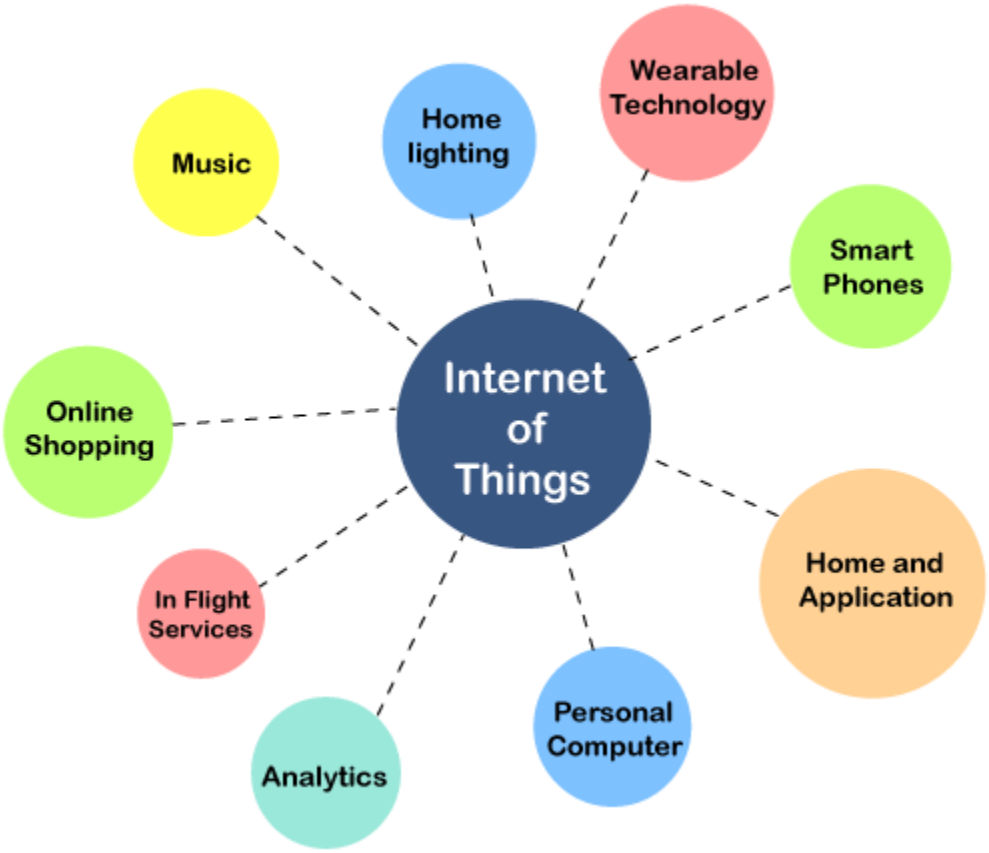
Internet of Things Applications

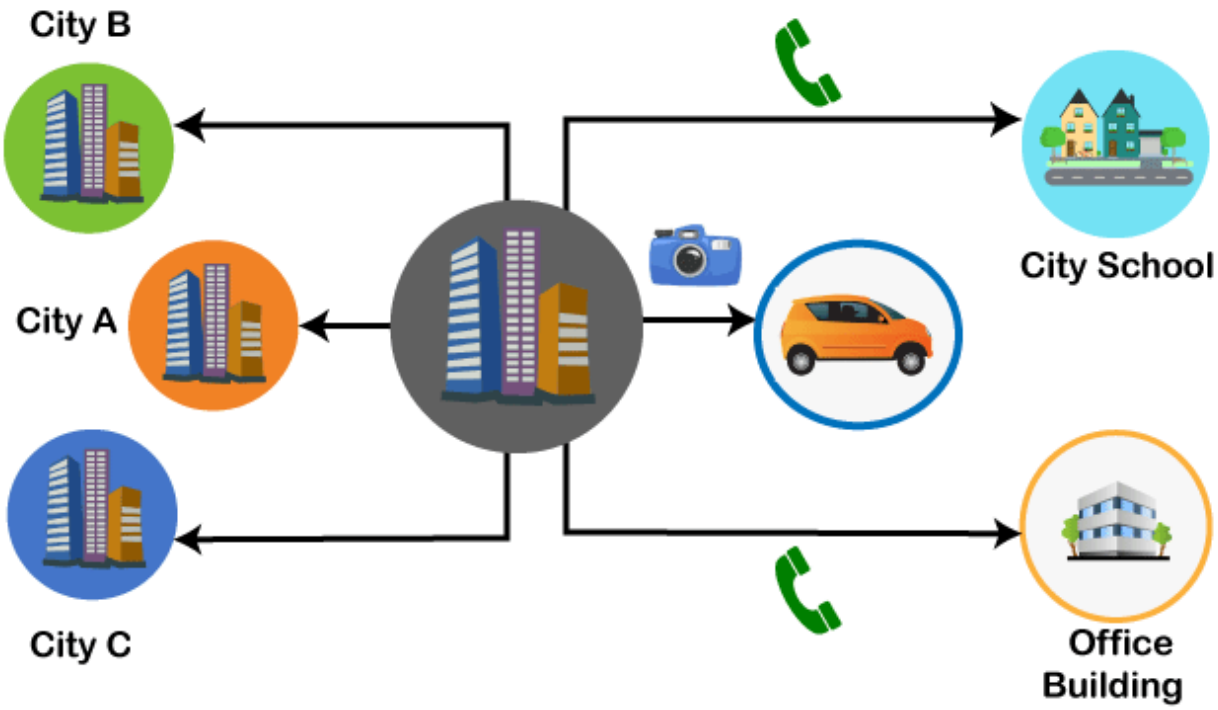
The **Internet of Things (IoT)** provides the ability to interconnect computing devices, mechanical machines, objects, animals or unique identifiers and people to transfer data across a network without the need for human-to-human or human-to-computer is a system of conversation. **IoT applications** bring a lot of value in our lives. The Internet of Things provides objects, **computing devices**, or **unique identifiers** and people's ability to transfer data across a network without the **human-to-human** or **human-to-computer interaction**.



A traffic camera is an intelligent device. The camera monitors **traffic congestion, accidents** and **weather conditions** and can access it to a common entrance.

This gateway receives data from such cameras and transmits information to the city's **traffic monitoring system**.

The intelligent system analyzes the situation, estimate their impact, and relay information to other cities connected to the same highway. It generates live instructions to drivers by smart devices and radio channels.



It creates a network of **self-dependent systems** that take advantage of real-time control.

Applications of IoT

1. Wearables

Wearable technology is the hallmark of IoT applications and one of the earliest industries to deploy IoT. We have fit bits, heart rate monitors and smartwatches these days.

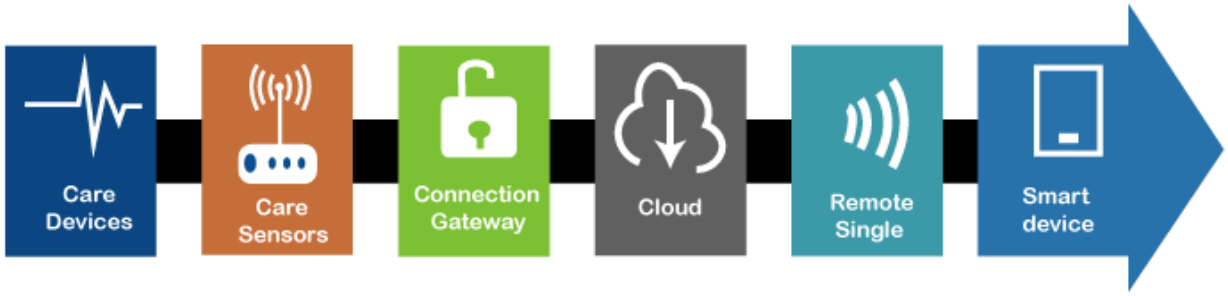
Guardian glucose monitoring device has been developed to help people with diabetes. It detects glucose levels in our body, uses a small electrode called the glucose sensor under the skin, and relates it to a radiofrequency monitoring device.

2. Smart Home Applications

The smart home is probably the first thing when we talk about the IoT application. The example we see the AI home automation is employed by **Mark Zuckerberg. Alan Pan's** home automation system, where a string of musical notes uses in-house functions.

3. Health care

IoT applications can transform reactive medical-based systems into active wellness-based systems. Resources that are used in current medical research lack important real-world information. It uses controlled environments, leftover data, and volunteers for clinical trials. The **Internet of Things** improves the device's **power, precision** and **availability**. IoT focuses on building systems rather than just tools. Here's how the IoT-enabled care device works.



4. Smart Cities

Most of you have heard about the term smart city. Smart city uses technology to provide services. The smart city includes improving transportation and social services, promoting stability and giving voice to their citizens.

The problems faced by Mumbai are very different from Delhi. Even global issues, such as clean drinking water, declining air quality, and increasing urban density, occur in varying intensity cities. Therefore, they affect every city.

Governments and engineers use the Internet of Things to analyze the complex factors of town and each city. IoT applications help in the area of water management, waste control and emergencies.

Example of a smart city - Palo Alto.

Palo Alto, San Francisco, is the first city to acquire the traffic approach. He realized that most cars roam around the same block on the streets in search of parking spots. It is the primary cause of traffic congestion in the city. Thus, the sensors were installed at all parking areas in the city. These sensors pass occupancy status to the cloud of each spot.

5. Agriculture

By the year **2050**, the world's growing population is estimated to have reached about 10 billion. To feed such a large population, agriculture needs to marry technology and get the best results. There are many possibilities in this area. One of them is Smart Greenhouse.

Farming techniques grow crops by **environmental parameters**. However, manual handling results in production losses, energy losses and labor costs, making it less effective.

The greenhouse makes it easy to monitor and enables to control the climate inside it.

6. Industrial Automation

It is one of the areas where the quality of products is an essential factor for a more significant investment return. Anyone can **re-engineer** products and their packaging to provide superior

performance in **cost** and **customer experience** with IoT applications. IoT will prove as a game-changer. In industrial automation, IoT is used in the following areas:

- **Product flow monitoring**
- **Factory digitization**
- **Inventory management**
- **Safety and security**
- **Logistics and Supply Chain Optimization**
- **Quality control**
- **Packaging customization**

FUTURE OF INTERNET TECHNOLOGY:

The uncertainty and business risk is always present in any new technology. In case of IoT, it is observed that many of the dangers are physically not present somewhat they are distorted or misstated. While it will take time to develop the IoT vision fully, the building blocks to start the process are ready to be used. The major requirements such as - hardware and software assets are either available in a less quantity or some of them are under development; it is also a fact that: the security and confidentiality concerns of IoT devices are not properly addressed over past decade. It is a whole and sole responsibility of stakeholders to collaborate and carry out the open standards to make IoT reliable, secure and interoperable. Therefore, allowing secured services to be delivered seamlessly. Over the next few years IoT is expected to make over $19 trillion. However, the problem associated with this : these ‘things' have myths surrounding them, some of which are impacting how organisations develop the apps to support them. In the above fig 5. An overview of the IoT Market in 2015 and 2020 is shown. The comparison of the percentage of total connected devices and the percent of market revenue of several verticals like Healthcare, Transport & Automotive, Retail & Banking, and Industrial & Manufacturing etc. in 2015 and 2020(expected) have been shown. By observing it can be expected that there might be an increase in percentage for some fields like Healthcare, Commercial Buildings, Smart Homes and Transport & Automotive while simultaneously a decrease might also be seen in Industrial & Manufacturing, Consumer Electronics and Retail & Banking over the years. However, there are some myths that hover around the certain future of IOT. Let us talk about each of them one by one.

IoT and Sensors:

The data produced by most sensors are not used efficiently. To help the technology evolve, 62% surveyed manufacturers believe that its functionality can be improved by advancing analytics features. More training on analytics tool was also thought to be one way by 45% people. More mobility, computing power and capacity to store data were also some factors mentioned by the manufacturers.

IoT and Mobile Data

The effectiveness of the generation of data from IoT sensors is poor. The data is usually collected by smartphones which have an integral role in IoT. The user interface for IoT applications are provided by the smartphones. However, they are not a good option. Consider the example of home automation: in case of critical home-monitoring and security applications, is it worth to rely upon a smartphone. What will happen? • When the person's smartphone goes into airplane mode during his travel? • Does the electricity shut down or, his home security gets interrupted? • What if the sensors stopped working abruptly? Today, with no IoT administrations, most of the network traffic is through the access points of Wi-Fi. What happens when that information increments by "n" times? Likewise, mobile networks and communicating gadgets have extreme disadvantages in regions, for example, consumption of power, cost, reliability and availability. So, will the smartphones and cellular communications will have a better place for running IoT Applications? The answer is absolute yes. But regarding performance, accessibility, cost, bandwidth, consumption of power and other key traits, the IoT will require altogether more varying and innovative variety of hardware and software solutions.

IoT and Volume of Data

The production of data of IoT applications is extensive. It is the fact that: The total amount of data being generated by IoT applications is not required to be stored on cloud as it consists of a lot of useless chatter generated by devices. The most significant challenge in this context is the selective storage of data on a cloud so that there will not be a storage issue in the future use of IoT devices. It also concludes that appropriate and correct data will be given to the user while rest (garbage) data produced by IoT devices will be deleted appropriately.

IoT and Datacenters

There is always constant argument that: Data in datacenters manages all the processes in IoT. It is a univocal fact that datacenter is entirely an essential factor for the IoT. We must also focus on the reliability of network which is used to run the IoT applications. High-speed Internet is equally important as its performance the functionalities like the reliable transmission of data, quick delivery of sensor data, fetching details from sensors to a cloud and vice versa.

IoT as a Future Technology IoT is an evolution in the multidisciplinary world. Microcontrollers and Microprocessors, sensors and networking devices are some of the basic building blocks of the IoT and these are in widespread use today. They have turned out to be all the more effective today, even as they get littler and more affordable to create.

IoT and current interoperability standards IoT in the long run included billions of interconnected devices over the internet. Looking at the boom of IoT, it will include numerous makers from around the globe producing numerous product categories. The term interoperability states that: All these devices must communicate, trade information and perform closely synchronized manner. They should also show the task without compromising security standards and overall performance of IoT devices.

IoT and privacy and security

Security and privacy are the main concerns while designing and developing IoT devices —and addressing these concerns must be a high priority. New technology often has scope for abuse, and it's smarter to solve the issue before it influences privacy and security, innovation or financial development. It is a responsibility of Manufacturers, standards organizations and policy-makers to address all the possible threats to the product. As a part of network layer security, manufacturers must think about the implementation of new security protocols that will be important to guarantee end-to-end transmission of delicate data.

IoT and limited vendors

Open platforms have always been a proven way for developers and merchants to build innovative hardware with constrained spending plans and assets. The behavior of IoT applications has heterogeneous nature. Hence it requires a wide variety of software and hardware. To manufactures all these IoT components, there must be a full number of vendors available in the market.

COMMUNICATIONS LAYER

The communication layer supports the connectivity of the devices. There are multiple potential protocols for communication between the devices and the cloud.

The most well-known three potential protocols are

- HTTP/HTTPS (and RESTful approaches on those)
- MQTT 3.1/3.1.1(Message Queuing Telemetry Transport)
- Constrained application protocol (CoAP)

HTTP is well known, and there are many libraries that support it. Because it is a simple textbased protocol, many small devices such as 8-bit controllers can only partially support the protocol – for example enough code to POST or GET a resource. The larger 32-bit based devices can utilize full HTTP client libraries that properly implement the whole protocol. There are several protocols optimized for IoT use. The two best known are MQTT6 and CoAP7. MQTT was invented in 1999 to solve issues in embedded systems and SCADA. It has been through some iterations and the current version (3.1.1) is undergoing standardization in the OASIS MQTT Technical Committee8. MQTT is a publish-subscribe messaging system based on a broker model. The protocol has a very small overhead (as little as 2 bytes per message), and was designed to support lossy and intermittently connected networks. MQTT was designed to flow over TCP. In addition, there is an associated specification designed for ZigBee-style networks called MQTT-SN (Sensor Networks). CoAP is a protocol from the IETF that is designed to provide a RESTful application protocol modeled on HTTP semantics, but with a much smaller footprint and a binary rather than a text- based approach. CoAP is a more traditional client-server approach rather than a brokered approach. CoAP is designed to be used over UDP. For the reference architecture we have opted to select MQTT as the preferred device communication protocol, with HTTP as an alternative option.

The reasons to select MQTT and not CoAP at this stage are • Better adoption and wider library support for MQTT;

• Simplified bridging into existing event collection and event processing systems; and

• Simpler connectivity over firewalls and NAT networks

However, both protocols have specific strengths (and weaknesses) and so there will be some situations where CoAP may be preferable and could be swapped in. In order to support MQTT we need to have an MQTT broker in the architecture as well as device libraries. We will discuss this with regard to security and scalability later. One important aspect with IoT devices is not just for the device to send data to the cloud/ server, but also the reverse. This is one of the benefits of the MQTT specification: because it is a brokered model, clients connect an outbound connection to the broker, whether or not the device is acting as a publisher or subscriber. This usually avoids firewall problems because this approach works even behind firewalls or via NAT. In the case where the main communication is based on HTTP, the traditional approach for sending data to the device would be to use HTTP Polling. This is very inefficient and costly, both in terms of network traffic as well as power requirements. The modern replacement for this is the WebSocket protocol9 that allows an HTTP connection to be upgraded into a full two-way connection. This then acts as a socket channel (similar to a pure TCP channel) between the server and client. Once that has been established,

it is up to the system to choose an ongoing protocol to tunnel over the connection. For the reference architecture we once again recommend using MQTT as a protocol with Web Sockets. In some cases, MQTT over Web Sockets will be the only protocol. This is because it is even more firewall-friendly than the base MQTT specification as well as supporting pure browser/JavaScript clients using the same protocol. Note that while there is some support for Web Sockets on small controllers, such as Arduino, the combination of network code, HTTP and Web Sockets would utilize most of the available code space on a typical Arduino 8-bit device. Therefore, we only recommend the use of Web Sockets on the larger 32-bit devices.

AGGREGATION/BUS LAYER

An important layer of the architecture is the layer that aggregates and brokers communications. This is an important layer for three reasons:

1. The ability to support an HTTP server and/or an MQTT broker to talk to the devices

2. The ability to aggregate and combine communications from different devices and to route communications to a specific device (possibly via a gateway)

3. The ability to bridge and transform between different protocols, e.g. to offer HTTP based APIs that are mediated into an MQTT message going to the device. The aggregation/bus layer provides these capabilities as well as adapting into legacy protocols. The bus layer may also provide some simple correlation and mapping from different correlation models (e.g. mapping a device ID into an owner‘s ID or vice-versa). Finally, the aggregation/bus layer needs to perform two key security roles. It must be able to act as an OAuth Resource Server (validating Bearer

Tokens and associated resource access scopes). It must also be able to act as a policy enforcement point (PEP) for policy-based access. In this model, the bus makes requests to the identity and access management layer to validate access requests. The identity and access management layer acts as a policy decision point (PDP) in this process. The bus layer then implements the results of these calls to the PDP to either allow or disallow resource access.

EVENT PROCESSING AND ANALYTICS LAYER

This layer takes the events from the bus and provides the ability to process and act upon these events. A core capability here is the requirement to store the data into a database. This may happen in three forms. The traditional model here would be to write a server-side application, e.g. this could be a JAX-RS application backed by a database. However, there are many approaches where we can support more agile approaches. The first of these is to use a big data analytics platform. This is a cloud scalable platform that supports technologies such as Apache Hadoop to provide highly scalable map reduce analytics on the data coming from the devices. The second approach is to support complex event processing to initiate near realtime activities and actions based on data from the devices and from the rest of the system.

Our recommended approach in this space is to use the following approaches:

• Highly scalable, column-based data storage for storing events

• Map-reduce for long-running batch-oriented processing of data

• Complex event processing for fast in-memory processing and near real-time reaction and autonomic actions based on the data and activity of devices and other systems

• In addition, this layer may support traditional application processing platforms, such as Java Beans, JAX-RS logic, message-driven beans, or alternatives, such as node.js, PHP, Ruby or Python.

CLIENT/EXTERNAL COMMUNICATIONS LAYER

The reference architecture needs to provide a way for these devices to communicate outside of the device-oriented system. This includes three main approaches. Firstly, we need the ability to create web-based front-ends and portals that interact with devices and with the event-processing layer. Secondly, we need the ability to create dashboards that offer views into analytics and event processing. Finally, we need to be able to interact with systems outside this network using machine-to-machine communications (APIs). These APIs need to be managed and controlled and this happens in an API management system. The recommended approach to building the web front end is to utilize a modular front-end architecture, such as a portal, which allows simple fast composition of useful UIs. Of course, the architecture also supports existing Web server-side technology, such as Java Servlets/ JSP, PHP, Python, Ruby, etc. Our recommended approach is based on the Java framework and the most popular Java-based web server, Apache Tomcat. The dashboard is a re-usable system focused on creating graphs and other visualizations of data coming from the devices and the event processing layer.

The API management layer provides three main functions:

• The first is that it provides a developer-focused portal (as opposed to the user focused portal previously mentioned), where developers can find, explore, and subscribe to APIs from the system. There is also support for publishers to create, version, and manage the available and published APIs:

• The second is a gateway that manages access to the APIs, performing access control checks (for external requests) as well as throttling usage based on policies. It also performs routing and load- balancing;

• The final aspect is that the gateway publishes data into the analytics layer where it is stored as well as processed to provide insights into how the APIs are used.

Security and Privacy

Every connected device creates opportunities for attackers. These vulnerabilities are broad, even for a single small device. The risks posed include data transfer, device access, malfunctioning devices, and always-on/always-connected devices.

The main challenges in security remain the security limitations associated with producing lowcost devices, and the growing number of devices which creates more opportunities for attacks.

Security Spectrum

The definition of a secured device spans from the most simple measures to sophisticated designs. Security should be thought of as a spectrum of vulnerability which changes over time as threats evolve.

Security must be assessed based on user needs and implementation. Users must recognize the impact of security measures because poorly designed security creates more problems than it solves.

Example − A German report revealed hackers compromised the security system of a steel mill. They disrupted the control systems, which prevented a blast furnace from being shut down properly, resulting in massive damage. Therefore, users must understand the impact of an attack before deciding on appropriate protection.

Challenges

Beyond costs and the ubiquity of devices, other security issues plague IoT −

- **Unpredictable Behavior** − The sheer volume of deployed devices and their long list of enabling technologies means their behavior in the field can be unpredictable. A specific system may be well designed and within administration control, but there are no guarantees about how it will interact with others.
- **Device Similarity** − IoT devices are fairly uniform. They utilize the same connection technology and components. If one system or device suffers from a vulnerability, many more have the same issue

- **Problematic Deployment** − One of the main goals of IoT remains to place advanced networks and analytics where they previously could not go. Unfortunately, this creates the problem of physically securing the devices in these strange or easily accessed places.
- **Long Device Life and Expired Support** − One of the benefits of IoT devices is longevity, however, that long life also means they may outlive their device support. Compare this to traditional systems which typically have support and upgrades long after many have stopped using them. Orphaned devices and abandon ware lack the same security hardening of other systems due to the evolution of technology over time.
- **No Upgrade Support** − Many IoT devices, like many mobile and small devices, are not designed to allow upgrades or any modifications. Others offer inconvenient upgrades, which many owners ignore, or fail to notice.
- **Poor or No Transparency** − Many IoT devices fail to provide transparency with regard to their functionality. Users cannot observe or access their processes, and are left to assume how devices behave. They have no control over unwanted functions or data collection; furthermore, when a manufacturer updates the device, it may bring more unwanted functions.
- **No Alerts** − Another goal of IoT remains to provide its incredible functionality without being obtrusive. This introduces the problem of user awareness. Users do not monitor the devices or know when something goes wrong. Security breaches can persist over long periods without detection.