoT network by mutual cooperation and becoming the part of the total system.

1.3 Physical Design of IoT

Physical Design of IoT refers to IoT Devices and IoT Protocols. Things are Node device which have unique identities and can perform remote sensing, actuating and monitoring capabilities. Communication established between things and cloud based server over the Internet by various IoT protocols.

1. Things

Basically Things refers to IoT Devices which have unique identities and can perform remote sensing, actuating and monitoring capabilities. Things are is main part of IoT Application. IoT Devices can be various type, Sensing Devices, Smart Watches, Smart Electronics appliances, Wearable Sensors, Automobiles, and industrial machines. These devices generate data in some forms or the other which when processed by data analytics systems leads to useful information to guide further actions locally or remotely.



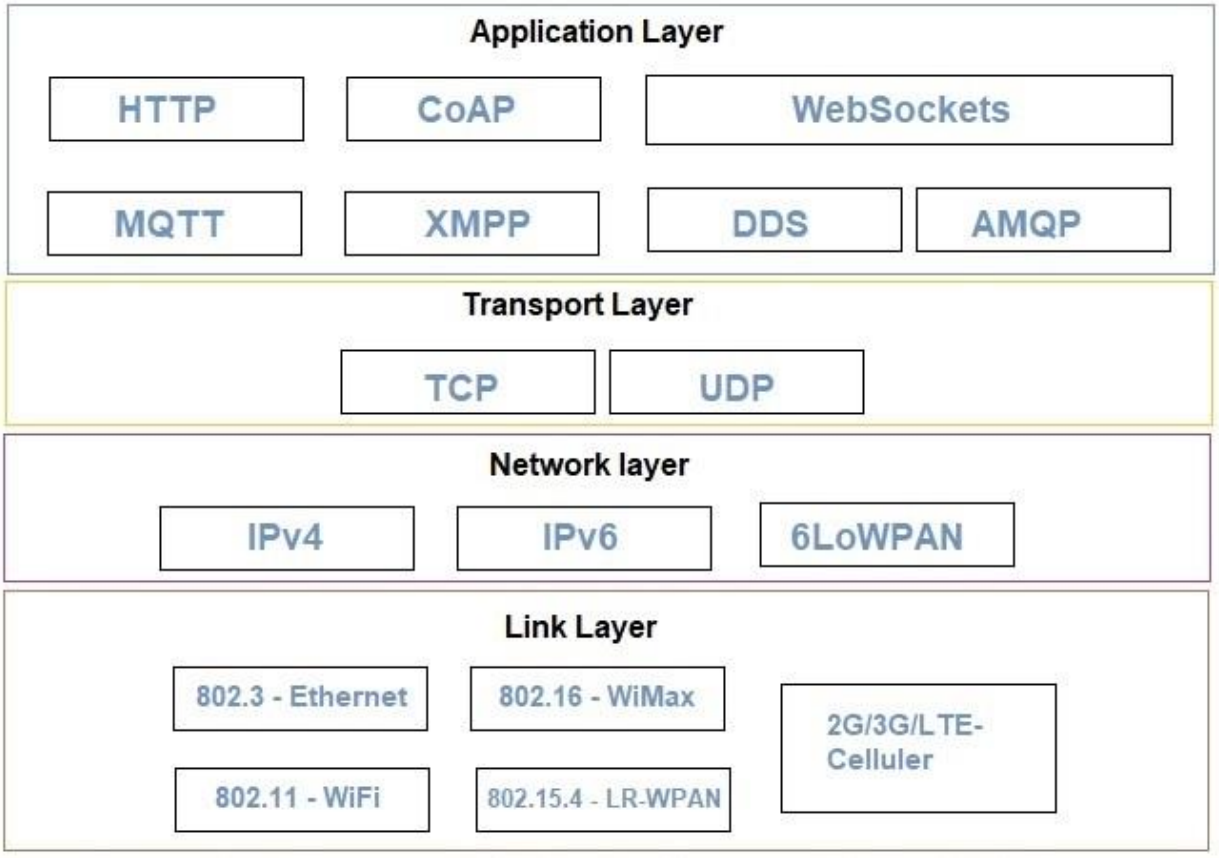
Generic Block Diagram of IoT Devices

For example, Temperature data generated by a Temperature Sensor in Home or other place, when processed can help in determining temperature and take action according to users. Above picture, shows a generic block diagram of IoT device. It may consist of several interfaces for connections to other devices. IoT Device has I/O interface for Sensors, Similarly for Internet connectivity, Storage and Audio/Video. IoT Device collect data from on-board or attached Sensors and Sensed data communicated either to other device or Cloud based sever. Today many cloud servers available for especially IoT System. These Platfrom known as IoT Platform. Actually these cloud especially design for IoT purpose. So here we can analysis and processed data easily.

How it works ? For example if relay switch connected to an IoT device can turn On/Off an appliance on the commands sent to the IoT device over the Internet.

2. IoT Protocols

IoT protocols help to establish Communication between IoT Device (Node Device) and Cloud based Server over the Internet. It helps to send commands to IoT Device and received data from an IoT device over the Internet. An image is given below. By this image you can understand which protocols used.



Link Layer

Link layer protocols determine how data is physically sent over the network's physical layer or medium (Coxial calbe or other or radio wave). This Layer determines how the packets are coded and signaled by the hardware device over the medium to which the host is attached (eg. coxial cable).

Here we explain some Link Layer Protocols:

- 802.3 – Ethernet : Ethernet is a set of technologies and protocols that are used primarily in LANs. It was first standardized in 1980s by IEEE 802.3 standard. IEEE 802.3 defines the physical layer and the medium access control (MAC) sub-layer of the data link layer for wired Ethernet networks. Ethernet is classified into two categories: classic Ethernet and switched Ethernet.
- 802.11 – WiFi : IEEE 802.11 is part of the IEEE 802 set of LAN protocols, and specifies the set of media access control (MAC) and physical layer (PHY) protocols for implementing wireless local area network (WLAN) Wi-Fi computer communication in various frequencies, including but not limited to 2.4 GHz, 5 GHz, and 60 GHz frequency bands.
- 802.16 – Wi-Max : The standard for WiMAX technology is a standard for Wireless Metropolitan Area Networks (WMANs) that has been developed by working group number 16 of IEEE 802, specializing in point-to-multipoint broadband wireless access.
- 802.15.4 -LR-WPAN : A collection of standards for Low-rate wireless personal area network. The IEEE's 802.15.4 standard defines the MAC and PHY layer used by, but not limited to, networking specifications such as Zigbee®, 6LoWPAN, Thread, WiSUN and MiWi™ protocols. The standards provide low-cost and low-speed communication for power constrained devices.
- 2G/3G/4G- Mobile Communication: These are different types of telecommunication generations. IoT devices are based on these standards can communicate over the cellular networks.

Network Layer

Responsible for sending of IP datagrams from the source network to the destination network. Network layer performs the host addressing and packet routing. We used IPv4 and IPv6 for Host identification. IPv4 and IPv6 are hierarchical IP addrssing schemes.

- IPv4 :An Internet Protocol address (IP address) is a numerical label assigned to each device connected to a computer network that uses the Internet Protocol for communication. An IP address serves two main functions: host or network interface identification and location addressing. Internet Protocol version 4 (IPv4) defines an IP address as a 32-bit number. However, because of the growth of the Internet and the

depletion of available IPv4 addresses, a new version of IP (IPv6), using 128 bits for the IP address, was standardized in 1998. IPv6 deployment has been ongoing since the mid-2000s.

- IPv6 : Internet Protocol version 6 (IPv6) is successor of IPv4. IPv6 was developed by the Internet Engineering Task Force (IETF) to deal with the long-anticipated problem of IPv4 address exhaustion. In December 1998, IPv6 became a Draft Standard for the IETF, who subsequently ratified it as an Internet Standard on 14 July 2017. IPv6 uses a 128-bit address, theoretically allowing 2128, or approximately 3.4×1038 addresses.
- 6LoWPAN : It is an acronym of IPv6 over Low-Power Wireless Personal Area Networks. 6LoWPAN is the name of a concluded working group in the Internet area of the IETF. This protocol allows for the smallest devices with limited processing ability to transmit information wirelessly using an internet protocol. 6LoWPAN can communicate with 802.15.4 devices as well as other types of devices on an IP network link like WiFi.

Transport Layer

This layer provides functions such as error control, segmentation, flow control and congestion control. So this layer protocols provide end-to-end message transfer capability independent of the underlying network.

- TCP : TCP (Transmission Control Protocol) is a standard that defines how to establish and maintain a network conversation through which application programs can exchange data. TCP works with the Internet Protocol (IP), which defines how computers send packets of data to each other. Together, TCP and IP are the basic rules defining the Internet. The Internet Engineering Task Force (IETF) defines TCP in the Request for Comment (RFC) standards document number 793.
- UDP : User Datagram Protocol (UDP) is a Transport Layer protocol. UDP is a part of Internet Protocol suite, referred as UDP/IP suite. Unlike TCP, it is unreliable and connectionless protocol. So, there is no need to establish connection prior to data transfer.

Application Layer

Application layer protocols define how the applications interface with the lower layer protocols to send over the network.

- HTTP : Hypertext Transfer Protocol (HTTP) is an application-layer protocol for transmitting hypermedia documents, such as HTML. It was designed for communication between web browsers and web servers, but it can also be used for other purposes. HTTP follows a classical client-server model, with a client opening a connection to make a request, then waiting until it receives a response. HTTP is a stateless protocol, meaning that the server does not keep any data (state) between two requests.

- CoAP : CoAP-Constrained Application Protocol is a specialized Internet Application Protocol for constrained devices, as defined in RFC 7252. It enables devices to communicate over the Internet. The protocol is especially targeted for constrained hardware such as 8-bits microcontrollers, low power sensors and similar devices that can't run on HTTP or TLS.
- WebSocket : The WebSocket Protocol enables two-way communication between a client running untrusted code in a controlled environment to a remote host that has opted-in to communications from that code. The security model used for this is the origin-based security model commonly used by web browsers.
- MQTT : MQTT is a machine-to-machine (M2M)/"Internet of Things" connectivity protocol. It was designed as an extremely lightweight publish/subscribe messaging transport and useful for connections with remote locations where a small code footprint is required and/or network bandwidth is at a premium.
- XMPP : Extensible Messaging and Presence Protocol (XMPP) is a communication protocol for message-oriented middleware based on XML (Extensible Markup Language). It enables the near-real-time exchange of structured yet extensible data between any two or more network entities.
- DDS : The Data Distribution Service (DDS™) is a middleware protocol and API standard for data-centric connectivity from the Object Management Group® (OMG®). It integrates the components of a system together, providing low-latency data connectivity, extreme reliability, and a scalable architecture that business and mission-critical Internet of Things (IoT) applications need.

1.4 LOGICAL Design of IoT

Refers to an abstract represent of entities and processes without going into the low level specifies of implementation.

1) IoT Functional Blocks

2) IoT Communication Models

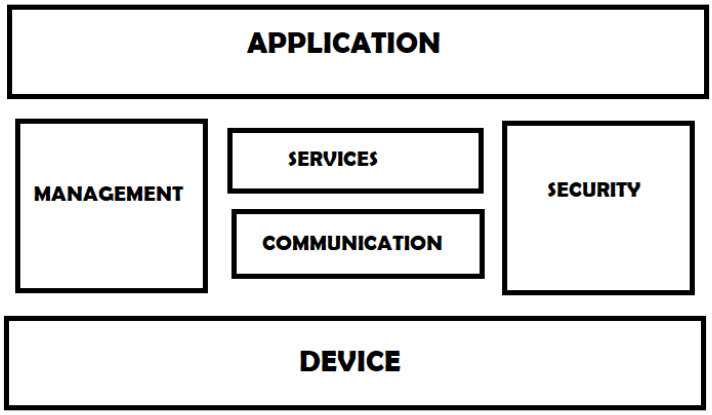
3) IoT Comm. APIs

IoT Functional Blocks

An IoT system comprises of a number of functional blocks that provide the system the capabilities for identification, sensing, actuation, communication and management.

Functional blocks are:

- **Device:** An IoT system comprises of devices that provide sensing, actuation and monitoring and control functions.
- **Communication:** Handles the communication for the IoT system.
- **Services:** services for device monitoring, device control service, data publishing services and services for device discovery.
- **Management:** This block provides various functions to govern the IoT system.
- **Security:** this block secures the IoT system and by providing functions such as authentication, authorization, message and content integrity, and data security.
- **Application:** This is an interface that the users can use to control and monitor various aspects of the IoT system. Application also allows users to view the system status and view or analyze the processed data.



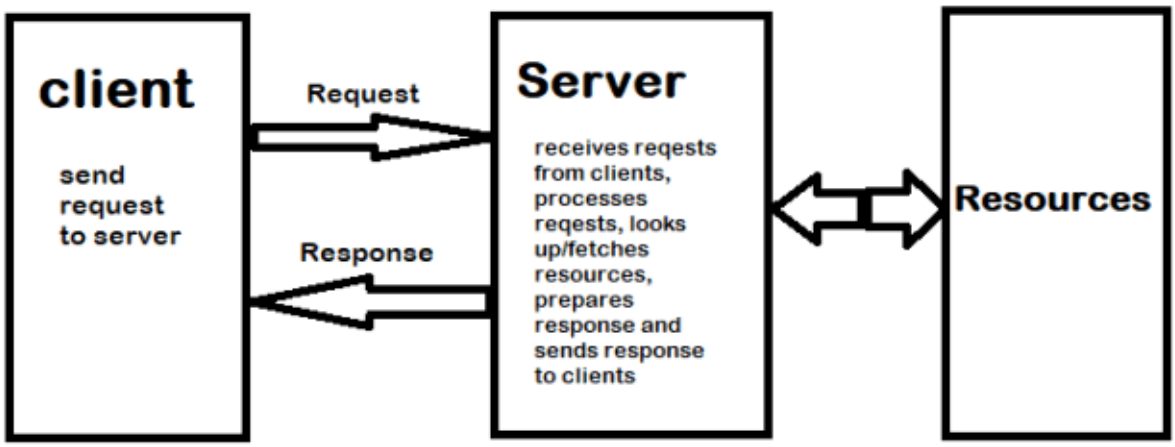
IoT Communication Models

Request-Response Model

Request-response model is communication model in which the client sends requests to the server and the server responds to the requests. When the server receives a request, it decides how to respond, fetches the data, retrieves resource representation, prepares the response, and then sends the response to the client. Request-response is a stateless communication model and each request-response pair is independent of others.

HTTP works as a request-response protocol between a client and server. A web browser may be the client, and an application on a computer that hosts a web site may be the server.

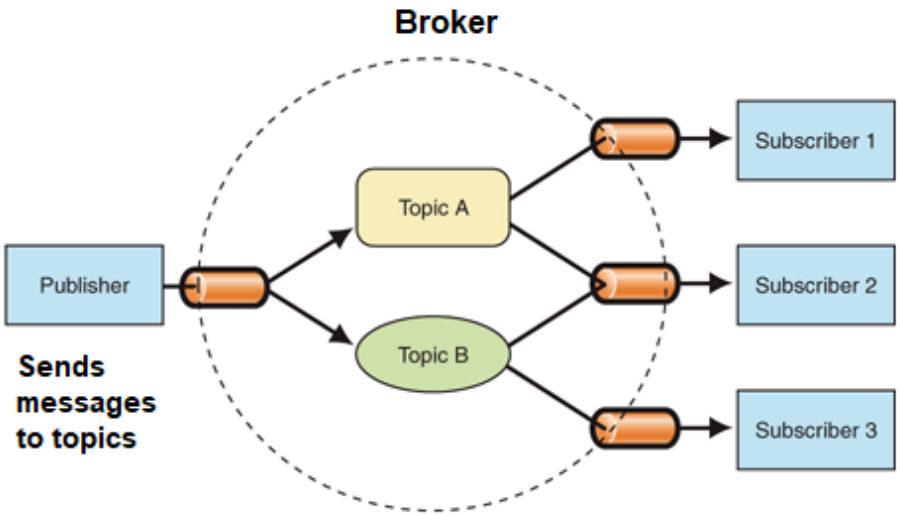
Example: A client (browser) submits an HTTP request to the server; then the server returns a response to the client. The response contains status information about the request and may also contain the requested content.



Request-Response Communication Model

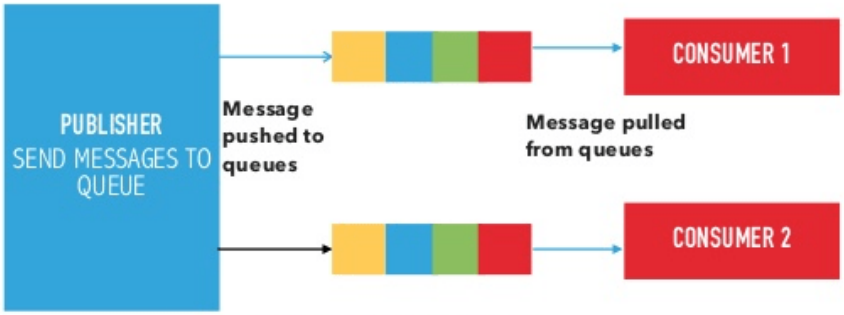
Publish-Subscribe Model

Publish-Subscribe is a communication model that involves publishers, brokers and consumers. Publishers are the source of data. Publishers send the data to the topics which are managed by the broker. Publishers are not aware of the consumers. Consumers subscribe to the topics which are managed by the broker. When the broker receive data for a topic from the publisher, it sends the data to all the subscribed consumers.



Push-Pull Model

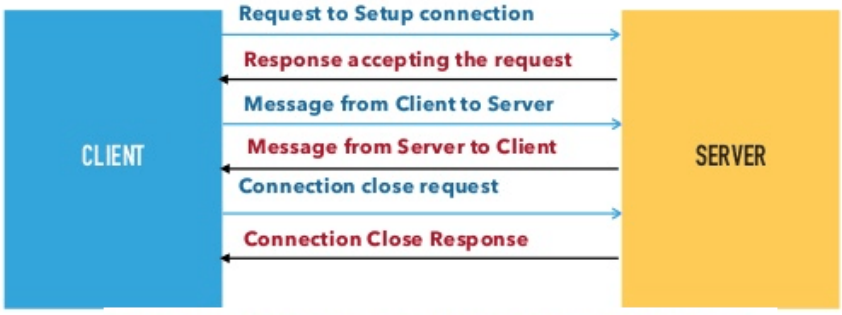
Push-Pull is a communication model in which the data producers push the data to queues and the consumers Pull the data from the Queues. Producers do not need to be aware of the consumers. Queues help in decoupling the messaging between the Producers and Consumers. Queues also act as a buffer which helps in situations when there is a mismatch between the rate at which the producers push data and the rate rate at which the consumer pull data.



PUSH PULL MODEL

Exclusive Pair Model

Exclusive Pair is a bidirectional, fully duplex communication model that uses a persistent connection between the client and server. Connection is setup it remains open until the client sends a request to close the connection. Client and server can send messages to each other after connection setup. Exclusive pair is stateful communication model and the server is aware of all the open connections.



EXCLUSIVE PAIR COMMUNICATION MODEL

IoT Communication APIs

Generally we used Two APIs For IoT Communication. These IoT Communication APIs are:

- REST-based Communication APIs
- WebSocket-based Communication APIs

REST-based Communication APIs

Representational state transfer (REST) is a set of architectural principles by which you can design Web services the Web APIs that focus on systems's resources and how resource states are addressed and transferred. REST APIs that follow the request response communication model, the rest architectural constraint apply to the components, connector and data elements, within a distributed hypermedia system. The rest architectural constraint are as follows:

Client-server – The principle behind the client-server constraint is the separation of concerns. for example clients should not be concerned with the storage of data which is concern of the serve. Similarly the server should not be concerned about the user interface, which is concern of the clien. Separation allows client and server to be independently developed and updated.

Stateless – Each request from client to server must contain all the information necessary to understand the request, and cannot take advantage of any stored context on the server. The session state is kept entirely on the client.

Cache-able – Cache constraints requires that the data within a response to a request be implicitly or explicitly leveled as cache-able or non cache-able. If a response is cache-able, then a client cache is given the right to reuse that repsonse data for later, equivalent requests. caching can partially or completely eliminate some instructions and improve efficiency and scalability.

Layered system – layered system constraints, constrains the behavior of components such that each component cannot see beyond the immediate layer with they are interacting. For example, the client cannot tell whether it is connected directly to the end server or two an intermediaryalong the way. System scalability can be improved by allowing intermediaries to respond to requests instead of the end server, without the client having to do anything different.

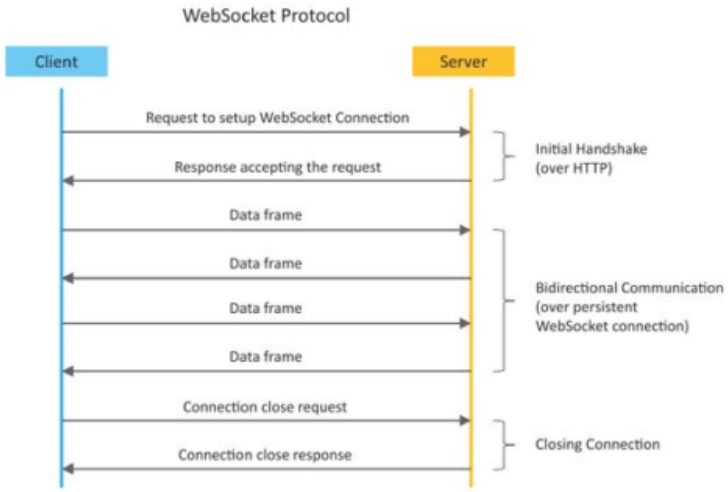
Uniform interface – uniform interface constraints requires that the method of communication between client and server must be uniform. Resources are identified in the requests (by URIsin web based systems) and are themselves is separate from the representations of the resources data returned to the client. When a client holds a representation of resources it has all the information required to update or delete the resource you (provided the client has required permissions). Each message includes enough information to describe how to process the message.

Code on demand – Servers can provide executable code or scripts for clients to execute in their context. this constraint is the only one that is optional.

A RESTful web service is a ” Web API ” implemented using HTTP and REST principles. REST is most popular IoT Communication APIs.

WebSocket based communication API

Websocket APIs allow bi-directional, full duplex communication between clients and servers. Websocket APIs follow the exclusive pair communication model. Unlike request-response model such as REST, the WebSocket APIs allow full duplex communication and do not require new coonection to be setup for each message to be sent. Websocket communication begins with a connection setup request sent by the client to the server. The request (called websocket handshake) is sent over HTTP and the server interprets it is an upgrade request. If the server supports websocket protocol, the server responds to the websocket handshake response. After the connection setup client and server can send data/mesages to each other in full duplex mode. Websocket API reduce the network traffic and letency as there is no overhead for connection setup and termination requests for each message. Websocket suitable for IoT applications that have low latency or high throughput requirements. So Web socket is most suitable IoT Communication APIs for IoT System.



1.5 M2M

Machine-to-Machine (M2M) refers to networking of machines(or devices) for the purpose of remote monitoring and control and data exchange.

- Term which is often synonymous with IoT is Machine-to-Machine (M2M).
- IoT and M2M are often used interchangeably.

Fig. Shows the end-to-end architecture of M2M systems comprises of M2M area networks, communication networks and application domain.

Machine-to-Machine is a network setup that enables devices to communicate freely with each other. This type of communication system is actively used in a variety of industries and has seen some major advancement in the last few decades, partially due to IP network systems facilitating communication between large numbers of devices over long distances.

Machine-to-machine (M2M) is a rather broad term, describing technology that enables devices on a network (typically cellular or wired network) to interact and exchange information. The system does not necessarily rely on human intervention during the communication process. Utilising M2M offers some distinctive benefits over other communication models:

- It is relatively easy to roll out and low maintenance.
- Can utilise both mobile and wired networks, indoors and outdoors.
- Provides high range
- Relatively low latency
- Energy and cost-efficient
- Large collection and processing of data is made possible
- Reducing human interference decreases the risk of human error

1.6 Difference between IoT and M2M

BASIS OF	IOT	M2M
Abbreviation	Internet of Things	Machine to Machine
Intelligence	Devices have objects that are responsible for decision making	Some degree of intelligence is observed in this
Connection type used	The connection is via Network and using various communication types.	The connection is a point to point
Communication protocol used	Internet protocols are used such as HTTP, FTP, and Telnet.	Traditional protocols and communication technology

BASIS OF	IOT	M2M
		techniques are used
Data Sharing	Data is shared between other applications that are used to improve the end-user experience.	Data is shared with only the communicating parties.
Internet	Internet connection is required for communication	Devices are not dependent on the Internet.
Scope	A large number of devices yet scope is large.	Limited Scope for devices.
Business Type used	Business 2 Business(B2B) and Business 2 Consumer(B2C)	Business 2 Business (B2B)
Open API support	Supports Open API integrations.	There is no support for Open Api's
Examples	Smart wearables, Big Data and Cloud, etc.	Sensors, Data and Information, etc.

1.7 Software Define Network

Software-defined networking (SDN) technology is an approach to network management that enables dynamic, programmatically efficient network configuration in order to improve network performance and monitoring, making it more like cloud computing than traditional network management.[1] SDN is meant to address the fact that the static architecture of traditional networks is decentralized and complex while current networks require more flexibility and easy troubleshooting. SDN attempts to centralize network intelligence in one network component by disassociating the forwarding process of network packets (data plane) from the routing process (control plane). The control plane consists of one or more controllers, which are considered the brain of the SDN network where the whole intelligence is incorporated. However, the intelligent centralization has its own drawbacks when it comes to security, scalability and elasticity and this is the main issue of SDN.

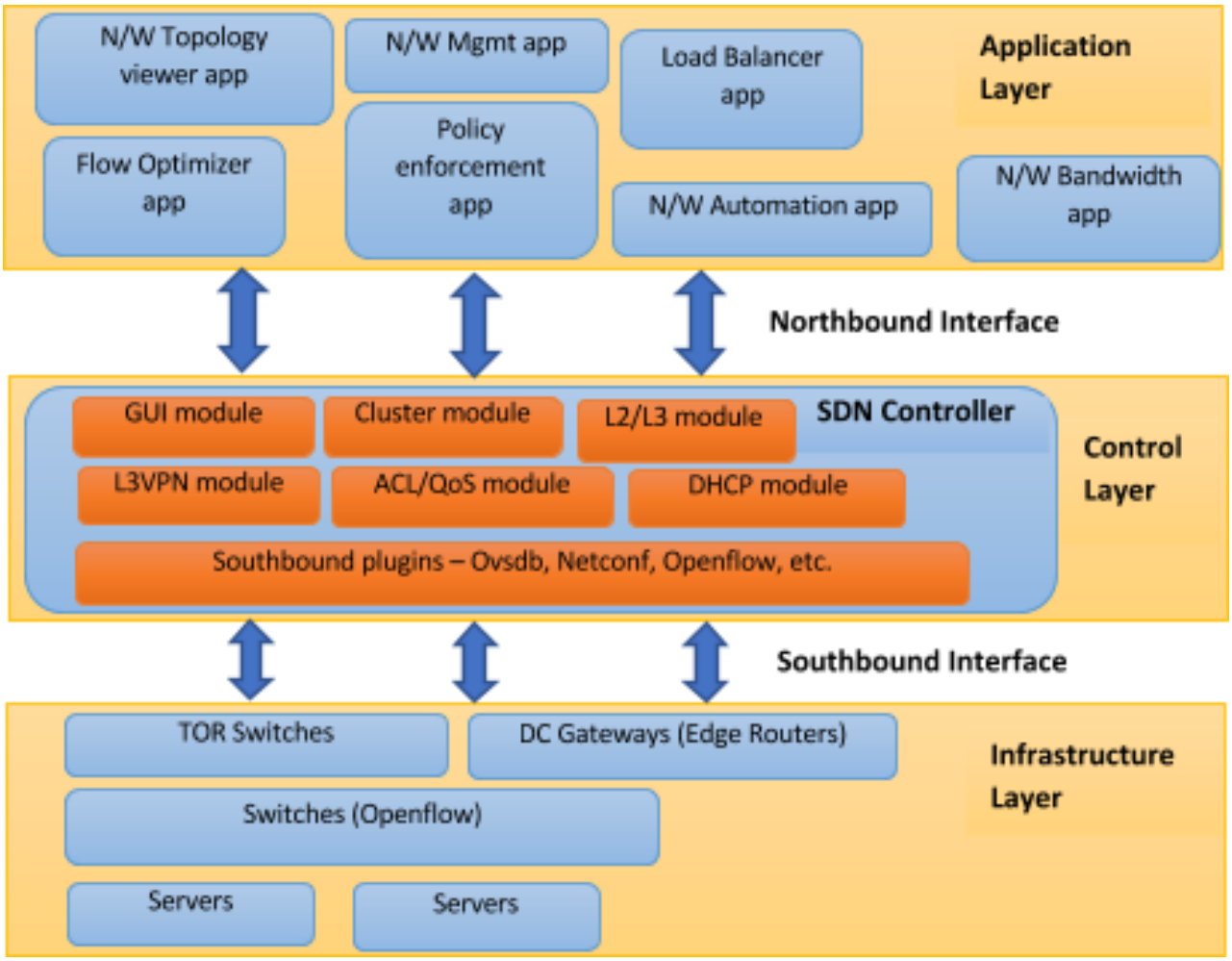
SDN was commonly associated with the OpenFlow protocol (for remote communication with network plane elements for the purpose of determining the path of network packets across

network switches) since the latter's emergence in 2011. However, since 2012 OpenFlow for many companies is no longer an exclusive solution, they added proprietary techniques. These include Cisco Systems' Open Network Environment and Nicira's network virtualization platform.SD-WAN applies similar technology to a wide area network (WAN).

SDN technology is currently available for industrial control applications that require extremely fast failover. One company boasts 100x Faster Failover for Mission-critical processes (fails over in less than 100 µs, compared to 10 ms for traditional networks) along with the elimination of certain Cyber Vulnerabilities that are associated with traditional network management switches.

SDN broadly consists of three layers:

1. Application layer
2. Control layer
3. Infrastructure layer



Let us try and understand these layers in bottom-to-up approach.

Infrastructure layer is composed of various networking equipment which forms underlying network to forward network traffic. It could be a set of network switches and routers in the data centre. This layer would be the physical one over which network virtualization would be laid down through the control layer (where SDN controllers would sit and manage underlying physical network).

Control layer is the land of control plane where intelligent logic in SDN controllers would reside to control network infrastructure. This is the area where every network vendor is working to come up with their own products for SDN controller and framework. Here in this layer, a lot of business logic is being written in controller to fetch and maintain different types of network information, state details, topology details, statistics details, and more.

Application layer is open area to develop as much innovative application as possible by leveraging all the network information about network topology, network state, network statistics, etc. There can be several types of applications which can be developed like those related to network automation, network configuration and management, network monitoring, network troubleshooting, network policies and security. Such SDN applications can provide various end-to-end solutions for real world enterprise and data centre networks. Network vendors are coming up with their set of SDN applications.

1.8 Challenges in IoT

- **Scaling**
  Scaling IoT device networks is a major issue for companies. While many companies can manage overall scaling issues, scaling the footprint for IoT presents some new challenges. Leaders should also be aware of issues in working with software, legacy devices, and the need for specialized solutions. Determining the departments responsible for IoT-related gateways, analytics, networking equipment, sensors, and software and hardware is also essential.
- **Implementation**
  Another challenge for many leaders is implementing IoT solutions. One of the main hurdles has been lack of support for deploying production quality. There is also the question of whether IT, operations, the C-suite, or some other department should handle implementation.
- **Data protection**
  Figuring out connectivity management is also a challenge. Executives face the issue of whether they should integrate connectivity management across any kind of environment. Additionally, determining if connecting the IoT ecosystem to disparate networks is required was also a factor. In a survey by IoT world, 36.73 percent of executives said that

they were not confident “their organization could secure and protect all data within their IoT ecosystem.”

- **Security**
  Executives also noted security in the IoT world survey, with 72.16 percent of leaders stating that they embed and incorporate security into the product life and design of IoT devices. However, companies are still struggling with IoT security. The IoT world survey found that more than one-third of executives will not be staying with security patches and fixes, while 57.6 percent said they had not trained IT staff on the latest IoT security updates. Many executives do not even conduct vulnerability testing to identify network weaknesses.
- **Using blockchain to make IoT deployments**
  Many companies are considering the use of ledger technology for creating more secure and efficient IoT deployments. Some of the most important reasons for leveraging blockchain are to reduce the risk of collusion and tampering, developing trust for users, accelerating the rate of data transactions, and reducing IoT overhead costs.

NETWORK AND COMMUNICATION ASPECTS

2.1 Wireless Mediam Access Issues

When it comes to communication using a wireless medium there is always a concern about the interference due to other present wireless communication technologies. Wireless means communication and message transfer without the use of physical medium i.e., wires.

Let us understand how communication is done between them. Different Mobile stations (MS) are attached to a transmitter/receiver which communicates via a shared channel by other nodes. In this type of communication, it makes it difficult for the MAC design rather than the wire line networks.

The very important issues which are observed are: Half Duplex operation, Time-varying channel, and Burst channel errors.

These are explained as following below.

1. **Half Duplex operation:**
   Half-duplex transmission means when the sender and receiver both are capable of sharing data but one at a time. In wireless transmission, it is difficult to receive data when the transmitter is sending the data because during transmission a large amount or a large fraction of signal energy is leaked while broadcasting. The magnitude of the transferred signal and received signal differs a lot. Due to which collision detection is even not possible by the sender as the intensity of the transferred signal is large than the received one. Hence this causes the problem of collision and the prime focus should be to minimize the collision

2. **Time-varying channel:**
   Time-varying channels include the three mechanisms for radio signal propagations they are Reflection, Diffraction, and Scattering.

- **Reflection** –
  This occurs when a propagating wave carrying information intrudes on an object that has very large dimensions than the wavelength of the wave.
- **Diffraction** –
  This occurs when the radio path between the transmitter and the receiver is collided by the surface with sharp edges. This is a phenomenon which causes the diffraction of the wave from the targeted position.
- **Scattering** –
  This occurs when the medium through from the wave is traveling consists of some objects which have dimensions smaller than the wavelength of the wave.

While transmitting the signal by the node these are time shifted and this is called multipath propagation. While when this node signals intensity is dropped below a threshold value, then this

is termed as fade. As a result Handshaking strategy is widely used so as a healthy communication can be set up.

3. **Burst channel errors :**

   Burst channel errors are called as a contiguous sequence of symbols, which are received in a communication channel, in which the first and last symbols has an error and there is no evidence of contiguous sub-sequence of corrected received symbols. When time-varying channels are used then signals strengths are introduced due to which errors are observed in transmission. For these channels in wire line networks, the Bit rate is high as 10^{-3}.

2.2 MAC Protocol

In IEEE 802 LAN/MAN standards, the medium access control (MAC, also called media access control[1]) sublayer is the layer that controls the hardware responsible for interaction with the wired, optical or wireless transmission medium. The MAC sublayer and the logical link control (LLC) sublayer together make up the data link layer. Within the data link layer, the LLC provides flow control and multiplexing for the logical link (i.e. EtherType, 802.1Q VLAN tag etc), while the MAC provides flow control and multiplexing for the transmission medium.

These two sublayers together correspond to layer 2 of the OSI model. For compatibility reasons, LLC is optional for implementations of IEEE 802.3 (the frames are then "raw"), but compulsory for implementations of other IEEE 802 physical layer standards. Within the hierarchy of the OSI model and IEEE 802 standards, the MAC sublayer provides a control abstraction of the physical layer such that the complexities of physical link control are invisible to the LLC and upper layers of the network stack. Thus any LLC sublayer (and higher layers) may be used with any MAC. In turn, the medium access control block is formally connected to the PHY via a media-independent interface. Although the MAC block is today typically integrated with the PHY within the same device package, historically any MAC could be used with any PHY, independent of the transmission medium.

When sending data to another device on the network, the MAC sublayer encapsulates higher-level frames into frames appropriate for the transmission medium (i.e. the MAC adds a syncword preamble and also padding if necessary), adds a frame check sequence to identify transmission errors, and then forwards the data to the physical layer as soon as the appropriate channel access method permits it. For topologies with a collision domain (bus, ring, mesh, point-to-multipoint topologies), controlling when data is sent and when to wait is necessary to avoid collisions. Additionally, the MAC is also responsible for compensating for collisions by initiating retransmission if a jam signal is detected. When receiving data from the physical layer, the MAC block ensures data integrity by verifying the sender's frame check sequences, and strips off the sender's preamble and padding before passing the data up to the higher layers.

Functions performed in the MAC sublayer

According to IEEE Std 802-2001 section 6.2.3 "MAC sublayer", the primary functions performed by the MAC layer are:

- Frame delimiting and recognition
- Addressing of destination stations (both as individual stations and as groups of stations)
- Conveyance of source-station addressing information
- Transparent data transfer of LLC PDUs, or of equivalent information in the Ethernet sublayer
- Protection against errors, generally by means of generating and checking frame check sequences
- Control of access to the physical transmission medium

In the case of Ethernet, according to 802.3-2002 section 4.1.4, the functions required of a MAC are:

- receive/transmit normal frames
- half-duplex retransmission and backoff functions
- append/check FCS (frame check sequence)
- interframe gap enforcement
- discard malformed frames
- prepend(tx)/remove(rx) preamble, SFD (start frame delimiter), and padding
- half-duplex compatibility: append(tx)/remove(rx) MAC address.

2.3 Routing protocols

Dynamic routes are routes learned via routing protocols. Routing protocols are configured on routers with the purpose of exchanging routing information. There are many benefits of using routing protocols in your network, such as:

- unlike static routing, you don’t need to manually configure every route on each router in the network. You just need to configure the networks to be advertised on a router directly connected to them.
- if a link fails and the network topology changes, routers can advertise that some routes have failed and pick a new route to that network.

Types of routing protocols

There are two types of routing protocols:

1. Distance vector (RIP, IGRP)

2. Link state (OSPF, IS-IS)

Cisco has created its own routing protocol – EIGRP. EIGRP is considered to be an advanced distance vector protocol, although some materials erroneously state that EIGRP is a hybrid routing protocol, a combination of distance vector and link state.

All of the routing protocols mentioned above are interior routing protocols (IGP), which means that they are used to exchange routing information within one autonomous system. BGP (Border Gateway Protocol) is an example of an exterior routing protocol (EGP) which is used to exchange routing information between autonomous systems on the Internet.

Distance vector protocols

As the name implies, distance vector routing protocols use distance to determine the best path to a remote network. The distance is something like the number of hops (routers) to the destination network.

Distance vector protocols usually send the complete routing table to each neighbor (a neighbor is directly connected router that runs the same routing protocol). They employ some version of Bellman-Ford algorithm to calculate the best routes. Compared with link state routing protocols, distance vector protocols are easier to configure and require little management, but are susceptible to routing loops and converge slower than the link state routing protocols. Distance vector protocols also use more bandwidth because they send complete routing table, while the link state procotols send specific updates only when topology changes occur.

RIP and EIGRP are examples of distance vector routing protocols.

Link state protocols

Link state routing protocols are the second type of routing protocols. They have the same basic purpose as distance vector protocols, to find a best path to a destination, but use different methods to do so. Unlike distance vector protocols, link state protocols don’t advertise the entire routing table. Instead, they advertise information about a network toplogy (directly connected links, neighboring routers…), so that in the end all routers running a link state protocol have the same topology database. Link state routing protocols converge much faster than distance vector routing protocols, support classless routing, send updates using multicast addresses and use triggered routing updates. They also require more router CPU and memory usage than distance-vector routing protocols and can be harder to configure.

Each router running a link state routing protocol creates three different tables:

- neighbor table – the table of neighboring routers running the same link state routing protocol.
- topology table – the table that stores the topology of the entire network.
- routing table – the table that stores the best routes.

Shortest Path First algorithm is used to calculate the best route. OSPF and IS-IS are examples of link state routing protocols.

Difference between distance vector and link state routing protocols

The following table summarizes the differences:

Distance vector	Link state
sends the entire routing table	sends only link state information
slow convergence	fast convergence
susceptible to routing loops	less susceptible to routing loops
updates are sometimes sent using broadcast	always uses multicast for the routing updates
doesn't know the network topology	knows the entire network topology
simpler to configure	can be harder to configure
examples: RIP, IGRP	examples: OSPF, IS-IS

2.4 Sensor deployment & Node discovery

The Internet of Things (IoT) is going to change our world.

It is estimated that there will be nearly 22 billion IoT devices by 2025. Extending internet connectivity to everyday objects will transform industries and create tremendous cost savings.

But how do non-internet-enabled devices gain connectivity capabilities?

Through wireless sensors.

With wireless sensors, the IoT is possible. Individuals and organizations can use wireless sensors to enable many different kinds of smart applications. From interconnected homes to smart cities, wireless sensors create the infrastructure upon which the IoT comes alive.

Understanding how wireless sensor technology works is crucial for anyone who intends to deploy IoT applications in the future. Through this article, you will gain a fundamental understanding of how wireless sensors work, emerging wireless standards for sensors, and what role they will play in the future.

- What is a wireless sensor?
- What are some examples of wireless sensors?
- What are the different types of wireless network topologies?
- What are the traditional wireless sensor protocols?
- What are the new and emerging LPWAN standards for wireless sensors?
- How does wireless sensor technology fit into the Internet of Things?
- Who provides wireless sensor solutions?
- Deploy wireless sensors for any IoT application with Radio Bridge

What is a wireless sensor?

A wireless sensor is a device that can gather sensory information and detect changes in local environments.

Sensors are designed to measure specific parameters about their physical surroundings and produce outputs, often electrical signals, for further processing. These parameters include many different types of stimuli, including air temperature, lighting levels, movements, and liquid leakages.

Due to the fact that wireless sensors don't actually perform heavy data processing locally, they consume very little power and can last years on a single battery if an optimal wireless technology is used. Additionally, sensors are easily supported on low-speed networks as they transmit very light data loads.

Passive sensors are self-powered devices that respond to inputs from surrounding environments. They don't actively probe and, therefore, do not require an external energy source. An example of a passive sensor would be a mercury-based thermometer that rises and falls with temperature but does not require external power. Active sensors, on the other hand, rely on external power to continually monitor local environments. Examples of active sensors include devices that use radar or sonar in order to probe surroundings. Active sensors are the primary topic of this article.

Wireless sensors can be grouped together in order to monitor environmental conditions throughout a region. These wireless sensor networks consist of many spatially dispersed sensors that communicate through wireless connections. Sensors in a common network share data either through nodes that consolidate information at a gateway, or where each sensor connects directly to the gateway assuming it can achieve the necessary range. Gateways act as

bridges that connect local sensors to the internet, functioning both as routers and wireless access points.

What are some examples of wireless sensors?

Wireless sensors come in all shapes and sizes. They can be used for many purposes, from detecting movement to monitoring air quality.

Wireless proximity sensors detect the presence or absence of different types of objects. There are several categories of proximity sensors that include inductive, ultrasonic, infrared, microwave, laser, pulse radar, and RF time-of-flight / doppler sensors. Some of the most common proximity sensors use hall effect sensors or reed switches to detect the presence of a magnet.

Inductive proximity wireless sensors require a metal target, which makes them useful for activities, such as monitoring industrial processes and optimizing traffic flow. Stadiums with retractable roofs may use inductive sensors to ensure that systems are working properly. At street intersections, inductive sensors can be buried within pavement to register when cars are waiting for the lights to change.

Capacitive proximity sensors can detect many kinds of materials, from liquids to rocks. As a result of their versatility, companies might use capacitive sensors to monitor hopper powder levels or rice and soybean presence on grain elevators. These sensors can also be used in a similar fashion to liquid detection sensors by monitoring tank levels.

Using high-frequency sound waves, ultrasonic proximity sensors can determine distances between two objects in harsh conditions. Ultrasonic sensors can be used to detect cars in automated car washes or obstructions in the way of garage door paths.

Wireless movement sensors can detect certain types of motion, including acceleration and tilt. For example, wireless movement sensor alarms can be configured to alert owners when their valuable assets or packages are on the move. With acceleration-based sensors, companies can monitor the movement of assets or measure impact forces. PIR motion detectors use infrared light to detect movement within a room. PIR sensors are often used in security and people detection applications.

Vibration sensors can help determine how much a machine is in use or monitor pump activity. For instance, the piezoelectric strip from the vibration sensor can be attached to a motor to detect on / off activity or to detect if a bearing is starting to fail. Building security firms may also use these sensors to quickly learn of window breaks or shattered glass on their properties.

Wireless liquid sensors detect the presence of liquids, such as water or fuel. A flow sensor can monitor when water movement has stopped, indicating potentially frozen pipes. Water sensor wireless alerts can tell homeowners when there is flooding or an overflow of some kind. Hotel owners can install water leak sensors throughout their establishments to identify leaks as soon as

they occur. Additionally, these sensors can be deployed to register dangerous fuel leaks at industrial sites. For companies or organizations that measure reservoir levels, ultrasonic liquid level sensors can be used to send alerts when levels are too low or too high.

Wireless push buttons transmit signals when buttons are pushed, which is particularly important in emergency health or personal situations. In this type of application, it is referred to as a Personal Emergency Response System (also called PERS) in which users can push one of these wireless panic buttons when they require assistance of any kind. Wireless push button sensors can also be used in customer service settings when guests need to be able to request assistance. In public restrooms, wireless push buttons can also function as service call buttons that indicate when cleanings are needed.

Wireless air sensors are used to assess air quality, including temperature, humidity, and the presence of harmful gases. Using wireless air temperature and humidity sensors, homeowners can detect heat system failures or measure humidity levels in greenhouses or in-home saunas. With air sensors, art gallery owners can carefully regulate room temperatures and restaurants can detect refrigeration failures before it's too late. Additionally, air sensors can be used to monitor HVAC efficiency and detect when air flow is hindered.

Wireless temperature sensors can not only track temperatures in the surrounding environment, but can also measure material temperatures. For example, probe-type temperature sensors can detect frozen pipes and warn homeowners that they need to take preemptive measures to avoid future problems. Warehouses will use air temp and humidity sensors measure heat index values to ensure the safety of their employees.

There are also temperature sensors specifically designed for in-water applications. Aquariums can use wireless water temperature sensors to monitor tank temperatures. Amusement parks and recreation centers can also track pool temperatures and make sure conditions are comfortable for guests.

Wireless optical sensors can detect optical events involving light levels or optical beam crossings. With light detection sensors, greenhouses can track solar and light levels throughout the day. Museums can configure light metering sensors to warn supervisors of when there is too much harmful UV radiation present. With ambient light sensors, homeowners are able to detect the presence of flames inside or automate lighting systems in order to reduce energy costs. PIR infrared motion sensors are often used in security applications or for people detection.

How does wireless sensor technology fit into the Internet of Things?

The success of the Internet of Things is highly dependent on wireless sensor technology. Wireless sensors will enable many major IoT applications across a wide variety of sectors and settings. With such a vast array of sensors available, IoT possibilities are virtually endless.

Smart homes, smart buildings, smart cities, smart agriculture and supply chain management are just a few of the areas that will be dramatically impacted by wireless sensor networks. Below are several specific use cases that demonstrate how versatile and important wireless sensors will be in the future.

Connecting everyday objects in smart homes

In a single home, residents may use a variety of wireless sensors to track surrounding conditions and bolster security. For example, a homeowner could install wireless door or window sensors throughout the house to monitor opening and closing events at all times. These same sensors can be used to detect activity around lock boxes and cabinets that hold valuables or private information. Homeowners can also install wireless dry contact sensors in doorbells or other remote control units throughout the house that send alerts when certain buttons are pushed. With wireless water sensors, leaking toilets or pipe issues can be quickly rectified before basements are flooded and carpets are ruined.

Improving service company responsiveness and effectiveness

Both plumbing and insurance companies can also benefit from wireless water leak sensors. Plumbing service providers could install these sensors at condo complexes and apartment buildings so that they are automatically warned when leaks occur in any units. Insurance firms could also deploy leak sensors in homes to minimize any damages that may occur from accidental flooding.

Supporting patient health care via real-time monitoring

At senior care facilities, wireless push buttons are particularly important as they can be configured to act like mobile PERS devices that will warn staff when help is needed. Just like in smart homes, these facilities can also use wireless door or window sensors to detect when residents are trying to leave their rooms unattended.

Enabling better and smarter product management

Grocery stores and retailers can protect assets with different types of wireless sensors located throughout their premises. With wireless air temperature sensors, facilities managers can track temperature levels in refrigeration units and make sure that perishables are safe. By using wireless temperature sensors, they can also measure the temperature of refrigerated foods, thus creating an effective cold chain monitoring system. For those with large glass window displays, wireless vibration sensors can be extremely useful for detecting glass breaks instantly and warning the appropriate authorities.

Improving security and safety in industrial settings

At car dealerships, fleet managers can install wireless acceleration-based movement sensors in vehicles so that they can receive notifications when cars move at night, a sign of potential theft. In warehouses, facility supervisors can calculate heat indexes with wireless air sensors and make sure they maintain healthy work environments for employees. Additionally, tilt sensors could be deployed on loading dock doors to detect opening and closing events.

Preserving and maintaining fragile artwork

In museums and art galleries, preservation specialists can place wireless humidity sensors in rooms to monitor and adjust air conditions in order to protect artifacts or artwork. Optical sensors could also be used to detect lighting levels and ensure that guests have the best possible viewing experiences.

Protecting and fixing local infrastructure quickly

Utility companies can install high-temperature probes on utility poles in order to detect transformer failures. With tilt sensors, they could also configure sensors to send alerts to maintenance personnel when poles are leaning or have been struck by vehicles.

These are just a few examples of how wireless sensors will enable IoT applications in our daily lives. Over time, we will continue to see innovation across different industries and other useful applications for the technology.

2.5 Data aggregation & dissemination

Data aggregation is the process of gathering data and presenting it in a summarized format. The data may be gathered from multiple data sources with the intent of combining these data sources into a summary for data analysis. This is a crucial step, since the accuracy of insights from data analysis depends heavily on the amount and quality of data used. It is important to gather high-quality accurate data and a large enough amount to create relevant results. Data aggregation is useful for everything from finance or business strategy decisions to product, pricing, operations, and marketing strategies.

What is an example of aggregate data?

Here is an example of aggregate data in business:

Companies often collect data on their online customers and website visitors. The aggregate data would include statistics on customer demographic and behavior metrics, such as average age or

number of transactions. This aggregated data can be used by the marketing team to personalize messaging, offers, and more in the user's digital experience with the brand. It can also be used by the product team to learn which products are successful and which are not. And furthermore, the data can also be used by company executives and finance teams to help them choose how to allocate budget towards marketing or product development strategies.

What is data aggregation in the financial and investing sectors?

Finance and investment firms are increasingly basing their recommendations on alternative data. A large portion of that data comes from the news, since investors need to stay up-to-date on industry and company financial trends. So, financial firms can use data aggregation to gather headlines and article copy and use that data for predictive analytics, to find trends, events, and shifting views that could affect the finances of the companies and products they are tracking.

This market information is available on news websites for free, but it is spread across hundreds of websites. Combing through each individual website manually is time-consuming and may produce unreliable datasets due to missing data. We'll talk more about how financial and investment firms can speed up the process in this use case at the end of this post.

What is data aggregation in the retail industry?

The retail and ecommerce industries have many possible uses for data aggregation. One is competitive price monitoring. Competitive research is necessary to be successful in the ecommerce and retail space. Companies have to know what they're up against. So, they must always be gathering new information about their competitors' product offerings, promotions, and prices. This data can be pulled from competitor's websites or from other sites their products are listed on. In order to get accurate information, the data needs to be aggregated from every single relevant source. That's a tall order for manual web data analysis.

Another way retail and ecommerce companies use data aggregation is to gather images and product descriptions to use on their site. These often come from manufacturers, and it is much easier to reuse the already-existing images and descriptions from them than to craft your own. Manually gathering product listings or competitor prices is time consuming and makes it almost impossible to make sure it is constantly up-to-date. After we take a look at the travel industry, we'll tell you how retail and ecommerce companies can aggregate and combine data more efficiently.

What is data aggregation in the travel industry?

Data aggregation can be used for a wide range of purposes in the travel industry. These include competitive price monitoring, competitor research, gaining market intelligence, customer sentiment analysis, and capturing images and descriptions for the services on their online travel sites. Competition in the online travel industry is fierce, so data aggregation or the lack thereof can make or break a travel company.

Travel companies need to keep up with the ever-changing travel costs and property availability. They also need to know which destinations are trending and which audiences they should target with their travel offers. The data needed to gain these insights is spread across many places on the internet, making it difficult to gather manually. That's where our data extraction and aggregation service, Web Data Integration, comes in.

Data Aggregation with Web Data Integration

Web Data Integration (WDI) is a solution to the time-consuming nature of web data mining. WDI can extract data from any website your organization needs to reach. Applied to the use cases previously discussed or to any field, Web Data Integration can cut the time it takes to aggregate data down to minutes and increase accuracy by eradicating human error in the data aggregation process. This allows companies to get the data they need, when they need it, from wherever they need it. All with built-in quality control to ensure accuracy.

WDI not only extracts and aggregates the data you need, it also prepares and cleans the data and delivers it in a consumable format for integration, discovery and analysis. So, if your company needs accurate, up-to-date data from the web, Web Data Integration is right for you.

WEB OF THINGS

3.1 Web of Things vs Internet of things

- From the developers 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 API s, which enable s 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.
- While IoT is about creating a network of objects, things, people, system and applications, WoT tries to integrate them to Web.
- Technically speaking WoT can be thought as flavor/Option of an application layer added over the IoT‘s network layer.
- However, the scope of the Internet of things applications is broader and includes systems that not accessible through the web (e.g. conventional WSN and RFID system)

Example Of Web of Things (WoT):

To illustrate how the Web of Things can deal with IoT limitations, let’s consider, the Smith, The owner of a famous hotel chain in several cities around the world.

Smith would like to digitally connect all the appliances in all the rooms of his hotels, so that he is able to monitor, control, and improve the management of his hotels from any place using a single Web-based “hotel control center” application.

At the same time, this system could also offer a more pleasant and personalized experience to each guest in his hotels, this can be done with Web of Things (WoT).

In the Web of Things (WoT), any device can be accessed using standard Web protocols.

Connecting different/diverse devices to the Web makes the integration across systems and applications much simpler.

3.2 Pillars of web

The Four Pillars of the Web of Things

Over the years, I have noticed a fundamental dissonance between visions of the future and the reality that unfolds. We are presented visions of conformists walking orderly, antiseptic halls in silver spandex and end up with piercings, tattoos and Foosball at the office. As technology advances, people seek more control over it to express their individuality.

That's exactly what's happening now. While the virtual Web made documents universal, the Web of Things is making machines interoperable and allowing consumers to tap into the Internet of Things. The technology centers on four pillars:

Smartphones: At the center of the Web of Things is our smartphones. We carry around more processing power than the Apollo program employed to put a man on the moon and we are increasingly using it as a universal remote control for our environment.

The phone itself is a sensor platform. Apps like Shazam and Viggle are able to recognize the media you are watching or listening to and serve related content to your smartphone or tablet. Cameras are able to recognize faces and objects and then search for related information while GPS notes our location and that of objects around us.

Most of all, we are using our smartphones to interact with other elements of the Web of Things, like Smart Homes, Smart Cars and Smart Retail,

Smart Homes: New super-efficient chips are putting connectivity everywhere and our home appliances will be as much a part of the Web of Things as our tablets or smartphones. This isn't a new idea, we've been hearing about "refrigerators that order your milk" for years now. However, what's emerging is profoundly different.

At CES 2012, Motorola showed of their 4Home system, which can sync any device with your smartphone. You can monitor your home through video feeds, control your home security, manage your energy output and preheat the oven from the car. Again, the vision was of technology running everything itself, the reality will be more control for consumers.

The smart home concept is still in its infancy, but with a little imagination we can see the possibility for a multitude of Web of Things mashups for the home. Food packaging that interacts with ovens to set time and temperature, clothing that interacts with washing machines to alert us when we're about to ruin that new silk blouse and so on.

Smart Cars: Our cars are becoming an integral part of the new Web of Things as well. Ford's Sync and Toyota's entune, which are already installed in production units, connect with both the web and smartphones.

Much of the capabilities are what you would expect: navigation, roadside assistance, Pandora, and other standard fare. However, some early apps are showing the true potential once outside developers get involved in a big way.

In Japan, McDonald's is experimenting with a system that will allow for downloading menus and in-car ordering. Ford is reaching out to medical device makers to collaborate on apps that help diabetics monitor glucose levels (a serious problem behind the wheel) and monitor allergens in the air for asthmatics.

Smart Retail: I've written before about the future of retail and it's clear that the Web of Things is already transforming the shopping experience. Major retailers like Wal-Mart and Target already have apps to help consumers navigate the store. Nieman Marcus just released one that alerts salespeople when a regular customer enters the store and gives them an account history.

My Best Fit does full body scans to suggest the optimal size in various brands. Kraft is experimenting with technology that can suggest what you might want to buy for dinner based on information gleaned from a facial scan. Disney has a mirror that lets kids try on virtual outfits.

Another hotbed of innovation is payments. Check out this overview of five mobile payment options that are already in market. Cash registers will soon be the exception rather than the rule. Wherever you look, the Web of Things is turning everyday experiences into a mash-up of data and physical objects.

3.3 Architecture and standardization of IoT

What is IoT architecture?

Because of outstanding opportunities IoT promises, more organizations seek for the inclusion of its products in their business processes. However, when it comes to reality, this brilliant idea appears too complicated to be implemented — given the number of devices and conditions needed to make it work. In other words, the problem of establishing a reliable architecture of Internet of Things inevitably enters the stage.

Among all, to deal with the whole variety of factors affecting IoT architecture, it's easier and more effective to find a reliable provider of IoT solutions. This decision will significantly reduce the number of resources spent on the way. Though it's possible to comprehend the process of creating software, the practical application of its 4 stages contains too many nuances and aspects to be described in simple words. Because of that, use this guide for establishing a proper understanding of what's going on during IoT architecture — but consider referring to the specialist to make this process actually happen. This decision will facilitate getting the needed result and guarantee being a satisfied client of a software development company.

Before revealing the secrets and providing a clear structure of this initiative, it's important to understand what this concept actually means. In essence, IoT architecture is the system of numerous elements: sensors, protocols, actuators, cloud services, and layers. Given its complexity, there exist 4 stages of IoT architecture. Such a number is chosen to steadily include these various types of components into a sophisticated and unified network.

In addition, Internet of Things architecture layers are distinguished in order to track the consistency of the system. This should also be taken into consideration before the IoT architecture process start.

Basically, there are three IoT architecture layers:

1. The client side (IoT Device Layer)

2. Operators on the server side (IoT Getaway Layer)

3. A pathway for connecting clients and operators (IoT Platform Layer)

In fact, addressing the needs of all these layers is crucial on all the stages of IoT architecture. Being the basis of feasibility criterion, this consistency makes the result designed really work. In addition, the fundamental features of sustainable IoT architecture include functionality, scalability, availability, and maintainability. Without addressing these conditions, the result of IoT architecture is a failure.

Therefore, all the above-mentioned requirements are addressed in 4 stages of IoT architecture described here — on each separate stage and after completing the overall building process.

An Overview of the Main Stages in the IoT Architecture Diagram

In simple terms, the 4 Stage IoT architecture consists of

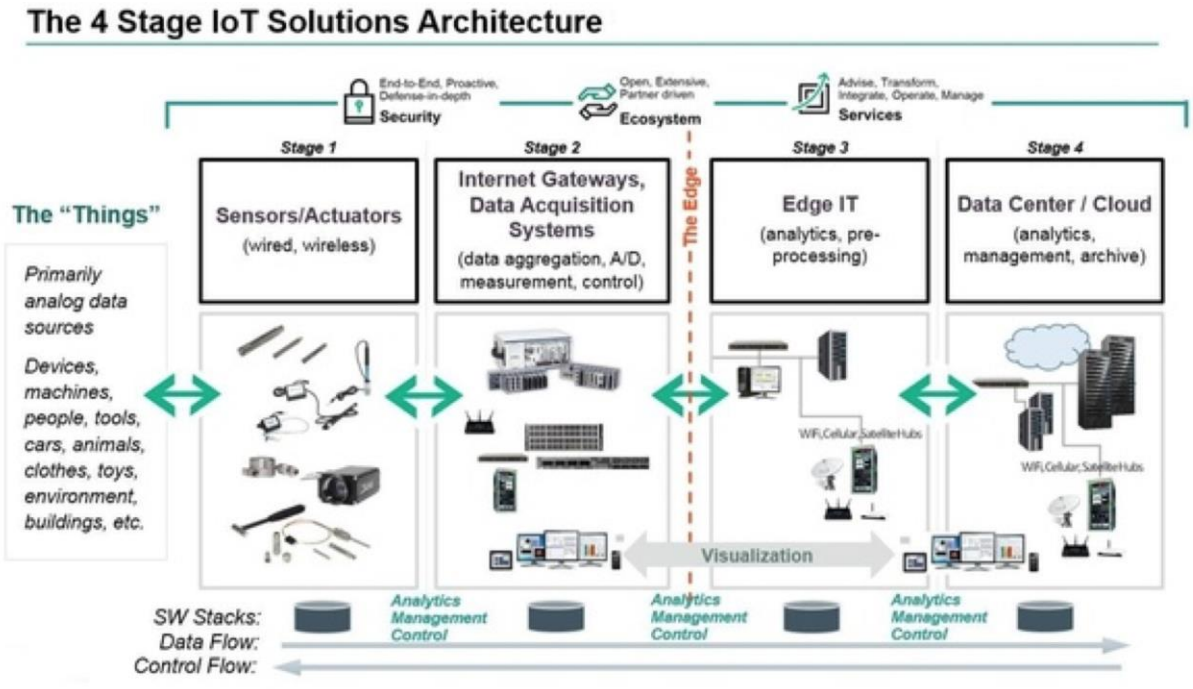
1. Sensors and actuators

2. Internet getaways and Data Acquisition Systems

3. Edge IT

4. Data center and cloud.

The detailed presentation of these stages can be found on the diagram below.



To get the proper understanding of the main actions and the importance of each stage in this process, refer to the detailed reviews presented below.

Stage 1. Networked things (wireless sensors and actuators)

The outstanding feature about sensors is their ability to convert the information obtained in the outer world into data for analysis. In other words, it's important to start with the inclusion of sensors in the 4 stages of an IoT architecture framework to get information in an appearance that can be actually processed.

For actuators, the process goes even further — these devices are able to intervene the physical reality. For example, they can switch off the light and adjust the temperature in a room.

Because of this, sensing and actuating stage covers and adjusts everything needed in the physical world to gain the necessary insights for further analysis.

Stage 2. Sensor data aggregation systems and analog-to-digital data conversion

Even though this stage of IoT architecture still means working in a close proximity with sensors and actuators, Internet getaways and data acquisition systems (DAS) appear here too. Specifically, the later connect to the sensor network and aggregate output, while Internet getaways work through Wi-Fi, wired LANs and perform further processing.

The vital importance of this stage is to process the enormous amount of information collected on the previous stage and squeeze it to the optimal size for further analysis. Besides, the necessary conversion in terms of timing and structure happens here.

In short, Stage 2 makes data both digitalized and aggregated.

Stage 3. The appearance of edge IT systems

During this moment among the stages of IoT architecture, the prepared data is transferred to the IT world. In particular, edge IT systems perform enhanced analytics and pre-processing here. For example, it refers to machine learning and visualization technologies. At the same time, some additional processing may happen here, prior to the stage of entering the data center.

Likewise, Stage 3 is closely linked to the previous phases in the building of an architecture of IoT. Because of this, the location of edge IT systems is close to the one where sensors and actuators are situated, creating a wiring closet. At the same time, the residing in remote offices is also possible.

Stage 4. Analysis, management, and storage of data

The main processes on the last stage of IoT architecture happen in data center or cloud. Precisely, it enables in-depth processing, along with a follow-up revision for feedback. Here, the skills of both IT and OT (operational technology) professionals are needed. In other words, the phase already includes the analytical skills of the highest rank, both in digital and human worlds. Therefore, the data from other sources may be included here to ensure an in-depth analysis.

After meeting all the quality standards and requirements, the information is brought back to the physical world — but in a processed and precisely analyzed appearance already.

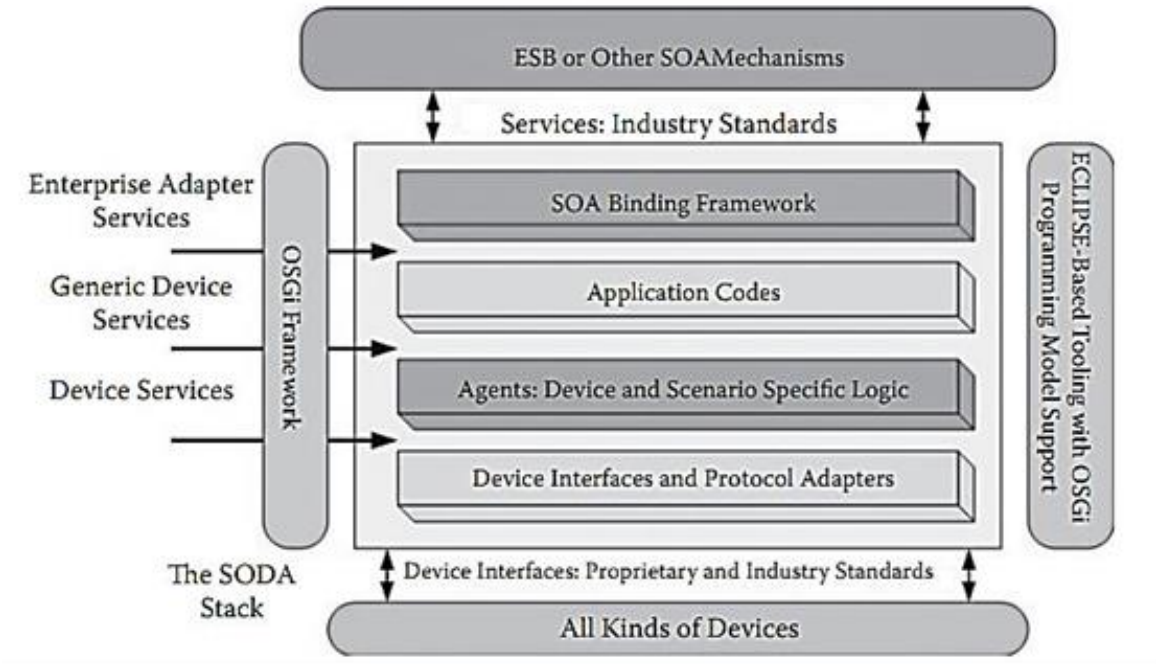
Stage 5 of IoT Architecture?

In fact, there is an option to extend the process of building a sustainable IoT architecture by introducing an extra stage in it. It refers to initiating a user's control over the structure — if only your result doesn't include full automation, of course. The main tasks here are visualization and management. After including Stage 5, the system turns into a circle where a user sends commands to sensors/actuators (Stage 1) to perform some actions.

And the process starts all over again.

3.4 Unified multitier-WoT architecture

- SOA/EAI versus SODA/MAI
- WOT/ IOT applications should inherit and enhance the existing data formats and protocols
- SOAP (simple object access protocol) is a protocol framework specification for exchanging structured information in the implementation of web services
- It relies on XML for its message format
- Usually hypertext transfer protocol (HTTP), simple mail transfer protocol (SMTP), Java messaging services (JMS)
- SOA is a set of principles and methodologies for designing and developing software in the form of interoperable services, usually over the Internet
- SOA requires metadata (unified WoT architecture also needs metadata)
- Web services description language typically describes the services, while the SOAP protocol describes the communication protocols
- Combination of existing SOA and EAI (Enterprise Application Integration) technologies is a good foundation for WOT/ IOT applications
- Service- Oriented Device Architecture (SODA) is proposed to enable device connection to an SOA
- Core of SODA standard is DDL (device description language) based on XML encodings
- DDL classifies devices into three categories: sensors, actuators, and complex devices



SODA ARCHITECTURE

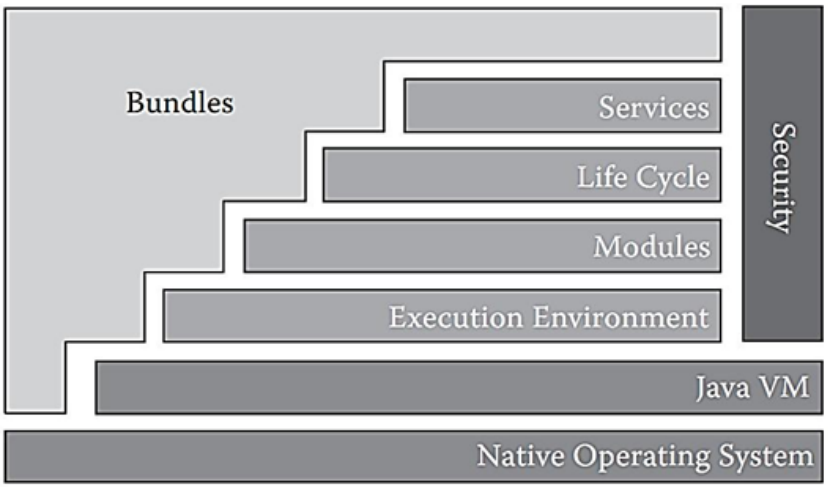
- Example of Device Description Language of SODA

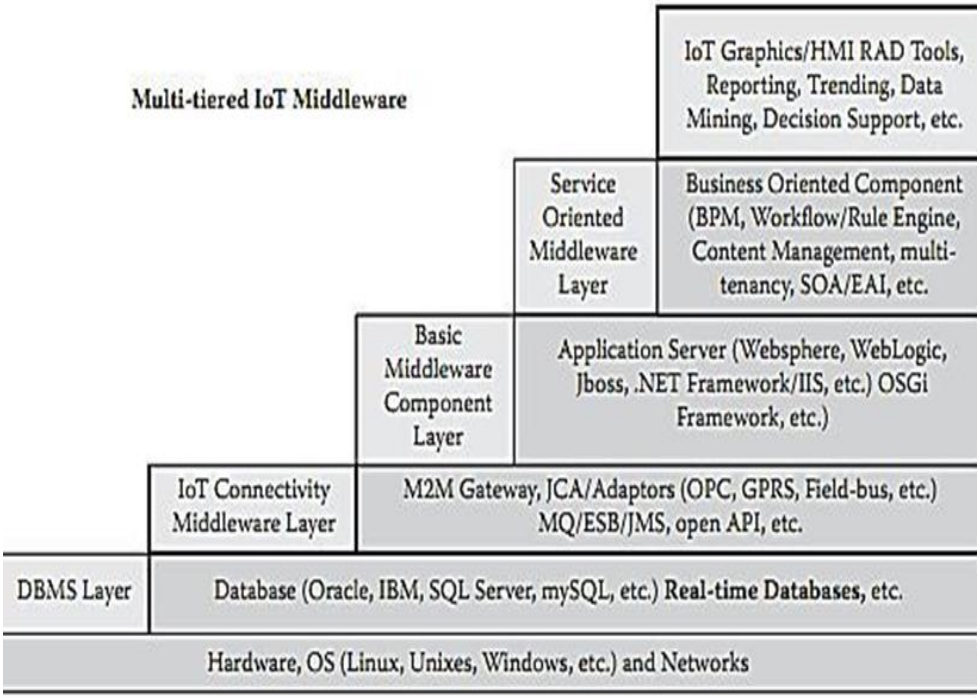
```
<Sensor>
<Description>…</Description>
<Interface>
<Signal id = "ADC1">…</Signal>
<Reading id = "Temp 1">
<Type>Physical</Type>
<Measurement>Temperature</Measurement>
<Unit>Centigrade</Unit>
<Computation>
<Type>Formula</Type>
<Expression> Temp 1 = (((ADC1/1023 * 3.3)-0.5)*
(1000/10)</Expression>
</Computation>
</Reading>
</Interface>
</Sensor>
```

OSGi: The Universal Middleware

- Open Services Gateway initiative

- Module system and service platform for the Java programming language that implements complete and dynamic component model





3.5 WoT portals and Business intelligence

- Web portal - website that functions as a point of access to information in the World Wide Web
- Portal presents information from diverse sources in a unified way
- Examples of public web portals include Yahoo, AOL, Excite, MSN
- Apart from standard search engine feature, web portals offer other services such as e-mail, news, stock prices, information, databases and entertainment.
- Categorizations of portals:
- Horizontal Portals - cover many areas
- Vertical Portals - focused on one functional area
- WOT portals are vertical portals
- When huge amount of data are collected in a IOT system, data mining can be conducted to acquire business intelligence (BI)
- Data mining deals with finding patterns in data that are by user definition, interesting and valid
- Interdisciplinary area -databases, machine learning, pattern recognition, statistics, visualization, etc.
- BI technologies provide historical, current, and predictive views of business operations
- Common functions of BI technologies are
- extract, transform, and load
- reporting, online analytical processing, analytics
- data mining, process mining, complex event processing
- business performance management, benchmarking, text mining, predictive analytics, and so on

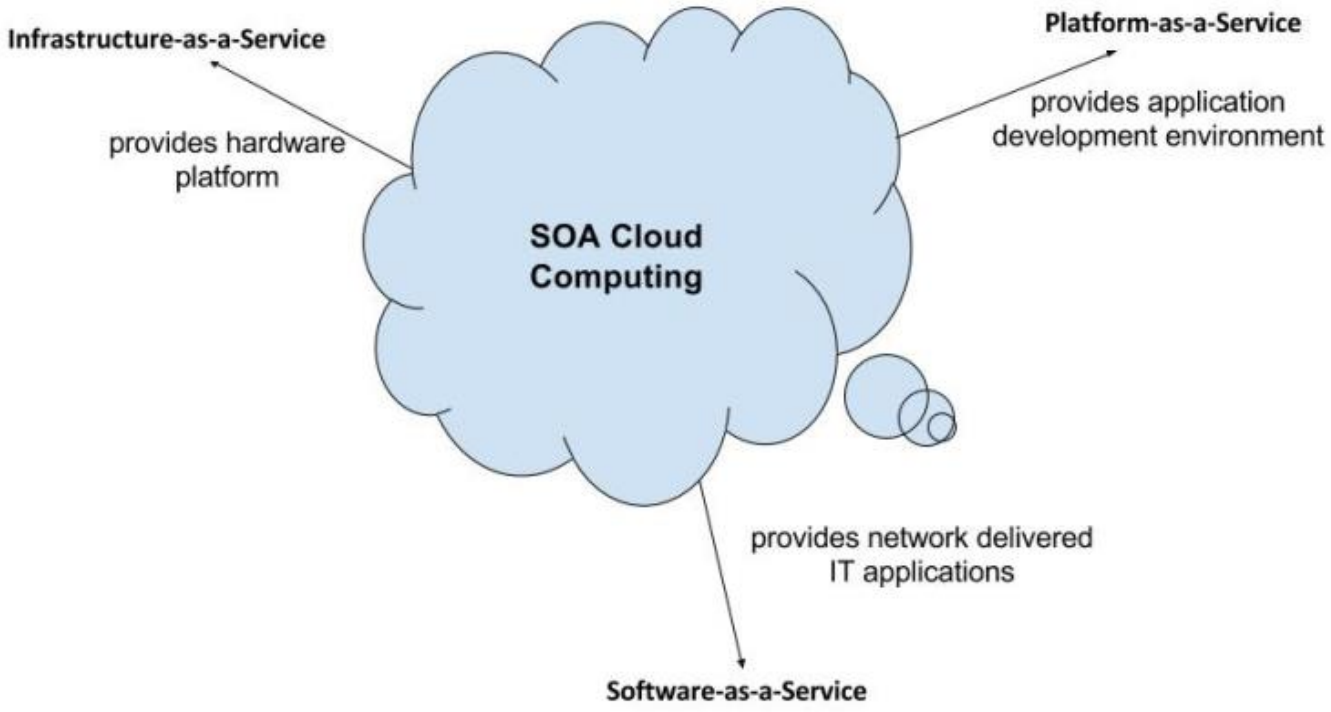
3.6 Cloud of things: Grid/SOA and cloud computing

Cloud computing is a model used for enabling convenient and usage-based network access to a configurable computing resources (eg. networks, servers etc) that can be provided and used rapidly.

- It provides a chance to business users to implement services with usage-based billing that is changed according to their requirements without need of consulting with IT department.
- It provides an abstraction layer between computing resources and its technical implementation details and sequentially enables computational resources to be used while avoiding efforts in infrastructure management.

Concepts in Cloud Computing

The below figure shows the SOA cloud computing along with the models:



Below are the models that are differentiated on the horizontal scaling basis in cloud computing:

- **Infrastructure-as-a-Service (IaaS)**: It provides a hardware platform as a service.
- **Platform-as-a-Service (PaaS)**: It provides end-users an application development environment delivered over the internet.

- **Software-as-a-Service (SaaS)**: It provides end-users a standardized, network-delivered IT applications.

The distinctions are made according to availability and the location of installation in the deployment models. Private clouds are internal company services whereas public clouds are the services that are available to the public on internet.

In the large companies where IT plays an important role, internal company cloud solutions are often built in their own data centers. Small and medium companies often use public cloud services. Cloud Computing provides a very flexible and scalable platform through processing external services and also has the ability to connect with customers, suppliers etc.

What is Service Oriented Architecture (SOA)?

The Service Oriented Architecture is an architectural design which includes collection of services in a network which communicate with each other. The complication of each service is not noticeable to other service. The service is a kind of operation which is well defined, self contained that provides separate functionality such as checking customer account details, printing bank statements etc and does not depend on the sate of other services.

History

The first report published on SOA by the analysts **Roy W.Schulte** and **Yefim V.Natis** in 1996.

Why to use SOA?

- SOA is widely used in market which responds quickly and makes effective changes according to market situations.
- The SOA keep secret the implementation details of the subsystems.
- It allows interaction of new channels with customers, partners and suppliers.
- It authorizes the companies to select software or hardware of their choice as it acts as platform independence.

Features

- SOA uses interfaces which solves the difficult integration problems in large systems.
- SOA communicates customers, providers and suppliers with messages by using the XML schema.
- It uses the message monitoring to improve the performance measurement and detects the security attacks.

- As it reuses the service, there will be lower software development and management costs.

Advantages

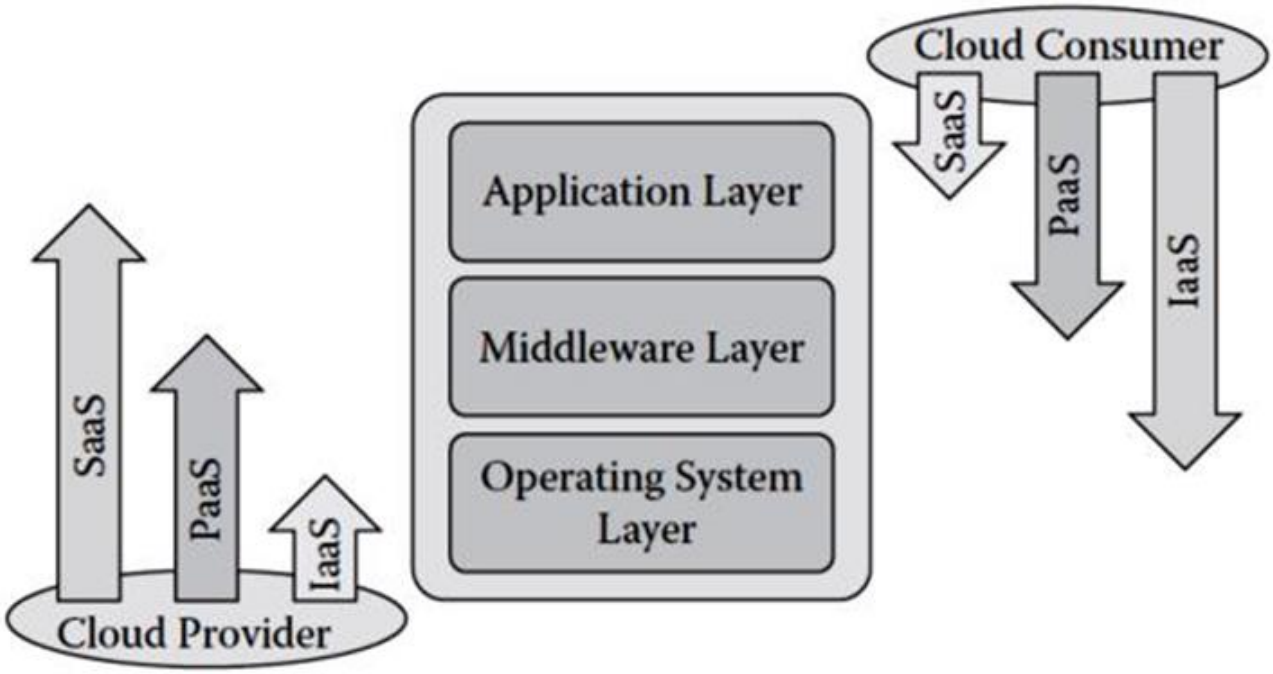
- SOA allows reuse the service of an existing system alternately building the new system.
- It allows plugging in new services or upgrading existing services to place the new business requirements.
- It can enhance the performance, functionality of a service and easily makes the system upgrade.
- SOA has capability to adjust or modify the different external environments and large applications can be managed easily.
- The companies can develop applications without replacing the existing applications.
- It provides reliable applications in which you can test and debug the independent services easily as compared to large number of code.

Disadvantages

- SOA requires high investment cost (means large investment on technology, development and human resource).
- There is greater overhead when a service interacts with another service which increases the response time and machine load while validating the input parameters.
- SOA is not suitable for GUI (graphical user interface) applications which will become more complex when the SOA requires the heavy data exchange.

3.7 Cloud middleware

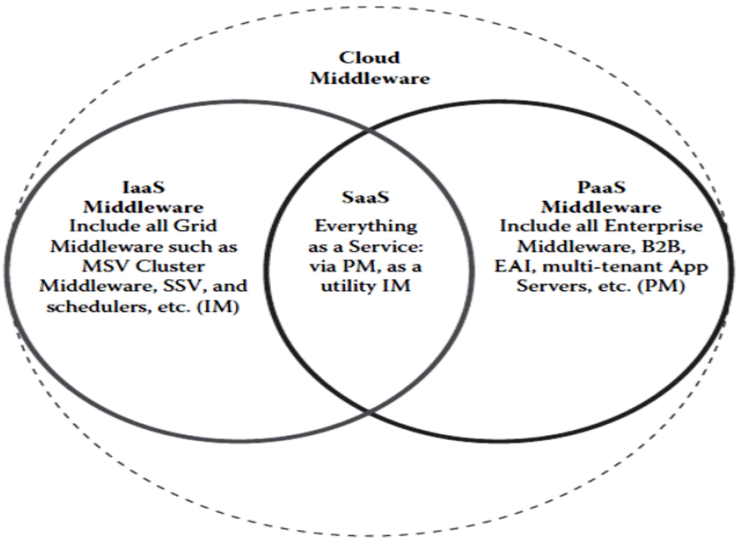
Like IOT, cloud computing system is also a multi tiered architecture built on a middleware stack.



- As an example, VAMOS [242], built by IBM, is a novel middleware architecture that runs its middleware modules at the hypervisor level
- Reduces I/O virtualization overhead by cutting down on the overall number of guest/hypervisor switches for I/O intensive workloads
- Applying VAMOS to a database application improved its performance by up to 32 percent
- Here, the middleware concept is extended to include software that does IPC not necessary over a network
- At the cluster computing or grid computing level, many types of work are done by middleware
- Parallel computing environments such as PVM and MPI are middleware by definition

- Hadoop system and the job scheduler such as Condor, LoadLeveler, and others are all middleware
- A number of grid middleware initiatives have been formed
- Some of those middleware are aggregately referred to as grid middleware
- Various grid middleware are

- Low-level middleware
- MPI, Open MPI
- PVM (parallel virtual machine)
- POE (parallel operating environment, IBM)
- Middleware for file systems and resources
- MPI-IP
- PVFS/GPFS (parallel virtual file system/general parallel file system IBM)
- Sector-Sphere
- Condor/PBS/LoadLeveler
- (IBM)
- High-level middleware
- Beowolf
- Globus Toolkit
- Gridbus
- Legion
- Unicore
- OSCAR/CAOS/Rocks
- OpenMosix/NSA/Perceus



3.8 Cloud Standards

- Cloud model is composed of the following:
- Three service models: IaaS, PaaS, and SaaS
- Four deployment models: private cloud, public cloud, community cloud, and hybrid cloud
- Five essential characteristics: on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service

RESOURCE MANAGEMENT IN IOT

4.1 Home Automation Applications

IoT based home automation can revive the way people use technology. There is a considerable range of possibilities when we speak about applications of home automation.

- Controlled electrical fixtures such as lights and air conditioners
- Simplified garden or lawn management
- HVAC
- Controlled smart home appliances
- Enhanced safety and security at home
- Water and air quality control and monitoring
- Voice based home assistant supporting natural language
- Smart locks and switches

4.2 Industry Applications

1. ABB: Smart robotics

Power and robotics firm ABB is one of the most visible to embrace the concept of predictive maintenance, using connected sensors to monitor its robots' maintenance needs — across five continents — and trigger repair before parts break. Also related to IoT is the company's collaborative robotics. Its YuMi model, which was designed to collaborate alongside humans, can accept input via Ethernet and industrial protocols like Profibus and DeviceNet.

2. Airbus: Factory of the Future

To say that assembling a commercial jetliner is an elaborate affair would be an understatement. Such craft have millions of components and tens of thousands of assembly steps, and the cost of mistakes during the process can be enormous. To tackle the complexity, Airbus has launched a digital manufacturing initiative known as Factory of the Future to streamline operations and bolster production capacity. The company has integrated sensors to tools and machines on the shop floor and given workers wearable technology — including industrial smart glasses — designed to reduce errors and bolster safety in the workplace. In one procedure, known as cabin-seat marking, the wearables enabled a 500% improvement in productivity while nearly eliminating errors.

3. Amazon: Reinventing warehousing

The online retail giant doesn't often get called an IIoT company, but, to be sure, the company is an innovator when it comes to warehousing and logistics. As MIT Technology Review has put it:

Amazon is "testing the limits of automation and human-machine collaboration." While the company's ambitions to use drones for delivery has won considerable media attention, the firm's fulfillment warehouses make use of armies of Wi-Fi-connected Kiva robots. The basic idea behind the Kiva technology, which Amazon acquired for $775 million in 2012, is that it makes more sense to have robots locate shelves of products and bring them to workers rather than have employees go to the shelves to hunt for products. In 2014, the robots helped the company cut its operating costs by 20%, according to Dave Clark, a senior vice president at Amazon.

4. Boeing: Using IoT to drive manufacturing efficiency

Aviation pioneer William Boeing quipped that it "behooves no one to dismiss any novel idea with the statement, 'It can't be done.'" The multinational aviation company founded in Boeing's name apparently still subscribes to that ethos. It is now working toward the long-term goal of making its service offerings more important than its products while being the most valuable information provider in aviation. The company has already made significant strides in transforming its business. Boeing and its Tapestry Solutions subsidiary have aggressively deployed IoT technology to drive efficiency throughout factories and supply chains. The company is also steadily increasing the volumes of connected sensors embedded into its planes.

5. Bosch: Track and trace innovator

In 2015, Bosch launched what would be the Industrial Internet Consortium's first test bed. The primary inspiration behind the so-called Track and Trace program is that workers would spend a sizable amount of their time hunting down tools. So the company added sensors to its tools to track them, starting with a cordless nutrunner. As the resolution of the tracking becomes more precise, Bosch plans to use the system to guide assembly operations.

6. Caterpillar: An IIoT pioneer

Heavy-equipment maker Caterpillar has long been an IoT projects pioneer. Recently, the company, which now often goes by "Cat," has been showing off the fruits of its investments in IoT technology. For instance, consider how it is using IoT and augmented reality (AR) applications to give machine operators an at-a-glance view of everything from fuel levels to when air filters need replacing. If an old filter expires, the company can send basic instructions for how to replace it via an AR app. The company's marine asset intelligence division is also an innovator. Last year, Forbes ran an article explaining how the company used sensor-driven analytics to save a bundle of money on boats and shipping vessels.

7. Fanuc: Helping to minimize downtime in factories

Robotics maker Fanuc is serious about reducing downtime in industrial facilities. Using sensors within its robotics in tandem with cloud-based analytics, the company can predict when failure of a component such as a robotic system or process equipment is imminent. While predictive maintenance is a familiar concept, Fanuc has embraced it more aggressively than most. Last year, GM awarded Fanuc's Zero Downtime (ZDT) system its Supplier of the Year Innovation Award.

8. Gehring: A pioneer in connected manufacturing

Gehring Technologies, a 91-year-old company that makes machines for honing metal, was early to embrace IIoT technology. Now, the company enables its customers to see live data on how Gehring's machines work before they place an order. It does so by using digital technology, beaming real-time information from a new machine to a customer to ensure that it meets the customer's requirements for precision and efficiency. Gehring uses the same cloud-based real-time tracking to reduce downtime and optimize its own manufacturing productivity through monitoring its connected manufacturing systems, visualizing and analyzing data from its machine tools in the cloud.

9. Hitachi: An integrated IIoT approach

The Japanese company stands out from other industrial companies in terms of its integration and experience across operational and information technology. While most other industrial conglomerates leverage partnerships to fill in the gaps in their IoT knowledge, Hitachi is more independent. The company has more than 16,000 employees focused on the technology in some capacity. While it offers an IoT platform known as Lumada, Hitachi also makes a plethora of products leveraging connected technology, including trains, which the company is beginning to sell as a service. Hitachi has also developed an IoT-enhanced production model that it claims has slashed production lead times by half within its Omika Works division, which manufactures infrastructure for electricity, traffic, steel manufacturing and other industries.

10. John Deere: Self-driving tractors and more

As the field of agriculture becomes more of a science and less of an art passed down the generational line, John Deere is responding by deploying Internet of Things technology — perhaps most notably with self-driving tractors. As The Washington Post wrote in 2015, Google didn't lead the self-driving vehicle revolution. John Deere did. The company also happens to be a pioneer in GPS technology. The most-advanced systems it uses in tractors are accurate to 2 centimeters. In addition, the company has deployed telematics technology for predictive maintenance applications.

4.2 Surveillance applications

In this age of IoT, the above attacks can easily cripple the system and even the entire IoT network if steps are not taken to protect and maintain the system. IoT applications and devices are often deployed in complex, uncontrolled and hostile areas and must, therefore, make provisions to tackle the below security challenges:

1. **Managing updates to the device and to the installed IoT application:** Regularly updating the IoT application with security patches must be enabled so that the system protection is up to date. The data of the system must be protected across all areas of confidentiality, availability, and integrity. This must be ensured across all surfaces, i.e., device, network, application and sensor tier. If the device is connected to the cloud, then the communication must be secured.

2. **Secure communication:** Chatter between devices must be secured via TLS or other protocols to ensure that the systems are not compromised.

3. **Monitor and detect:** Run constant scans and ensure audit logs are written and monitored for attack entries. Other preventive mechanisms must also be in place to avoid attacks.

4. **Authentication and authorization:** Password protection is a must for IoT applications, and they must be strong to avoid compromising the system by a brute force attack.

5. **Secure devices:** Firewalls, hardening, lightweight encryption, and disabling device backdoor channels are all ways to protect the IoT system from damage.

6. **Data integrity**: Data protection is a must for secure systems, and care must be taken for the same in the IoT domain as well. All sensitive data must be encrypted during transmission and storage.

7. **Secure control applications**: Applications accessing IoT applications must be fully secure in order to prevent the client IoT system from being compromised.

To conclude, securing applications is of paramount importance as they are mission critical and bringing them down can result in serious repercussions in real life. The security challenge must be managed, monitored and avoided.

4.4 Other IoT applications

. Medicine and Health care

- Health care has been a major user of IoT applications, where IoT applications are helping the users to gather statistical data and further control and automate the medical process. According to a recent survey, the IoT market share has been increased from USD $298 Billion in the year 2014 to USD $700 Billion in the year 2017. IoT technology is

being embedded in health care devices including both wearable and implantable devices that are being used to monitor and improve patients' medical conditions. With the advancement in IoT in the medical and health care domain, investors, as well as the public, will be benefited in many ways.

- Overall life-sustaining care costs will see a decline and improved health monitoring systems will be benefitting millions of people every day. A method was proposed at IoT annual meet in 2018 which was aimed at reducing childhood obesity. A robot was built with the help of AI and was assigned to collect medical data with the help of sensors placed on the chest of the children. Further, a questionnaire-based survey on the food diet, physical activities, and other environmental factors were collected from the population and transmitted over to webserver. This is one of the examples of advanced IoT functionality in the health care sector.

2. Business Analytics

IoT devices embedded in machines generate a large amount of data that is being used by BI (Business Intelligence tools) such as Power BI to generate useful insights and predict future outcomes. With the help of business analytics tools, the data generated from IoT are used to study customer behavior to increase customer satisfaction rates and provide better customer experience. In the near future, BI tools will be embedded within things such as wearable health monitoring systems, which can make an instant decision based on the current data. Data recorded from the user's behavior and everyday habits will give better opportunities for the caretakers and hospitals to tackle any sickness in advance.

3. Automotive IoT

The Internet of Vehicles (IoV) has seen rapid growth in recent days. Many researchers and organizations are spending a large amount of time and resources to reach the full potential of the Internet of Vehicles. The concept of connected cars is not too far from being reality. IoT will prove to be a game-changer and close the existing gap between the automobile and software industry. The main idea behind the concept of connected cars is to create a network of running vehicles and things such as traffic lights so that communication can be established between them. With the help of vehicle to vehicle and vehicle to an infrastructure network, the system to manage the traffic can be developed which will eventually replace the traditional traffic light system. Few of the applications of IoT in the automobile are given below:

- **Entertainment System**: Several smart apps such as the car navigation systems and voice assistance systems are already making their way to the cars. With the help of IoT, these features have been embedded in the vehicles. Automakers have partnered with Google for their apps such as google maps, google assistant and Play store services.

- **Maintenance**: With the help of IoT now vehicle owners will be aware which of the vehicle parts needs to be serviced and be safe from any breakdown. With the help of embedded sensors, IoT will be able to monitor the functionality of components such as the engine, breaks, and electrical systems.

4. Smart city and Homes

- Smart city IoT application is designed to provide improved and better-living conditions. With the growth in technology and population, IoT will play a major role in managing the city and population. Many services such as energy-saving lights, weather reporting system and streetlights will be embedded with IoT solutions for sustainable and cost-effective reasons.

- Home automation has seen rapid growth in recent times. Consumers have been provided with services like lightning control for their homes, voice-based controlling, smart air quality adjustment, AI experience and smart locks with the IoT enabled in homes.

- The biggest reason people are attracted to smart home technology is because of security features. For example, with the help of a simple IoT device, lights of the house can be monitored when on a vacation, this function will keep the intruders away. Webcams can be installed with the help of this application to monitor the home, the major advantage here is one can control the connected devices remotely using a web interface or just a simple mobile application.

Conclusion- Applications of IoT

`In this article, we have seen multiple applications of IoT in the health care industry, the automobile industry, in smart homes and the city and in the Cloud. IoT is one of the fastest-growing technology and in this age, where we are moving from the age of the products to an age of services and experience, IoT plays a key role in this technological revolution. The day is not too far when you will come home by a self-driven car and your home will auto-detect your presence to open the door for you and start playing the music depending upon your mood.

4.5 Clustering

Clustering is the concept where your IOT devices are linked to a gateway in a particular area. This is opposed to the use of "meshing" technology whereby the devices are interconnected and communicate with each other and spread out the "signal" that way. Clustering is argued as more reliable method of communications as it requires less power than Meshing and will not be let down necessarily by a group of IOT devices. It is similar to the concept of Star Distribution vs interconnection. It uses a hub in each geographical area meaning that there is a single point of failure and the hub or gateway can communicate where the issue is when your IOT devices fail.

4.6 Synchronization

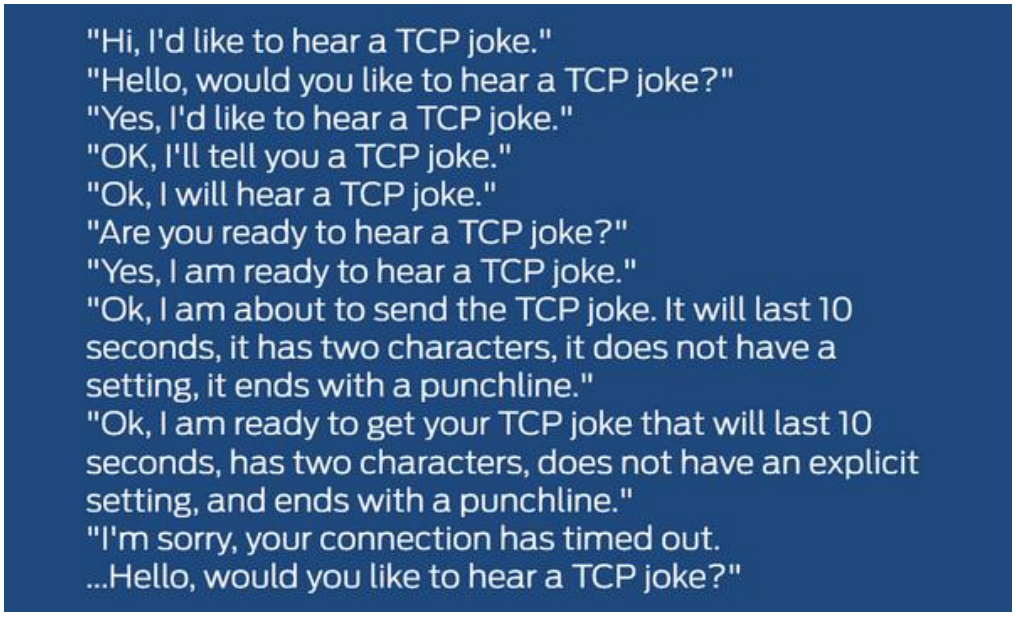
Unless you're a networking nerd, **synchronization** is probably more familiar as a term used with wristwatches or iTunes than as an IoT term, but the future of the IoT may actually depend on this topic.

Synchronization — the way an IoT device adjusts its internal clock in order to align with the clocks of other devices in a network — lies (surprisingly) at the center of many of today's IoT challenges, particularly for low-power IoT.

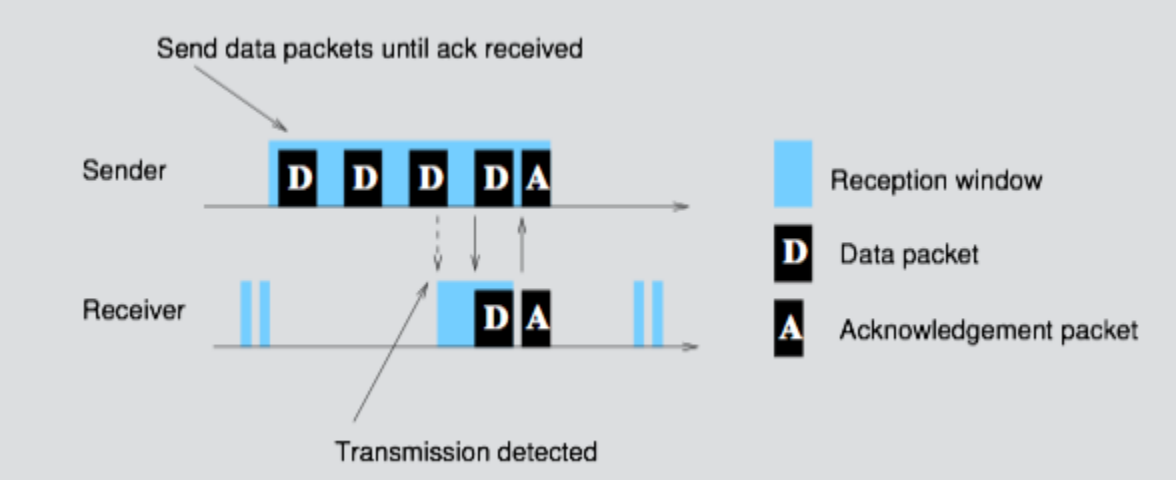
Clocks help devices pinpoint the moment when, for example, a sensor measurement is going to be shared with the network. If your device's clock is out sync with those of other devices in the network, it will miss messages, collide with other messages being sent by other devices, or waste energy trying to get back in sync.

Clocks drift out of synchronization, especially those using low cost, commodity computing parts that are often used in low power IoT. So to keep networking running efficiently, clocks need to be synchronized in order to make the data flow in a reliable way.

More than a few inventors of wireless IoT technologies didn't focus too intensely on synchronization, perhaps because they were using TCP/IP as their networking model, which while I'm thinking about it reminds me — even if slightly off topic — of this:



Most "low power" IoT protocols implemented something similarly byzantine when they designed their method for network sync. For example, here is a picture of 6lowPAN — which famously claims to be a low power means of implementing IPv6 on a wireless network — initiating the sync process:



For 6lowPAN, this process is repeated many times — let's refer to it as "strobing" — until the endpoint has synchronized its listening cycle with the host. Unfortunately, with 6lowPAN all this "strobing" takes power, can only be done one endpoint at a time, and if the data rate is low the endpoint will burn up lots of battery life as it listens and strobes.

For 6lowPAN and others in the IoT using "old school" network sync, the cost of not getting it right is high for at 5 reasons:

1) Battery Life

Like politicians promising to change Washington, most low power IoT technologies don't tell the truth about battery life. Cellular people you already know who you are.

ZigBee, Thread and others are also guilty because bad sync processes do to batteries what badly under-inflated tires do to your car's gas mileage. Multi-year battery life is what makes low power IoT … low power. Bad sync = bad battery life.

2) Connection Time

Some wireless technologies can take many seconds or even minutes to connect, due almost entirely to weak synchronization schemes. For an on-demand world where we expect immediate results when it comes to IoT, a bad sync method in a mission critical environment can render obsolete information created only seconds earlier.

Smart city or public safety applications, for example, are poorly served with slow-sync technologies. Slow-sync protocols are also a no-go for IoT control apps like implementing a kill switch on a piece of industrial equipment.

3) Dense-Packed Endpoint Environments

Environments with lots of endpoints are intimidating to IoT protocols with weak sync schemes. As in, they shouldn't even get into the ring to pretend to compete.

Imagine trying to run a query in a warehouse with 2,000 endpoints and establishing sync with each endpoint— one-by-one — in order to engage in a group broadcast or to query a group of endpoints or to send out a security patch. Industrial IoT environments are particularly sensitive to this issue.

4) Indoor Location

A growing part of battery-powered IoT has to do with locating things. Outdoors, we seem to be relying more and more on GPS, but indoors is another matter.

Being able to locate something indoors in any kind of real-time way requires fast synchronization with a gateway/access point or, more importantly, with other endpoints on a peer-to-peer basis. Slow-sync protocols are a no-go for these applications.

5) Security

IoT technologies with weak sync schemes take longer to exchange keys and are more vulnerable to unwanted discovery and spoofing. Fast-sync protocols are also better able to support two-factor authentication and can remain in a quiet/listen-before-talk mode that protects privacy and inhibits unauthorized discovery.

4.7 Software Agent

A software agent is an piece of software that functions as an agent for a user or another program, working autonomously and continuously in a particular environment. It is inhibited by other processes and agents, but is also able to learn from its experience in functioning in an environment over a long period of time.

Software agents offer various benefits to end users by automating repetitive tasks. The basic concepts related to software agents are:

- They are invoked for a task
- They reside in "wait" status on hosts
- They do not require user interaction
- They run status on hosts upon starting conditions
- They invoke other tasks including communication

There are a number of different software agents, including:

- Buyer Agents or Shopping Bots: These agents revolve around retrieving network information related to good and services.

- User or Personal Agents: These agents perform a variety of tasks such as filling out forms, acting as opponents in games, assembling customized reports and checking email, among other tasks.

- Monitoring and Surveillance Agents: These agents observe and report on equipment.

- Data-Mining Agents: These agents find trends and patterns in many different sources and allow users to sort through the data to find the information they are seeking.