Unit 1

Introduction to Internet of Things

1.1. INTRODUCTION TO INTERNET OF THINGS (IOT)

The IOT is a technology concept and/or an architecture which is an accumulation of already available technologies. Here by things we mean all available digital devices. IOT has evolved from the convergence of wireless technologies, micro electromechanical systems and the Internet.

The IOT is defined as a paradigm in which objects equipped with sensors, actuators, and processors communicate with each other to serve a meaningful purpose. IoT could also be looked at as simply an interaction between the *physical* and *digital* world. Once stand-alone devices and applications now have the potential to be connected to a network through sensors, actuators, processors, and transceivers. Starting from the bottom level, the data flow gets generated from any "thing" through sensors that are being sent out to the cloud through communication gateway for analysis, which turns out to be useful information.

The 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. IoT architecture adopts a model which can also be called as an event-driven model. The visionaries have also realized that this IoT ecosystem has business applications in areas of Home Automation, Factory/assembly line automation, Retail, Medical/Preventive healthcare, Automotive and more.

The definition of the IoT has evolved due to the convergence of multiple technologies, real-time analytics, machine learning, commodity sensors, and embedded systems. Traditional fields of embedded systems, wireless sensor networks, control systems, automation (including home and building automation), and others all contribute to enabling the Internet of Things.

The IoT sounds like a straightforward enough description of what it is simply the interconnectivity of smart devices, of which is it predicted there will be almost 50 billion by 2020. The most common example given for this is a smart fridge that can inform its owner when they are running out of milk while they're out shopping, but it encompasses far more than just helpful household appliances, enabling more complex systems like smart cities and virtual power plants.

The term was coined in 1999 by Kevin Ashton, who described IoT as: "If we had computers that knew everything there was to know about things—using data they gathered without any help from us—we would be able to track and count everything, and greatly reduce waste, loss, and cost. We would know when things needed replacing, repairing or recalling, and whether they were fresh or past their best."

As per author this term is slightly misnomer as the analog part of this technology can be called **things** which are controlled or rather actuated by Internet acting as media through digital electronic devices that in turn are actually sensors and controllers.

The Internet of Things is already appearing more commonly in our homes in the form of smart heating systems, smart meters for electricity and entertainment devices like the Amazon Echo and Google Home. Fitness devices like Fitbits also bring the IoT to our wrists to allow us to track our exercise routines, while smart cars are becoming more common, even if self-driving cars remain a long way off.

The Internet currently connects people to people (P2P) and is now being called Internet Phase 1. The next Phase of the Internet is just beginning, and will connect *machine to people everyday devices* (M2P), and everyday devices to each other *machine to machine (*M2M*) applications.*

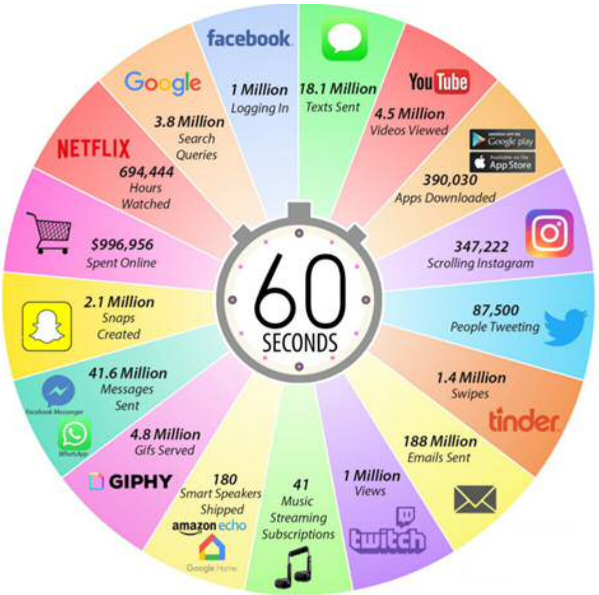


Fig. 1.1. Volume of Internet Transactions all Over the World in One Minute.

Gartner forecasts that 20.4 billion connected things will be in use worldwide by 2020,A new forecast from IDC estimates that there will be 41.6 billion connected IoT devices, or "things," generating 79.4 zettabytes (ZB) of data in 2025. Most of the wireless routers and access points can support about 250 devices connected at once. This WiFi connection number includes computers, cameras, tablets, smart phones, appliances, and a wide variety of other devices that are now internet-enabled. Gartner Founded in 1979, is a leading research and advisory company to provide flagship technology research to senior leaders. International Data Corporation (IDC, founded in 1964) is a Chinese provider of market intelligence, advisory services, and events for the information technology, telecommunications, and consumer technology markets.

1.2. BASIC COMPONENTS

In broad terms there are four main components of an IoT system:

1. The **Thing** itself (that is, the device or gadget that ought to be a sensor or actuator)
2. The local network (this can include a gateway, which translates proprietary communication protocols to Internet Protocol)
3. The Internet and cloud
4. Back-end services (enterprise data systems, or PCs and mobile devices)

1.2.1 Building Blocks of IoT

A **thing**, in the Internet-of-Thing, can be a *person* or *animal* or *any object* that can generate data through a built-in sensor, or any other natural or man-made object that can be assigned an IP address and provided with the ability to transfer data over a network.

Any Automated system can be a machine or a process and can also be called a **process control system**. Function of an process control system is constantly watched by input devices (sensors) that give signals to a controller. In response to this, controller sends a signal to external output devices (actuator or operative instruments) that actually control how system functions in an assigned manner (for simplification it is recommended that you draw a block diagram of operations' flow).

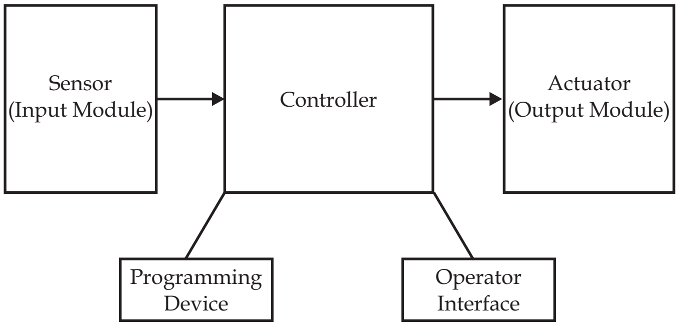


Fig. 1.2. The Concept of Automatic Control Through Iot

Each component of any process control system plays an important role, regardless of its size. For example, without a sensor, controller wouldn't know what exactly goes on in the process. In heavy automated system, programmable logic controller (PLC) is usually the central part of an process control system and in case of built in devices it can be micro-controller. With execution of a program stored in program memory, PLC or microcontroller continuously monitors status of the system through signals from input sensing devices. Based on the logic implemented in the program, PLC determines which actions need to be executed with actuator i.e. output instrument. To run more complex processes it is possible to connect more PLC / controllers to a central computer. A real system could look like Fig. 1.2. We shall discuss the function of all of them.

1. **Sensors**

Sensors are devices used to provide signals representative of conditions in a machine or process. In many situations, these devices provide analog signals that represent a range of values.

Discrete-type sensors provide information that represents the presence or absence of an object. The sense in the following illustration, for example detects the presence of bottles on a conveyor.

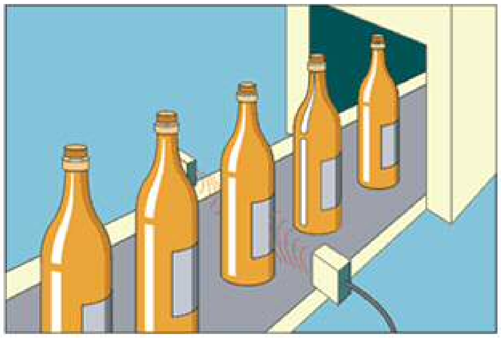


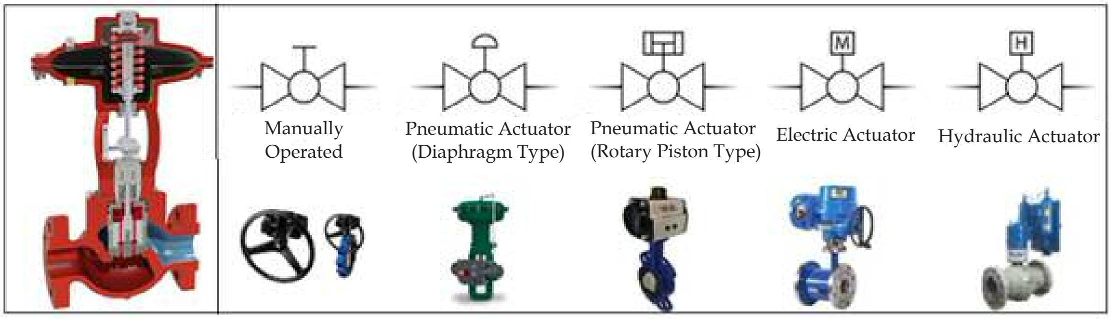
Fig. 1.3. Sensor automatically senses and counts automatically.

A better term for a sensor is a **transducer**. A transducer is any physical device that converts one form of energy into another. So, in the case of a sensor (a thing in itself), the transducer converts some physical phenomenon (mostly in analog form) into an electrical impulse that can then be easily converted in digital form for sending signal to to controller for controlling intelligent devices or physical phenomena. A microphone is a sensor that takes vibrational energy (sound waves), and converts it to electrical energy in a useful way for other components in the system to correlate back to the original sound.

A *sensor* is a *transducer*, whose purpose is to sniff a wide variety of information ranging from Location, Weather/Environment conditions, Grid parameters, Movement on assembly lines, Jet engine maintenance data to Health essentials of a patient and generate output as an electrical or optical signal. The sensors in the IOT are called as a *node* that will collect information and sent to the outside world, through communication protocols – Bluetooth, BLE, ZigBee, Z-wave, Wi-Fi or through wired communication. Weshall study about sensors in depth in unit 2 of this book. These nodes will be forwarding the data to a device called Gateway.

2. Actuator

An actuator is a component of an IoT system that is responsible for moving and controlling a mechanism or system, for example by opening a valve. In simple terms, it is a "mover". An actuator requires a control signal and a source of energy. The actuator is something that converts energy into motion. It also can be used to apply a force. An actuator typically is a mechanical device that takes energy — usually energy that is created by electric current, hydraulic fluid pressure, or pneumatic pressure— and converts it into some kind of motion. Actuators can modify the physical state of a device and can move (translate, rotate, etc.) simple devices or activate/deactivate functionalities of more complex ones.



Manually Operated
Pneumatic Actuator (Diaphragm Type)
Pneumatic Actuator (Rotary Piston Type)
M
Electric Actuator
H
Hydraulic Actuator

Fig. 1.4. Different Types of Solenoid operated Fluid Valve Actuator

3. Gateway : IoT Connectivity and Edge Gateways

An IoT **gateway** is an intermediate device between sensors and devices and the applications that create value from their data and access. The gateway allows you to efficiently collect and securely transport data from devices, remote users, and applications to serve a particular need.

An IoT gateway aggregates sensor data, translates between sensor protocols, and processes sensor data before sending it onward. IoT gateways perform several critical functions such as; device connectivity, protocol translation, data filtering and processing, security, and management.

IoT gateways may also operate as platforms for the application code that processes data and becomes an intelligent part of a device-enabled system. IoT gateways sit at the intersection of edge systems – sensors/devices/controllers and the Cloud.

In effect a gateway is a *physical device* or s*oftware program* that serves as the connection point between *the cloud and controllers, sensors and intelligent devices vide Fig. 1.5.* We shall read about gateway in detail in para 1.10. and cloud technology in para 1.11. The main function of the IoT.

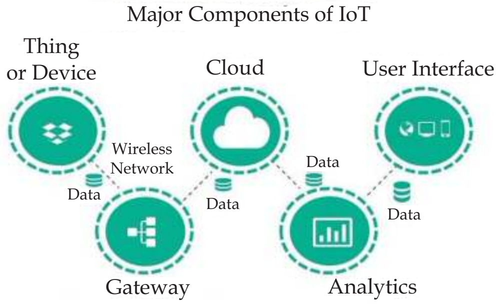


Fig. 1.5. (a) Schematic diagram of IoT gateway (b) Picture of Gateway Device

Edge computing is the deployment of data-handling activities in close proximity to sources of data capture from things, such as mobiles laptops, tablets or smartphones.

Gateways are further discussed in para 1.10 also.

4. Functioning of Sensors and Actuators

Another type of transducer that you will encounter in many IoT systems is an actuator. In simple terms, an actuator operates in the reverse direction of a sensor. It takes an electrical input and turns it into physical action. For instance, an electric motor, a hydraulic system, and a pneumatic system are all different types of actuators.

In a typical IoT system, a sensor may collect information and route to a control center where a decision is made and a corresponding command is sent back to an actuator in response to that sensed input. Later, we will discuss where the control center resides in the greater IoT system.

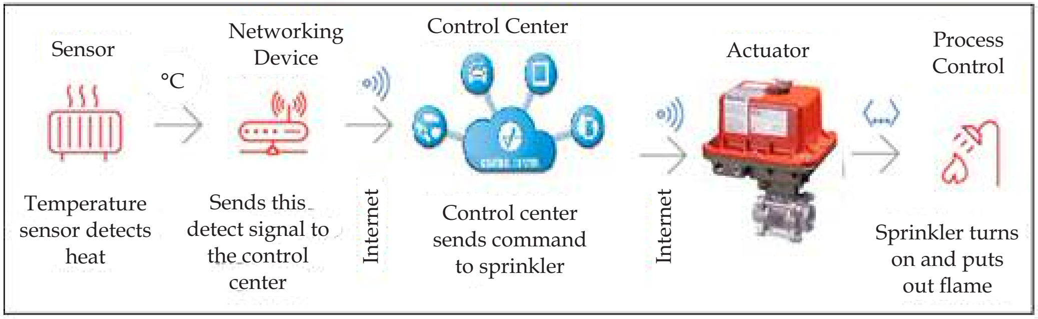

Sensor
°C
Networking Device
Control Center
Actuator
Process Control
Temperature sensor detects heat
Sends this detect signal to the control center
Internet
Control center sends command to sprinkler
Internet
Sprinkler turns on and puts out flame

Fig. 1.6 : Working of a typical IoT system for Process control (Automation)

There are many different types of sensors in an IoT system. Flow sensors, temperature sensors, voltage sensors, humidity sensors, and the list goes on. In addition, there are multiple ways to measure the same thing. For instance, airflow might be measured by using a small propeller like the one you would see on a weather station. Alternatively, as in a vehicle measuring the air through the engine, airflow is measured by heating a small element and measuring the rate at which the element is cooling.

5. Microcontroller

Most IoT devices use some kind of microcontroller. Microcontrollers can be thought of as tiny computers that are added to any physical object or space to give it a ‹brain. ‘ They contain one or more computer processors, along with memory and programmable input/output peripherals — all in a single integrated circuit.A microcontroller is a compact integrated circuit designed to govern a specific operation in an embedded system. A typical microcontroller includes a processor, memory and input/output (I/O) peripherals on a single chip.

A **microcontroller** (MCU for **microcontroller** unit) is a small computer on a single metal-oxide-semiconductor (MOS) integrated circuit chip. In modern terminology, it is similar to, but less sophisticated than, a *system on a chip* (SoC); a SoC may include a **microcontroller** as one of its components. It handles all the number crunching and local data manipulation and decision-making. The input ports collect data from sensors. While the outputs support any necessary actuation or local control in the IoT device. Usually, microcontrollers control various devices or subsystems within embedded applications.

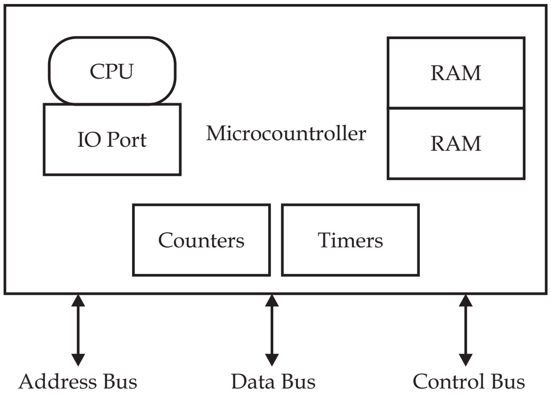


Fig. 1.7 : A microcontroller used in IoT Applications

For example, a car might have many microcontrollers that control various individual systems within, such as the anti-lock braking system, traction control, fuel injection or suspension control. All the microcontrollers communicate with each other to inform the correct actions. Some might communicate with a more complex central computer within the car, and others might only communicate with other microcontrollers. They send and receive data using their I/O peripherals and process that data to perform their designated tasks.

Core Elements of a Microcontroller

The core elements of a microcontroller are:

(*a*) **The processor (CPU)**. A processor can be thought of as the brain of the device. It processes and responds to various instructions that direct the microcontroller’s function. This involves performing basic arithmetic, logic and I/O operations. It also performs data transfer operations, which communicate commands to other components in the larger embedded system.

(*b*) **Memory.** A microcontroller's memory is used to store the data that the processor receives and uses to respond to instructions that it's been programmed to carry out. A microcontroller has two main memory types:

(*c*) **Program memory.** which stores long-term information about the instructions that the CPU carries out. Program memory is non-volatile memory, meaning it holds information over time without needing a power source.

(*d*) **Data memory.** which is required for temporary data storage while the instructions are being executed. Data memory is volatile, meaning the data it holds is temporary and is only maintained if the device is connected to a power source.

(*e*) **I/O peripherals**. The input and output devices are the interface for the processor to the outside world. The input ports receive information and send it to the processor in the form of binary data. The processor receives that data and sends the necessary instructions to output devices that execute tasks external to the microcontroller.

While the processor, memory and I/O peripherals are the defining elements of the microprocessor, there are other elements that are frequently included. The term *I/O peripherals* itself simply refers to supporting components that interface with the memory and processor. There are many supporting components that can be classified as peripherals. Having some manifestation of an I/O peripheral is elemental to a microprocessor, because they are the mechanism through which the processor is applied.

Other supporting elements of a microcontroller include:

(*a*) **Analog to Digital Converter (ADC)**. An ADC is a circuit that converts analog signals to digital signals. It allows the processor at the center of the microcontroller to interface with external analog devices, such as sensors.

(*b*) **Digital to Analog Converter (DAC)**. A DAC performs the inverse function of an ADC and allows the processor at the center of the microcontroller to communicate its outgoing signals to external analog components.

(*c*) **System bus**. The system bus is the connective wire that links all components of the microcontroller together.

(*d*) **Serial port**. The serial port is one example of an I/O port that allows the microcontroller to connect to external components. It has a similar function to a USB or a parallel port but differs in the way it exchanges bits.

A control loop requires a sensor to measure the process variable, control logic to process data, as well as calculate an instruction, and a controlled device to execute the instruction. A controller is defined as a device that has inputs (sensors), outputs (controllable devices) and the ability to execute control logic (software). We shall read in detail about types of Microcontrollers in unit 4 and 5 of this book.

(*e*) **IoT software.** IoT software addresses its key areas of networking and action through platforms, embedded systems, partner systems, and middleware. These individual and master applications are responsible for data collection, device integration, real-time analytics, and application and process extension within the IoT network. They exploit integration with critical business systems (e.g., ordering systems, robotics, scheduling, and more) in the execution of related tasks. The software aspect is further discussed in Unit 6 of this book.

(*f*) **Data Collection.** This software manages sensing, measurements, light data filtering, light data security, and aggregation of data. It uses certain protocols to aid sensors in connecting with real-time, machine-to-machine networks. Then it collects data from multiple devices and distributes it in accordance with settings. It also works in reverse by distributing data over devices. The system eventually transmits all collected data to a central server.

(*g*) **Device Integration.** Software supporting integration binds (dependent relationships) all system devices to create the body of the IoT system. It ensures the necessary cooperation and stable networking between devices. These applications are the defining software technology of the IoT network because without them, it is not an IoT system. They manage the various applications, protocols, and limitations of each device to allow communication.

(*h*) **Real-Time Analytics.** These applications take data or input from various devices and convert it into viable actions or clear patterns for human analysis. They analyze information based on various settings and designs in order to perform automation-related tasks or provide the data required by industry.

Application and Process Extension : These applications extend the reach of existing systems and software to allow a wider, more effective system. They integrate predefined devices for specific purposes such as allowing certain mobile devices or engineering instruments access. It supports improved productivity and more accurate data collection.

1.2.2. IoT Application Frame Work

An IoT platform is a middleware between the IoT gateways and hardware-related layers of IoT devices on one hand and the application and business layers on the other. IoT platforms are not just about technology but also about the user benefits and use cases.

Understanding IoT Platform

Often referred to as middleware solutions, internet of Things (IoT) platforms are a combination of function from multiple vendors, which include the following:

1. Sensors or controllers
2. Software to analyze and interpret the data
3. A gateway device to amass and transfer data back and forth to the data network
4. A communication network to send data
5. The end application services creating much of the value

Collectively, the above solutions are referred to as the value chain of IoT. One of the main objectives of IoT is connecting devices with each other. An IoT platform ensures this by providing the basic layer of services with interoperability between nodes, cloud services as well as basic IP networking, security, application layer and device management. This also ensures high level IoT application development. Put simply, an IoT platform provides all the key ingredients to build secure and efficient IoT applications.

The 5 Layers of the IoT Technology Stack

The IoT technology stack is nothing else than a range of technologies, standards and applications, which lead from the simple connection of objects to the Internet to the most easy and most complex applications that use these connected things, the data they gather and communicate and the different steps needed to power these applications.

The greatest challenge of managing an IoT solution is that there are five layers in the IoT technology stack, and decisions need to be made at each layer.

For reference, the five layers are shown in figure 1.8 :

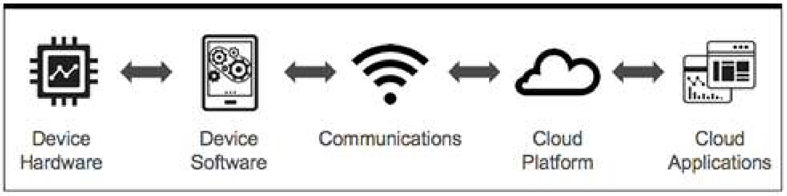


Fig. 1.8. The IoT technology Stack

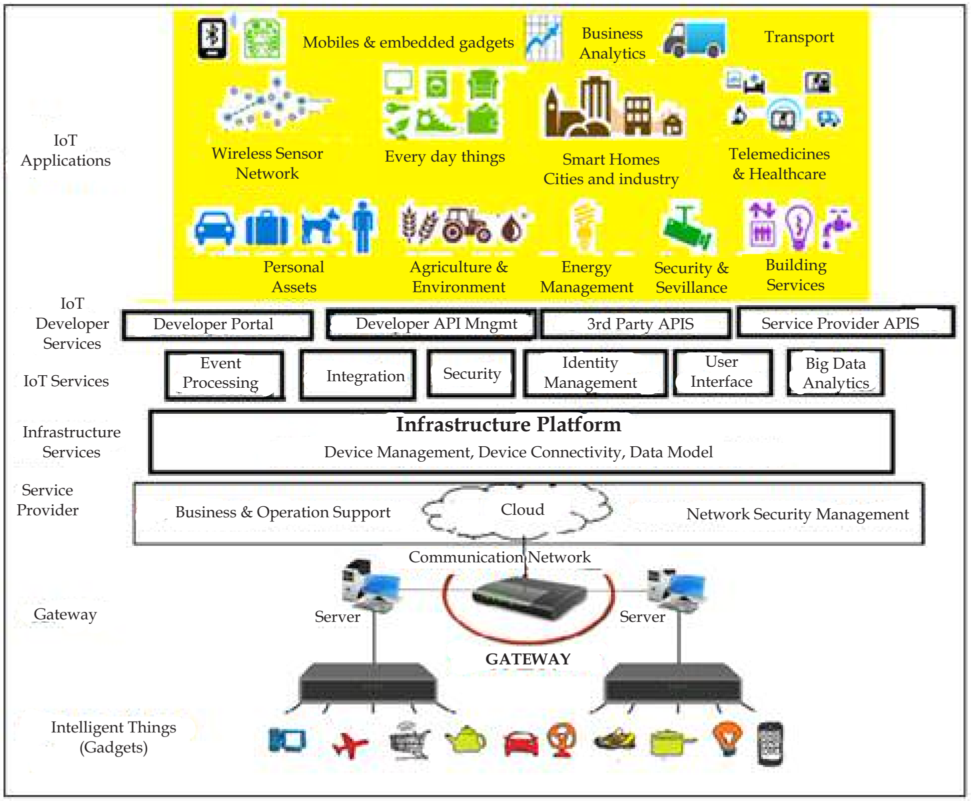


Fig. 1.9. IoT Application Framework

1.2.3. The Internet Gateway

The data from the sensors starts in analog form. That data needs to be aggregated and converted into digital streams for further processing downstream. Data acquisition systems (DAS) perform these data aggregation and conversion functions. The DAS connects to the sensor network, aggregates outputs, and performs the analog-to-digital conversion. The Internet gateway receives the aggregated and digitized data and routes it over Wi-Fi, wired LANs, or the Internet.

These systems often sit in close proximity to the sensors and actuators. For example, a pump might contain a half-dozen sensors and actuators that feed data into a data aggregation device that also digitizes the data. This device might be physically attached to the pump. An adjacent gateway device or server would then process the data and forward it to the next Stage.

Why preprocess the data? The analog data streams that come from sensors create large volumes of data quickly. The measurable qualities of the physical world in which your business may be interested—motion, voltage, vibration, and so on—can create voluminous amounts of constantly changing data. Think how much sensor data a complex machine like an aircraft engine might generate in one day, and there's no theoretical limit to the number of sensors that could be feeding data into an IoT system. What's more, an IoT system is always on, providing continuous connectivity and data feeds. IoT data flows can be immense—I've seen as much as 40 TB/second in one case. That's a lot of data to transport into the data center. It's best to preprocess it. This is further discussed in para 1.7.

Another reason not to pass the data on to the data center in this form is that analog data has specific timing and structural characteristics that require specialized software to process. It's best to convert the data into digital form first.

Intelligent gateways can build on additional, basic gateway functionality by adding such capabilities as analytics, malware protection, and data management services. These systems enable the analysis of data streams in real time. Although delivering business insights from the data is a little less immediate at the gateway than it would be when sent directly from the sensor/actuator zone, the gateway has the compute power to render the information in a form that is more understandable to business stakeholders.

Gateways are still edge devices—they're external to the data center—so geography and location matter. In the pump example, if you have 100 pump units and want to process data on-premises, you might have instant data at the pump level, aggregate the information to create a plantwide view for the facility, and pass the data on to the data center for companywide view. DAS and gateway devices may end up in a wide variety of environments, from the factory floor to mobile field stations, so these systems are usually designed to be portable, easy to deploy, and rugged enough to withstand variations in temperature, humidity, dust, and vibration. Pre-processing of data aspect is further dealt in this book in para.1.11

1.2.4. Edge IT

Once IoT data has been digitized and aggregated, it's ready to cross into the realm of IT. However, the data may require further processing before it enters the data center. This is where edge IT systems, which perform more analysis, come into play. Edge IT processing systems may be located in remote offices or other edge locations, but generally these sit in the facility or location where the sensors reside closer to the sensors, such as in a wiring closet.

Because IoT data can easily eat up network bandwidth and swamp your data center resources, it's best to have systems at the edge capable of performing analytics as a way to lessen the burden on core IT infrastructure. If you just had one large data pipe going to the data center, you'd need enormous capacity. You'd also face security concerns, storage issues, and delays processing the data. With a staged approach, you can preprocess the data, generate meaningful results, and pass only those on. For example, rather than passing on raw vibration data for the pumps, you could aggregate and convert the data, analyze it, and send only projections as to when each device will fail or need service.

Here's another example: You might use machine learning at the edge to scan for anomalies that identify impending maintenance problems that require immediate attention. Then you could use visualization technology to present that information using easy-to-understand dashboards, maps, or graphs. Highly integrated compute systems, such as hyper-converged infrastructure, are ideally suited to these tasks because they're relatively fast, and easy to deploy and manage remotely.

1.2.5. IoT Service Providers

Internet of Things (IoT) services represents a set of end-to-end services in which businesses contract with external providers to design, build, install and operate IoT solutions, including advisory consulting for IoT planning. IoT service providers represent a range of small, midsize and large service firms that build and deploy IoT solution applications across industries. The focus of this market is on the medium and large service providers supporting key vertical markets for IoT adoption such as manufacturing, healthcare, transportation and retail. This market's IoT service focus aligns with the design, build and install of an IoT solution and includes IoT planning services for an IoT-enabled digital business environment.

1.2.6. IoT infrastructure

IoT infrastructure provides a set of managed services to support IoT solutions from the edge of the network (sensors/actuators), through gateway devices, across managed M2M connections, to a Cloud K5 IoT platform where data can be presented to the digital business applications that drive the innovation and value from IoT.

An IoT solution typically consists of various levels of hardware/software integration:

1. Physical Devices and Controllers
2. Connectivity
3. Edge Computing
4. Data Accumulation and Abstraction
5. Application
6. Collaboration and Processes

The IoT services layer is completely independent from the underlying devices, communication protocols and connectivity semantics. This layer includes a core set of services to build IoT applications (i.e. composite applications) across a range of industry sectors. The IoT services layer helps the enterprise to:

1. Analyze data in real-time (event processing)
2. Act on machines to machine (M2M) data and events (integration services)
3. Provide historical, real-time and predictive analytics (analytics services)
4. Visualize operational and analytical data through mobile/desktop (UI = User Interface services).
5. Manage data security and identity of devices/apps (security and identity management Service)
6. The IoT developer services layer enables developers to build applications using IoT services, development kits, software tools and services. This layer helps expose the platform to a range of applications and use-cases.

   The role of middleware is to provide the infrastructure and IoT services which in turn help drive innovation, enable new revenue streams, and improve operational efficiencies.

1.2.7. IoT Development Service

IoT Development Services include:

1. Implementation of Iot
2. Iot Platforms Development
3. Iot Module Development
4. Iot Testing

1.2.8. Machine-to-Machine(M2M) and IoT Technology

M2M, or, is a direct communication between devices using wired or wireless communication channels. M2M refers to the interaction of two or more devices/machines that are connected to each other. These devices capture data and share with other connected devices, creating an intelligent network of things or systems. Devices could be sensors, actuators, embedded systems or other connected elements.

M2M technology could be present in our homes, offices, shopping malls and other places. Controlling electrical appliances like bulbs and fans using RF or Bluetooth from your smartphone is a simple example of M2M applications at home. Here, the electrical appliance and your smartphone are the two machines interacting with each other.

Some of the differences between **M2M** and the **IoT** are listed in the table 1.1.

Table 1.1.1Differences between M2M and the IoT

M2M	IoT
M2M is about direct communication between machines.	The IoT is about sensors automation and Internet platform.
It support point-to-point communication.	It supports cloud communication.
Devices do not necessarily rely on an Internet connection.	Devices relay on an Internet connection.
M2M is mostly hardware-based technology	The IoT is both hardware- and software-based technology.
Machines normally communicate with a single machine at a time.	Many users can access at one time over the Internet.
A device can be connected through mobile or other network.	Data delivery depends on the Internet protocol (IP) network.

1.3. BUSINESS PROCESS MANAGEMENT IN IOT

1.3.1. IoT Data Management

The IoT is changing the way we live our lives and that is something that will only grow and grow, and it's certainly something that all businesses need to adapt to. There are some obvious benefits and some aspects that will require adjustments to processes. Here are some of the main changes and challenges facing companies as the IoT becomes more ever-present:

Data: As consumers use more and more devices that record data, there are opportunities for businesses to use this data for marketing and product development purposes, but only if the processes are in place to measure, analyze and report on this data. Business process management can automate this process and ensure that it remains effective and agile enough to keep pace with technological changes.

The last capability you should consider in a IoT data platform is how you can query the data in a manner that is clear and meaningful. It's one thing to get all your data in place, but the value of the data is only realized when it's turned into information that can help solve a problem. We want organizations to focus on their core competency, like making great appliances or services that deliver value to their customers, rather than focusing on cloud infrastructure that makes it possible. At Buddy, our job is to provide an end-to-end turn-key solution to connect the world's IoT devices and provide real-time business insights for decision making. That can take the form of simple dashboard or deep analytics through integration with partners and services.

Keep it simple when you're evaluating IoT data platform options. If a platform is connected, allows two-way communication to the device, device management and a visual IoT data graph, these are the main areas you should focus on during your process.

1.3.2. Business Process Management (BPM)

Business process management is the way that companies analyze the processes that have been designed and introduced to help them operate. Often these processes can become bogged down in minutiae and end up being less efficient than they should be, so BPM is there to evaluate what improvements are needed and to then implement them in a way that minimizes disruption. This should lead to better efficiency, improved staff morale and an upturn in profits.

Business Process Management was established to analyze, discover, design, implement, execute, monitor and evolve collaborative business processes within and across organizations.

New ways of buying: The IoT gives consumers the chance to buy directly from their devices, whether it's an Amazon Echo or a smartphone or even that legendary refrigerator ordering fresh milk. Technology is making everything faster and more easily, so they will also be expecting faster deliveries and better service. BPM needs to be used to manage the processes that will allow this kind of development to meet the demand. IoT software and tools can help with this though, with inventories able to be tracked automatically.

Innovation: Whether it's new product development or upgrading existing products or services, the IoT offers the opportunities for businesses to deliver exciting new benefits for their customers.

Customer service: Another area where processes need to be managed carefully because of the changes that the IoT have brought in is customer service. Products that utilize the internet should really be able to be fixed over the internet when something goes wrong. Consumers expect it and businesses should be able to deliver it, so BPM is needed to ensure that customer service processes are effective, efficient and resilient enough to cope.

Centralized BPM: Business process management isn't simply something that is needed to make the IoT run more smoothly, the benefits can flow back in the opposite direction too. Integrating BPM software into devices means that the data can be analyzed from a central location and any changes can be fed back out again.

1.3.3. Integrating Business Process Management (BPM) with the Internet of Things

These are some of the benefits and implications of BPM and the Internet of Things. Smart devices are taking over our homes and workplaces, but they will only be as successful as the processes that help to manage them and their applications. Competition is fierce, whether it's between the Amazon Echo and Google Home or other rival products that aim to corner the market, while the Internet of Things in a business sense either touches or will touch almost all companies in years to come.

A Price water house Coopers report, Sensing the Future of the Internet of Things predicted that: "IoT is transforming the everyday physical objects that surround us into an ecosystem of information that will enrich our lives. From refrigerators to parking spaces to houses, the IoT is bringing more and more things into the digital fold every day, which will likely make the IoT a multi-trillion dollar industry in the near future."

The businesses that will thrive in the future are those that are prepared for the changes to come and that is where business process management will always give some an advantage. Jared Cooper in Fast Company summed up the perils facing those who don't get their processes in line in time: "Smart homes and other connected products won't just be aimed at home life. They'll also have a major impact on business. And just like any company that blissfully ignored the Internet at the turn of the century, the ones that dismiss the Internet of Things risk getting left behind."

If you don't want your business to be left behind by the Internet of Things, you may need some help with business process management, and that's where Tallyfy comes in. We can help you get your processes ready for the technological challenges that are coming with a free demonstration of our BPM software. All you need to do is get in touch and be ready to see your business running like clockwork.

1.3.4. Legacy BPM Software

"Old BPM" software is tired and broken. It never worked for business users. Here's why:

Users are now deciding to buy software themselves. Old BPM was bought by your IT department, who didn't generally care about user experience – as long as it was made by a large/boring company.

Cloud tools are now free to try by anyone, anytime. With Old BPM you had to call sales and wait for 50 questions just to look at it and finally decide it sucks.

People want to share workflows with clients. With Old BPM you were stuck with trying to automate internal processes only. Your clients would be very scared and run a mile from it.

People expect to integrate cloud tools without IT. With Old BPM you had get engineers to write code to make a simple integration. That's now become a drag-and-drop service.

People expect to work on phones. This means giant, clunky flowcharts in Old BPM are dead – because they don't fit on your phone's screen – and only define "the perfect process".

People are tired of flowcharts. Old BPM was all about the high priest telling you how a process can/will be done, and you would obey. Now – modern workers and teams are paid to collaborate.

People expect all the benefits of the cloud. Old BPM was never cloud-born and was never designed for the cloud. And that creates a massive bunch of missed opportunities.

Companies of all sizes need process management – and never had it. Since Old BPM was so expensive and complicated, only large companies could afford it. The rest of us were left out.

People are excited about AI – but confused about where to begin. With Old BPM you have zero chance of using AI without an army of engineers. With cloud-born systems like Tallyfy – it's childs' play to use any AI you like to run amazing automations for photos, voice, video and more.

1.3.5. Benefits of IoT and BPM Integration

Embedding intelligence by way of real-time data gathering from gateways and devices and consuming them through business processes helps businesses achieve not just cost savings and efficiency but also helps them generate more revenue patterns. Businesses need to overcome several business and service challenges to be able to realize smooth orchestration and manageability of disparate systems.

The power of the Internet of Things (IoT) comes from extracting and exploiting process, business, and customer data that are locked inside enterprises. Inside devices, inside machines, inside infrastructure. These data boost the productivity of human and capital assets, enhance visibility into processes, help secure enterprises against attack, and drive profitability by identifying new business opportunities. The IoT value cycle explores how IoT data can improve operations by optimizing four parameters: visibility, security, profitability, and productivity. Enhanced visibility is achieved by securely connecting all data sources to processes, using algorithms to extract value. Security comes from protecting company, employee, and customer IoT data, in-motion or at-rest, from attack, and by governing those data throughout their life cycles. The third parameter, profitability, is achieved by using IoT solutions to better understand customers and their preferences. Productivity, the fourth and final parameter, is achieved by using IoT data to squeeze the most from production processes and empower the people who run them.

1.4. HOW CAN ORGANIZATIONS PROFIT FROM THE IOT?

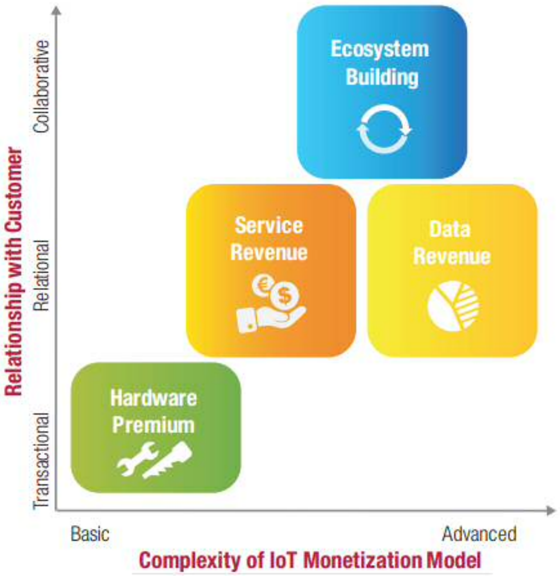


Fig. 1.10. Monetization Models for the IoT (Source: Capgemini Consulting analysis Product Selling is an Organization’s)

1. **Ecosystem Building (Referring to Fig. 1.10)**

In this model, organizations create a platform where they ideally make money from both other product vendors and end consumers .

Example: Smart Things sell sits own products and services while creating a platform for other IoT companies to sell services that interlink with it. Smart Things is an Internet of Things startup that offers a centralized hub and an assortment of both in-house and third-party IoT products. The company was launched in 2012 and raised over $1.2 million on Kickstarter (a crowd funding platform) within 18 months. The company has a smart phone app that is used to control its hub and all of its connected devices. It offers broad guidance to developers who want to make products for its platforms offering them design guidelines. The hub is priced at $99. Various products that the company sells as part of the platform include locks, switches, environment sensors, alarms among others. Smart Things works with partners such as Belkin, Sonos, and Philips, and on operating systems such as Android and iOS. Over 1,000 devices and 8,000 applications have been made till August 2014 when the company was acquired by Samsung Electronics for approximately $200 million.

Ecosystem Building Allows Monetization from Dual-Sided Markets The IoT thrives in a connected ecosystem – the bigger the ecosystem, the greater is the value generated for all stakeholders. In an ecosystem, the focus is not on selling a product or a service, but on providing a shared platform to other players in the ecosystem – hardware manufacturers, software developers, service providers and the like. In such a model, the platform promoter ideally makes money from both end customers as well as other platform users. Platform users pay the promoter for listing and the promoter also gets a share whenever a product is sold to the end customer on the platform. A shared platform brings multiple benefits to participants.

2. **Hardware Premium** is the most basic form of monetization model (figure 1.10). Here, organizations add connectivity options to an existing or new product and offer remote device

management in the form of mobile apps. This basic level of connectivity and control enables organizations to charge a premium for their product.

An example of this model is LIFX which produces remotely programmable LED light bulbs that can be controlled through a smart-phone app. These bulbs are sold at a premium, and are priced around 10 times higher than a compact fluorescent bulb.

From a consumer perspective, a key driver for buying hardware premium products is the novelty factor involved in controlling hitherto standalone devices.

3. **Service revenue.** In this model, organizations convert what has been a traditional product into a service by tying in a recurring pricing model for specific features

Example. Volkswagen's "Car-Net" service off security features, maintenance assistance and navigation tools for a set subscription fee.

4. **Data Revenue.** In this model organizations generate revenues by selling packaged data gathered from Sensors.

Example. Michelin Solutions packages insights generated from the data that it gathers through sensors embedded inside customer vehicle.

In the video game industry, **games as a service** (**GAAS**) represents providing video games or game content on a continuing revenue model, similar to software as a service. Games as a service are ways to monetize video games either after their initial sale, or to support a free-to-play model. Games released under the GaaS model typically receive a long or indefinite stream of monetized new content over time to encourage players to continue paying to support the game.

1.5. WHAT IS THE IOT VALUE CHAIN

The IoT value chain explains the building blocks of IoT, how value is created, who they players are, and how they interact with each other to deliver value.

Looking at the IoT value chain as a pyramid, at the base is all the connected devices: phones, fitness bands, connected cars, smart homes, and other devices on the consumer side; in industry, you have things like building sensors, smart cities, and connected factories, for example.

Stepping up a level from the base brings in the network and connectivity—how devices are connected and communicate. It's also where service providers collect device and network data and upload it to the cloud.

Finally, at the top of the value chain, are applications and services that are closest to the eventual end users—enterprises and consumers.

1.5.1. How Can Service Providers Move up In the Chain?

For instance, telecom carriers may offer cellular and non-cellular IoT network integration services or device lifecycle management on behalf of enterprise customers. These diversifications lay a foundation for them to adopt more sophisticated roles in the future.

Of course, each of these roles and services has special requirements and skills needed. This means providers will have to expand out of their comfort zone if they want to move up the IoT value chain.

Look at the example of self-driving cars. When a critical mass of these vehicles are operating on the street, they'll require extremely reliable and fast connections like 5G to communicate constantly. Service providers are in pole position to connect the vehicles. But, there is clearly the potential to provide more, like applications in those cars, or platforms for them to communicate.

Fig. 1.11 showcases three different ways to move up the value chain as per Ericsson's Exploring IoT strategies report.

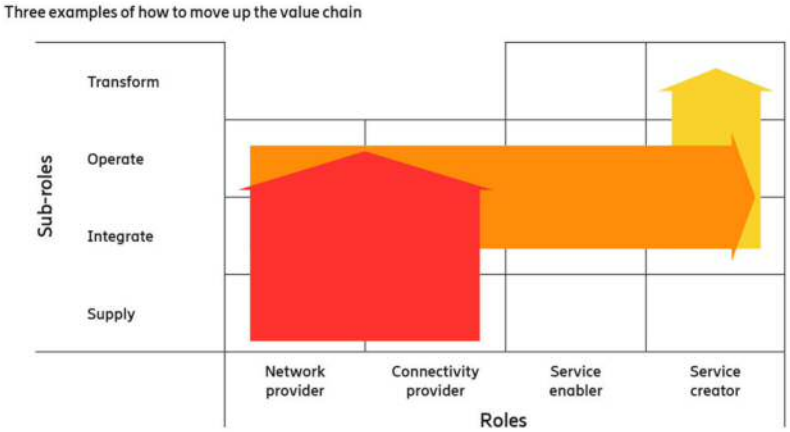


Fig. 1.11. Three different ways to move up the value chain.

1.6. IOT VALUE CYCLE

Fig. 1.12 describes the IoT value Cycle divided in four groups, each are to be studied in details as per their respective parameter.

1. Visibility Parameters

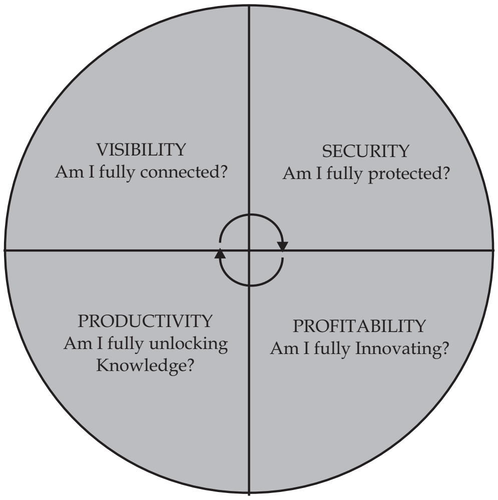


Fig. 1.12. IoT Value Cycle

(*a*) M2M, cellular, and telematics

(*b*) Industrial grade wireless

(*c*) Switching and data centers

(*d*) Remote sites, users, data centers

(*e*) Management of devices, users, apps

Telematics can involve any of the following:

1. the technology of sending, receiving and storing information using telecommunication devices to control remote objects.

2. the integrated use of telecommunications and informatics for application in vehicles and to control vehicles on the move.
3. global navigation satellite system technology integrated with computers and mobile communications technology in automotive navigation systems.

2. Security Parameters

(*a*) Data at-rest and in-motion
(*b*) Physical security
(*c*) Secure BYOD
(*d*) Application security
(*e*) Compliance, health, and safety

BYOD is short for "Bring Your Own Device," a phrase that refers to the practice of allowing employees to bring their own mobile devices to work for use with company systems, software, networks, or information. The use of cloud tools in the enterprise is becoming increasingly common, enabling employees to collaborate and work incredibly efficiently. On top of this, when employees are allowed to work from their personal devices (known as bring your own device or BYOD), it makes it even easier for them to share information and complete tasks. However, BYOD also makes it more difficult for businesses to oversee and protect the flow of corporate data. In light of this, Bit glass surveyed IT experts to learn about what organizations are doing to secure BYOD.

3. Profitability Parameters

(*a*) Service excellence
(*b*) Engagement and differentiation
(*c*) Ease of use and interaction
(*d*) Loyalty and product validation
(*e*) Monetization as a service (to be discussed in para 1,7)

Johnston, R. & Clark G. have written in their book Service operations management book that Service Excellence means that it is not about exceeding the expectations of customers, but primarily about "delivering what is promised and dealing well with any problems and queries that arise". We often hear companies saying that good customer service is very important for them, but in reality, actions do not seem to support that statement. Instead of that many organizations today tend to focus only on choosing faster and easier ways to get fast recognition and easy money. Company managers forget that front-line determines the success of the company.

We, as a customers, do not care about back-office procedures and efforts, we just want to be noticed, served and that our need will be met, so that we leave our money to the company. We do care about Excellent Service.

Customer Service is an integral part of a large or small company. It is easier to retain a customer than find a new one. Without a good customer service, companies would lose clients to their competition, which is the last thing an organization wants especially in these tough times.

Montgomery Emerson in (with 25 years experience in various industries as a business consultant. Specialist in Customer Service, Customer Relationship Management,) in his article deals with presenting yourself in the best position possible to deliver superior customer service. These ideas are not new, but they are tried and true principles that will set you apart from others.

1. **Be a Great Listener.** Customers love to vent. They want you to physically feel the emotional angst they are going through at that moment in time when you pickup the call. A Customer Service representative needs to have the patience to listen to the issues and problems before jumping in with a solution. As a great listener you will immediately be a credible resource to the person on the other end within 30 seconds. Take the time to hear what they have to say before jumping in with your resolution. If you interrupt the flow of the information coming in, the customer will immediately sense you are not paying attention and become even more irate and ask to talk to someone else. Which is the last thing you want to have happen.

2. **Phone Presence.** As a Customer Service Representative, you will most likely be on the phone 90% of the time. Having the ability to speak articulately and pronounce words so that you are clear and easily understood is a crucial in this role. A representative that cannot get their point across to a customer when they need help will only frustrate them further.

3. **Promote the Desire to Help.** A Customer Support Representative needs to have the fundamental desire to assist others. To understand their issues quickly and provide solutions that will make them feel that the company cares about them, no matter what the issue is. If you do not have this basic character trait, then a customer support position may not be for you.

4. **Dedicated Problem Solver.** The reason you are in this position, is to listen to the problem being conveyed by the customer and offering solutions to those issues, quickly. Chances are, the company has given you the necessary training, knowledge and processes to fulfill this obligation. Your job is to execute the process based on the training and information you have. In addition, you will have a support system of supervisors and knowledge experts that you can draw from if needed.

5. **Patience.** Another important character trait that is difficult to master is patience and empathy. Having the patience to listen to the customer and work through each call is a critical factor to keep your own head when others are screaming at you over the phone. It is easy to get caught up in the emotions being forced upon you, causing premature burn out. Having the ability to be a calming influence can help relieve the stress the customer may be feeling at that time.

6. **Communication.** The art of Communication is a learned skill that customer service representatives need to become more aware of as they develop over time. It is easy to to tell a customer that you can help them because that is what they want. Telling a customer "No" is a difficult task, but can be done with respect and sincerity.

7. **Sincerity and Honesty.** A Customer will know when you are being truly honest with them or condescending. If you present the facts and your position in wanting to help them through the issues with best possible solution, they will be very appreciative of your efforts.

8. **Do as You Say.** Establishing trust is critical in satisfying the customer. If you are going to investigate and call them back, do it. Make a commitment and stick to it. It will become a principle that many customers will expect and know you will deliver on.

4. Productivity Parameters

(*a*) Uptime, high MTBF, low MTTR
(*b*) Customer behavior
(*c*) Contractor and staff management
(*d*) Kanban, efficiency, and throughput
(*e*) Responsiveness

MTBF, or Mean Time Between Failures, is a metric that concerns the average time elapsed between a failure and the next time it occurs. These lapses of time can be calculated by using a formula. Whereas the MTTR, or Mean Time To Repair, is the time it takes to run a repair after the occurrence of the failure.

Kanban (*signboard or billboard* in Japanese) is a scheduling in an inventory control system used in just-in-time manufacturing for lean manufacturing and just-in-time manufacturing (JIT). Taiichi Ohno, an industrial engineer at Toyota, developed kanban to improve manufacturing efficiency. Kanban is one method to achieve JIT. The system takes its name from the cards that track production within a factory.

1.7. MONETIZATION AS A SERVICE

Monetization refers to the conversion of an investment into cash, which usually happens via a liquidity event. When investors, such as venture capital or private equity firms, invest in a private company, they expect to get their money back plus a specified return within a certain period of time.

Monetization goes hand-in-hand with capitalism—and is just about as old

There are many ways to make money from your website. In addition to advertising networks that let you earn money by displaying ads on your site, there are other effective ways to make money by monetizing content.

Website owners and bloggers have been successfully monetizing content to generate revenue for quite some time. However, in spite of the many monetization options, it’s not always easy for publishers to make money online. Competition among publishers is fierce and website owners need to be savvy and come up with new ways to generate revenue.

1. **Publishing your content on Amazon** Kindle’s new blog publishing platform is a great way to make money from your blog. This is particularly beneficial if you already have a blog. All you need to do is add it to Kindle’s publishing platform and start making money from your existing content. This facility is currently available only to UK and US residents only. Although you will only get 30 percent share of total revenue generated, it can still be good source of added revenue for your blog content.
2. **Taggstar** is a free widget that makes the images on your site interactive, engaging and social. It aims to give publishers a new way of monetizing images by adding layers of content to make them interactive and shareable. It brings your images to life whilst offering the potential to generate revenue. It’s unique ‘Shop the image’ feature turns pictures on your website into a shopping experience so you can earn from them.
3. **The Nimble Network** claims to be the first ever peer to peer deal sharing network. It allows you to make money by providing targeted offers, solutions and products that publishers can give their audiences. It is a premium transactional offers network which connects local and national advertisers to consumers through a network of publishers. It allows publishers to share deals.
4. **Virurl** allows you to make money by monetizing your content. They are not displayed as typical ads so users are more likely to click and ensures a higher click through. There are two ways to earn money, by displaying a widget on your website or blog content as well as by monetizing your social stream. You can get started and earn money by installing the content discovery widget on your website and sharing links on social platforms. This allows you to maximize your potential revenues as it extends your reach to a larger audience.

5. **Web Answers** is a unique Ad Sense revenue-sharing Question and Answer site where you can get paid to answer questions. Its simplicity and earning potential really make it a great site.

There are two main ways to earn money on this site. You earn ongoing advertising royalties on all questions awarded to your account via "Best Answer". Additionally, you have the opportunity to earn a share of revenue generated by displaying advertisements on your behalf on other areas of the site. Your share is calculated via a proprietary formula used on this site. The more value you provide, the higher will be your revenue potential.

1.8. ROLE OF CLOUD TECHNOLOGY IN IOT COMPUTING

The word cloud is used to denote free or chargeable services available by service providers for using remote servers**.**

IoT is redefining experiences between people, machines, and facilities.

The network will provide connectivity, power, policy, compute, security and manageability at scale to IoT deployments. The IoT devices will most definitely need connectivity to the controllers that will be controlling the devices. The connectivity to the network could be wired or wireless. The number of connected devices is expected to grow to 50 billion by the year 2020. There is a more bullish estimate of this – 200 billion by the year 2020. That would be anywhere between 6 and 24 devices for every person on earth.

1.8.1. Cloud Computing

In the simplest terms, cloud computing means storing and accessing data and programs over the Internet instead of your computer's hard drive. The cloud is just a metaphor for the Internet.

Cloud Computing is the on-demand availability of computer system resources, specially data storage and computing power, without direct active management by the user. The term is generally used to describe data centers available to many users over the Internet. Large clouds, predominant today, often have functions distributed over multiple locations from central servers. If the connection to the user is relatively close, it may be designated an **edge server.**

1.8.2. Types of Cloud Services

Organizations of every type, size, and industry are using the cloud for a wide variety of use cases, such as data backup, disaster recovery, email, virtual desktops, software development and testing, big data analytics, and customer-facing web applications. For example, healthcare companies are using the cloud to develop more personalized treatments for patients. Financial services companies are using the cloud to power real-time fraud detection and prevention. And video game makers are using the cloud to deliver online games to millions of players around the world.

(*a*) Personal Cloud Computing

(*b*) Public and

(*c*) Hybrid

1. **Personal Cloud Computing.** The following are the major examples of cloud computing readers may be probably using:

(*a*) **Google Drive.** This is a pure cloud computing service, with all the storage found Online so it can work with the cloud apps: Google Docs, Google Sheets, and Google Slides. Drive is also available on more than just desktop computers; you can use it on tablets like the iPad or on smart phones, and there are separate apps for Docs and Sheets, as well. In fact, most of Google's services could be considered cloud computing: Gmail, Google Calendar, Google Maps, and so on.

(*b*) **Apple iCloud.** Apple's cloud service is primarily used for online storage, backup, and synchronization of your mail, contacts, calendar, and more. All the data you need is available to you on your IOS, Mac OS, or Windows device (Windows users have to install the iCloud control panel). Naturally, Apple won't be outdone by rivals: it offers cloud-based versions of its word processor (Pages), spreadsheet (Numbers), and presentations (Keynote) for use by any iCloud subscriber. iCloud is also the place iPhone users go to utilize the Find My iPhone feature that's all important when the handset goes missing.

(*c*) **Amazon Cloud Drive**. Storage at the big retailer is mainly for music, preferably MP3s that you purchase from Amazon, and images—if you have Amazon Prime, you get unlimited image storage. Amazon Cloud Drive also holds anything you buy for the Kindle. It's essentially storage for anything digital you'd buy from Amazon, baked into all its products and services.

2. **Hybrid Cloud Services.** Hybrid services like *Box, Drop box*, **and** *Sugar Sync*, all say they work in the cloud because they store a synced version of your files online, but they also sync those files with local storage. Synchronization is a cornerstone of the cloud computing experience, even if you do access the file locally.

Likewise, it's considered cloud computing if you have a community of people with separate devices that need the same data synced, be it for work collaboration projects or just to keep the family in sync.

3. Public Cloud Services. Public clouds are owned and operated by companies that offer rapid access over a public network to affordable computing resources. With public cloud services, users don't need to purchase hardware, software or supporting infrastructure, all of which is owned and managed by providers.

Enterprises create applications and software through cloud services (SaaS, we will read about this in para 1.17, which can connect devices and enable device registration, on-boarding, remote device updates, and remote device diagnosis in minimal time with a reduction in the operational and support costs. Cloud introduces DevOps within the IoT ecosystem, which helps organizations automate many processes remotely.

1.8.3. Simple Architecture of Cloud Computing

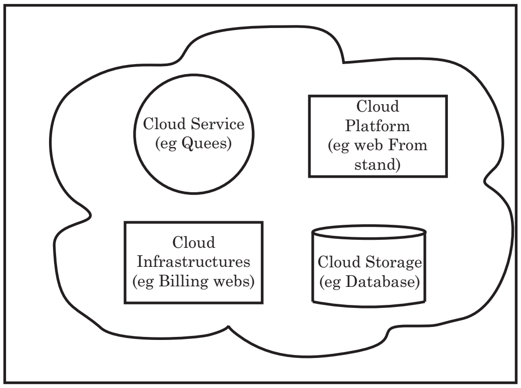


Fig. 1.13. Simple Architecture of Cloud Computing

These components typically consist of a front end platform (fat client, thin client, mobile device), back end platforms (servers, storage), a cloud based delivery, and a network (Internet, Intranet, Intercloud).

Clients are just the desktops where they have their place on desks. These might be also in the form of laptops, mobiles, tablets to enhance mobility. Clients hold the responsibility of interaction which pushes for the management of data on cloud servers.

Clouds may be limited to a single organization (enterprise clouds, or be available to many organizations (public cloud).

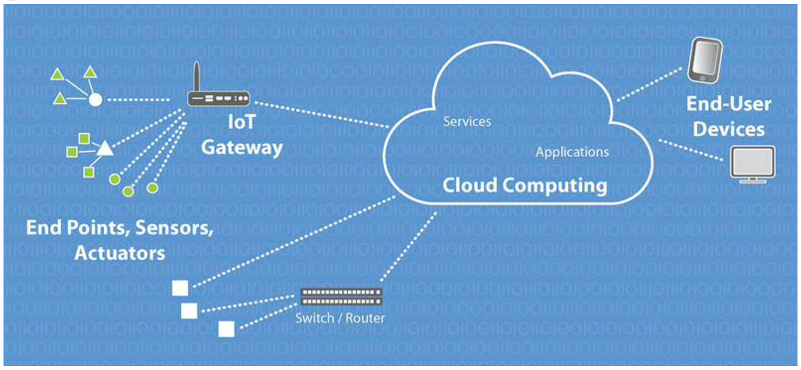


Fig. 1.14. Using Cloud computing via IoT gateways

Advocates of public and hybrid clouds envisage that cloud computing allows companies to avoid or minimize up-front IT infrastructure costs. Proponents also claim that cloud computing allows enterprises to get their applications up and running faster, with improved manageability and less maintenance, and that it enables IT teams to more rapidly adjust resources to meet fluctuating and unpredictable demand.1 Cloud providers typically use a "pay-as-you-go" model, which can lead to unexpected operating expenses if administrators are not familiarized with cloud-pricing models.

The availability of high-capacity networks, low-cost computers and storage devices as well as the widespread adoption of hardware virtualization, service-oriented architecture and autonomic and utility computing has led to growth in cloud computing.[2] By 2019, Linux was the most widely used operating system, including in Microsoft's offerings and is thus described as dominant. The Cloud Service Provider (CSP) will screen, keep up and gather data about the firewalls, intrusion identification or/and counteractive action frameworks and information stream inside the network.

1.8.4. Key Characteristics of Cloud computing

Cloud computing exhibits the following key characteristics:

1. Agility for organizations may be improved, as cloud computing may increase users' flexibility with re-provisioning, adding, or expanding technological infrastructure resources.
2. Cost reductions are claimed by cloud providers. A public-cloud delivery model converts capital expenditures (e.g., buying servers) to operational expenditure. This purportedly lowers barriers to entry, as infrastructure is typically provided by a third party and need not be purchased for one-time or infrequent intensive computing tasks. Device and location independence enable users to access systems using a web browser regardless of their location or what device they use (e.g., PC, mobile phone). As infrastructure is off-site (typically provided by a third-party) and accessed via the Internet, users can connect to it from anywhere.
3. Maintenance of cloud computing applications is easier, because they do not need to be installed on each user's computer and can be accessed from different places (e.g., different work locations, while traveling, etc.).
4. Multi tenancy enables sharing of resources and costs across a large pool of users thus allowing for:

(*a*) centralization of infrastructure in locations with lower costs (such as real estate, electricity, etc.)

(*b*) peak-load capacity increases (users need not engineer and pay for the resources and equipment to meet their highest possible load-levels)

(*c*) utilisation and efficiency improvements for systems that are often only 10–20% utilised.

5. Productivity may be increased when multiple users can work on the same data simultaneously, rather than waiting for it to be saved and emailed. Time may be saved as information does not need to be re-entered when fields are matched, nor do users need to install application software upgrades to their computer.

6. Reliability improves with the use of multiple redundant sites, which makes well-designed cloud computing suitable for business continuity and disaster recovery.

7. Security can improve due to centralization of data, increased security-focused resources, etc., but concerns can persist about loss of control over certain sensitive data, and the lack of security for stored kernels. Security is often as good as or better than other traditional systems, in part because service providers are able to devote resources to solving security issues that many customers cannot afford to tackle or which they lack the technical skills to address. However, the complexity of security is greatly increased when data is distributed over a wider area or over a greater number of devices, as well as in multi-tenant systems shared by unrelated users. In addition, user access to security audit logs may be difficult or impossible. Private cloud installations are in part motivated by users' desire to retain control over the infrastructure and avoid losing control of information security.

1.8.5. Need of Cloud Computing in IoT Applications

Cloud computing, as well as IoT, work towards increasing the efficiency of everyday tasks and both have a complementary relationship. On one hand, IoT generates lots of data while on the other hand, cloud computing paves way for this data to travel. There are many cloud providers who take advantage of this to provide a pay-as-you-use model where customers pay for the specific resources used. Also, cloud hosting as a service adds value to IoT startups by providing economies of scale to reduce their overall cost structure.

Just like cloud computing is built on the tenets of speed and scale, IoT applications are built on the principle of mobility and widespread networking. Hence, it is essential that both cloud and IoT form cloud-based IoT applications in a bid to make the most out of their combination. This alliance has led to the success of IoT. In addition to this, here are a few more pointers as to why the cloud is important from the point of view of IoT's success.

1. Provides remote processing power

2. Cloud as a technology empowers IoT to move beyond regular appliances such as air conditioners, refrigerators etc. This is because the cloud has such a vast storage that it takes away dependencies on on-premise infrastructure. With the rise of miniaturization and transition of 4G to higher internet speeds, the cloud will allow developers to offload fast computing processes.

3. Provides security and privacy

4. IoT's role in harnessing mobility is immense. However, its prowess would be incomplete without security. Cloud has made IoT more secure with preventive, detective and corrective controls. It has enabled users with strong security measures by providing effective authentication and encryption protocols. In addition to this, managing and securing the identity of users has been possible for IoT products with the help of biometrics. All of this is possible because of cloud's security.

5. Removes entry barrier for hosting providers
6. Today, many innovations in the field of IoT are looking at plug-and-play hosting services. Which is why the cloud is a perfect fit for IoT. Hosting providers do not have to depend on massive equipment or even any kind of hardware that will not support the agility IoT devices require. With the cloud, most hosting providers can allow their clients a ready-to-roll model, removing entry barriers for them.
7. Facilitates inter-device communication
8. Cloud acts as a bridge in the form of a mediator or communication facilitator when it comes to IoT. Many powerful APIs like Cloud flare, Cloud Cache and Dropstr are enabled by cloud communications, allowing easy linking to smart phones. This eases devices to talk to each other and not just us, which essentially is the tenet of IoT cloud.

It would be fair to say that cloud can accelerate the growth of IoT. However, deploying cloud technology also has certain challenges and shortcomings. Not because the cloud is flawed as a technology but the combination of IoT cloud can burden users with some obstacles. If you ever go ahead with an IoT cloud solution, it is better if you know the kind of challenges you may face in advance.

1.8.6. What are the challenges posed by cloud and IoT together?

1. **Handling a large amount of data.** Handling a large amount of data can be overwhelming especially when there are millions of devices in the picture. This is because the overall performance of applications is at stake. Hence, following the NoSQL movement could be beneficial, but it is not tried and tested for the long run. Which is why there exists no sound or fool-proof method for the cloud to manage big data.
2. **Networking and communication protocols.** Cloud and IoT involve machine-to-machine communications among many different types of devices having various protocols. Managing this kind of a variation could be tough since a majority of application areas do not involve mobility. As of now WiFi and Blue tooth are used as a stop-gap solution to facilitate mobility to a certain extent.

1.8.7. Popular Cloud based IoT Platforms

The following platforms had been adjudged as most popular Cloud based IoT Platforms

1. Amazon Web Services IoT Platform
2. Microsoft Azure IoT Hub
3. IBM Watson IoT Platform
4. Google Cloud Platform
5. Oracle
6. Sales force
7. Bosch
8. Cisco IoT Cloud Connect
9. General Electrics Predix
10. SAP

We will discuss them one by one.

1. **Amazon Web Services IoT Platform**

Amazon dominates the consumer cloud market. They were the first to really turn cloud computing into a commodity way back in 2004. Since then they've put a lot effort into innovation and building features, and probably have the most comprehensive set of tools available.

Pricing is based on messages sent and received by AWS IoT. Each IoT interaction can be thought of as a message between a device and a server. Amazon charges per million messages sent or received. There are no minimum fees, and you won't get charged for messages to the following AWS services:

(*a*) Amazon S3

(*b*) Amazon Dynamo DB

(*c*) AWS Lambda

(*d*) Amazon Kinesis

(*e*) Amazon SNS

(*f*) Amazon SQS

2. Microsoft Azure IoT Hub

They have cloud storage, machine learning, and IoT services, and have even developed their own operating system for IoT devices. This means they intend to provide a complete IoT solution provider.

The pricing is done in 4 tiers based on how much data your devices will generate. Below 8,000 messages per unit per day is free. It does get complicated when you start to integrate with other Microsoft services, but they have a great pricing calculator to help you out.

Like Amazon, Google, Oracle and IBM, Microsoft also has some other cool services you can use on their could platform. These include things like machine learning data analytics so you can build some really cool applications

3. IBM Watson IoT Platform

IBM is another IT giant trying to set itself up as an Internet of Things platform authority. They try to make their cloud services as accessible as possible to beginners with easy apps and interfaces. You can try out their sample apps to get a feel for how it all works. You can also store your data for a specified period, to get historical information from your connected devices.

Pricing works on three main metrics:

(*a*) Data Exchanged

(*b*) Data Analyzed

(*c*) Edge Data Analyzed

You'll get 100 MB of each for free every month, IBM Watson also offers some cool security possibilities based on machine learning and data science.

4. Google Cloud Platform

Google Cloud IoT is a complete set of tools to connect, process, store, and analyze data both at the edge and in the cloud. The platform consists of scalable, fully-managed cloud services; an integrated software stack for edge/on-premises computing with machine learning capabilities for all your IoT needs. They claim that "Cloud Platform is the best place to build IoT initiatives, taking advantage of Google's heritage of web-scale processing, analytics, and machine intelligence".

Their focus is on making things easy and fast for your business, where instant information is expected. And, offer "Google grade" security. Using this platform also lets you take advantage of Google's private global fiber network. Pricing on Google Cloud is done on a per-minute basis. It is usually cheaper than Amazon Web Services and even has a price comparison tool to show you how much you'll save. But doesn't have the same extensive tools and documentation. Like Microsoft, Google also has its own IoT operating system (based on Android).

5. **Oracle.** Oracle is a platform as a service provider that seems to be focusing on manufacturing and logistics operations. They want to help you get your products to market faster.

Pricing for Oracle is calculated per device. There is a set number of messages per device, per month, with an additional cost if you go over this number.

Apart from above the following are next five more popular service providers.

6. Sales force
7. Bosch
8. Cisco IoT Cloud Connect
9. General Electrics Predix
10. SAP

Pricing is easy to understand, with 3 tiers for

- Developers
- Medium Business
- Enterprise

1.9. FOG COMPUTING

Fog computing or fog networking, also known as fogging, is an architecture that uses edge devices to carry out a substantial amount of *computation, storage, communication locally* and *routed over the internet backbone.*

An **edge device** is a device which provides an entry point into enterprise or service provider core networks. Examples include *routers, routing switches, integrated access devices (IADs), multiplexers,* and *a variety of metropolitan area network (MAN)* and *Wide area network (WAN) access devices.* In addition an edge device is a type of networking device that connects an internal local area network (LAN) with an external wide area network i.e. **Internet.** It provides interconnectivity and traffic translation between different networks on their entering edges or the network boundaries.

Fog computing is a concept of distributed computing paradigm that provides *data, compute, storage* and *application services* closer to client or near-user edge devices, such as network routers. Furthermore, fog computing handles data at the network level, on smart devices and on the end-user client side (e.g. mobile devices), instead of sending data to a remote location for processing.

Fog computing can be perceived both in large cloud systems and big data structures, making reference to the growing difficulties in accessing information objectively. This results in a lack of quality of the obtained content. The effects of fog computing on cloud computing and big data systems may vary. However, a common aspect is a limitation in accurate content distribution, an issue that has been tackled with the creation of metrics that attempt to improve accuracy.

Fog networking consists of a control plane and a data plane. For example, on the data plane, fog computing enables computing services to reside at the edge of the network as opposed to servers in a data-center. Compared to cloud computing, fog computing emphasizes proximity to end-users and client objectives, dense geographical distribution and local resource pooling, latency reduction and backbone bandwidth savings to achieve better quality of service (QoS) and edge analytics/stream mining, resulting in superior user-experience and redundancy in case of failure while it is also able to be used in Assisted Living scenarios.

Fog networking supports the Internet of Things (IoT) concept, in which most of the devices used by humans on a daily basis will be connected to each other. Examples include phones, wearable health monitoring devices, connected vehicle and augmented reality using devices such as the Google Glass.

1.10. IOT GATEWAYS REVISITED

A gateway is a hardware device that acts as a "gate" between two networks. It may be a *router, firewall, server,* or *other device* that enables traffic to flow in and out of the network. While a gateway protects the nodes within network, it also a node itself.

A gateway is a node (router) in a computer network (vide Fig. 1.15) a key stopping point for data on its way to or from other networks. Thanks to gateways, we are able to communicate and send data back and forth. The Internet wouldn't be any use to us without gateways (as well as a lot of other hardware and software).

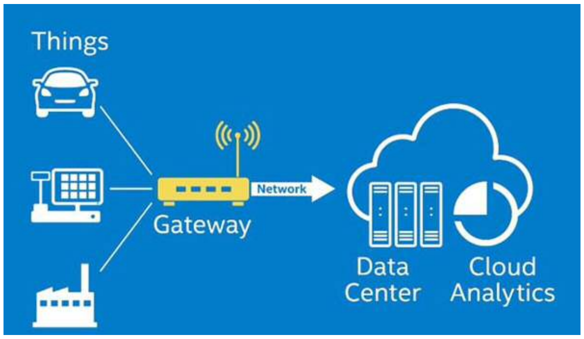


Fig. 1.15. IoT Gateway (Note: the graphic Nicolas used comes from this gateway page of Intel, one of the larger market players who states that IoT gateways need to enable scalability, manageability, advanced security, performance at the edge and faster, more flexible deployment).

Gateways regulate traffic between two dissimilar networks, while routers regulate traffic between similar networks. The easiest way to illustrate this point is through an example. Because TCP/IP is also the primary protocol of the Internet, you could use a router to connect your network to the Internet.

The gateway acts as a bridge between these IOT objects and the internet. Gateways can connect to the IoT devices that communicate via specific protocols, store and parse the information and then send them over to cloud servers for processing and analytics. IOT gateways not only abstract the medium of communication but also provide the secure channel required for the transmission of this data. Gateways usually run real-time operation systems (RTOS) or a form of Linux to drive their systems. Hardware and software level encryption is built right into the gateway to provide a secure channel for communication.

The definition of an IoT gateway has changed over time as the market developed. Just like traditional gateways in networks do, IoT gateways function like bridges – and they bridge a lot.

According to Nicolas Windpassinger the writer of book "Digitize or Die"- , a high level of *interoperability, redundancy, connectivity, pre-processing of data, aggregation of data, remote control* and *management* leads to the requirement for gateways.

The essential definition of an IoT gateway used to revolve around the bridging of the things of the Internet of Things on one hand and the network (routers, base stations and so forth), cloud and/or data center infrastructure on the other.

In an increasingly complex IoT reality with more large scale IoT projects, loads of IoT communication protocols, industrial and IoT standards and many types of sensors and sensor data sitting a bit everywhere (all depending on the scope on the project) a device was and is needed to:

1. bridge it all and enable those various things and sensors and data to 'talk' with each other (despite speaking a different language) and
2. making some sense of it all before sending all this data somewhere else where it really – ideally – leads to real sense, actionable insights and actions in whatever shape.

Today the **IoT gateway** is often defined as the bridge between the "full" edge (including the edge systems) and its various components on one hand and the cloud (and business applications or whatever infrastructure where the data goes to/through) on the other.

In order to make the convergence of Information Technology (IT) and Operational Technology (OT) a reality in any given project (and beyond the human and other factors) you need IoT gateways and IoT platforms. Operational Technology (OT) includes the hardware and software dedicated to detecting or causing changes in physical processes through direct monitoring and/or control of physical devices such as valves, pumps, etc. Gateway design is discussed in para. 5.5.5 of this book.

1.11. CLOUD BASED DATA PREPROCESSING

As smart things collect huge amount of sensor data, compute and storage resources are required to analyze, store, and process this data. The most common compute and storage resources are cloud based because the cloud offers massive data handling, scalability, and flexibility. But this will not be sufficient to meet the requirements of many IoT applications because of the following reasons.

1. **Mobility.** Most of the smart devices are mobile. Their changing location makes it difficult to communicate with the cloud data center because of changing network conditions across different locations.
2. **Reliable and real time actuation.** Communicating with the cloud and getting back responses takes time. Latency sensitive applications, which need real time responses, may not be feasible with this model. Also, the communication may be lossy due to wireless links, which can lead to unreliable data.
3. **Scalability.** More devices means more requests to the cloud, thereby increasing the latency.
4. **Power constraints.** Communication consumes a lot of power, and IoT devices are battery powered. They thus cannot afford to communicate all the time.

1.12. FOG (NEAR EDGE) COMPUTING AND SMART GATEWAYS

To solve the problem of mobility, researchers have proposed **Mobile Cloud Computing** (MCC) . But there are still problems associated with latency and power. MCC also suffers from mobility problems such as frequently changing network conditions due to which problems such as signal fading and service degradation arise.

As a solution to these problems, we can bring some compute and storage resources to the edge of the network instead of relying on the cloud for everything. This concept is known as **fog computing** . The fog can be viewed as a cloud, which is close to the ground. Data can be stored, processed, filtered, and analyzed on the edge of the network before sending it to the cloud through expensive communication media. The fog and cloud paradigms go together. Both of them are required for the optimal performance of IoT applications. A smart gateway can be employed between underlying networks and the cloud to realize fog computing as shown in Figure 1.16.

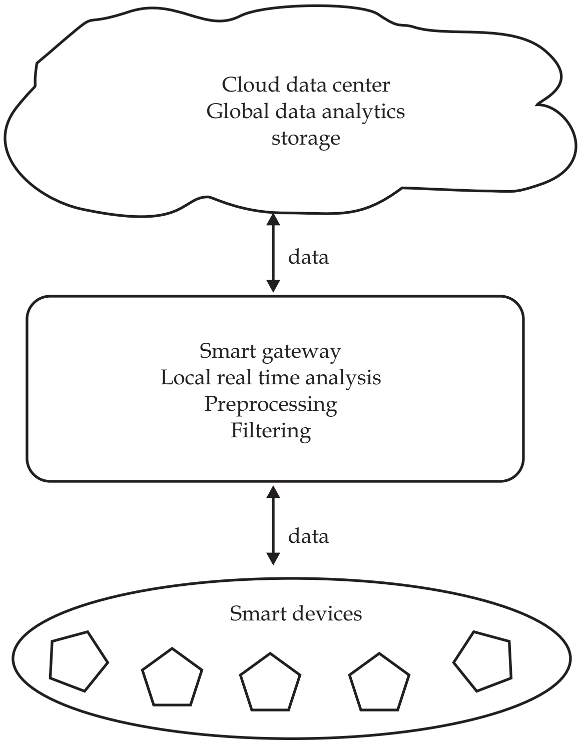


Fig. 1.16. Smart gateway for preprocessing.

1.12.1. Features of Fog Computing

The features of fog computing are as follows:

1. **Low latency.** less time is required to access computing and storage resources on fog nodes (smart gateways).
2. **Location awareness**. as the fog is located on the edge of the network, it is aware of the location of the applications and their context. This is beneficial as context awareness is an important feature of IoT applications.
3. **Distributed nodes**: fog nodes are distributed unlike centralized cloud nodes. Multiple fog nodes need to be deployed in distributed geographical areas in order to provide services to mobile devices in those areas. For example, in vehicular networks, deploying fog nodes at highways can provide low latency data/video streaming to vehicles.
4. **Mobility**: the fog supports mobility as smart devices can directly communicate with smart gateways present in their proximity.
5. **Real time response**: fog nodes can give an immediate response unlike the cloud, which has a much greater latency.
6. I**nteraction with the cloud**: fog nodes can further interact with the cloud and communicate only that data, which is required to be sent to the cloud.

The tasks performed by a smart gateway [20] are collecting sensor data, preprocessing and filtering collected data, providing compute, storage and networking services to IoT devices, communicating with the cloud and sending only necessary data, monitoring power consumption of IoT devices, monitoring activities and services of IoT devices, and ensuring security and privacy of data.

1.12.2. Applications of Fog Computing

Some applications of fog computing are as follows :

1. **Smart vehicular networks**: smart traffic lights are deployed as smart gateways to locally detect pedestrians and vehicles through sensors, calculate their distance and speed, and finally infer traffic conditions. This is used to warn oncoming vehicles. These sensors also interact with neighboring smart traffic lights to perform traffic management tasks. For example, if sensors detect an approaching ambulance, they can change the traffic lights to let the ambulance pass first and also inform other lights to do so. The data collected by these smart traffic lights are locally analyzed in real time to serve real time needs of traffic management. Further, data from multiple gateways is combined and sent to the cloud for further global analysis of traffic in the city.
2. **Smart grid**: the smart electrical grid facilitates load balancing of energy on the basis of usage and availability. This is done in order to switch automatically to alternative sources of energy such as solar and wind power. This balancing can be done at the edge of the network using smart meters or micro-grids connected by smart gateways. These gateways can analyze and process data. They can then project future energy demand, calculate the availability and price of power, and supply power from both conventional and alternative sources to consumers.

1.13. SERVICE MODELS IN CLOUD COMPUTING

Though service-oriented architecture advocates Everything as a service (with the acronyms **EaaS** or **XaaS**, [3] or simply **aas**), cloud-computing providers offer their "services" according to different models, of which the three standard models per NIST are

1. Infrastructure as a Service (IaaS),
2. Platform as a Service (PaaS), and
3. Software as a Service (SaaS).

These models offer increasing abstraction; they are thus often portrayed as a *layers* in a stack: *infrastructure-, platform-* and *software-as-a-service*, but these need not be related. For example, one can provide SaaS implemented on physical machines (bare metal), without using underlying PaaS or IaaS layers, and conversely one can run a program on IaaS and access it directly, without wrapping it as SaaS.

1.14. EVERYTHING-AS-A-SERVICE (XAAS)

Also known as anything-as-a-service, it facilities the flexibility for users and companies to customize their computing environments to craft the experiences they desire, all on demand. XaaS is dependent on a strong cloud services platform and reliable Internet connectivity to successfully gain traction and acceptance among both individuals and enterprises.

XaaS is a cloud computing term for the extensive variety of services and applications emerging for users to access on demand over the Internet as opposed to being utilized via on-premises means.

XaaS originated as software-as-a-service (SaaS) and has since expanded to include services such as infrastructure-as-a-service, platform-as-a-service, storage-as-a-service, desktop-as-a-service, disaster recovery-as-a-service, and even nascent operations like marketing-as-a-service and healthcare-as-a-service.

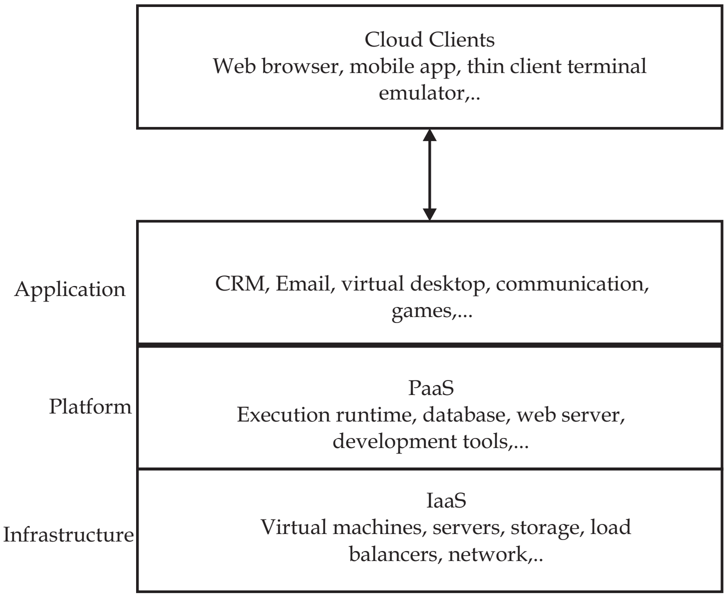

Cloud Clients
Web browser, mobile app, thin client terminal emulator,..
Application
CRM, Email, virtual desktop, communication, games,...
Platform
PaaS
Execution runtime, database, web server, development tools,...
Infrastructure
IaaS
Virtual machines, servers, storage, load balancers, network,..

Fig. 1.17. Cloud computing service models arranged as layers in a stack

We have already learnt about monetisation as Service in para 1.7 which is a part of every thing as service.

1.15. INFRASTRUCTURE AS A SERVICE (IAAS)

IaaS or is basically a virtual provision of computing resources over the cloud. An IaaS cloud provider can give you the entire range of computing infrastructures such as:

1. storage,
2. servers,
3. high level APIs
4. networking hardware alongside
5. maintenance and safety support.

IaaS refers to online services that provide high-level APIs used to de-reference various low-level details of underlying network infrastructure like physical computing resources, location, data partitioning, scaling, security, backup etc. Businesses can opt for computing resources of their requirement without the need to install hardware on their premises. *Amazon Web Services, Microsoft Azure,* and *Google Compute Engine* are some of the leading IaaS cloud service providers.

A hyper visor runs the virtual machines as guests. Pools of hypervisors within the cloud operational system can support large numbers of virtual machines and the ability to scale services up and down according to customers varying requirements. Linux containers run in isolated partitions of a single Linux kernel running directly on the physical hardware. Linux cgroups and name spaces are the underlying Linux kernel technologies used to isolate, secure and manage the containers. Containerisation offers higher performance than virtualization, because there is no hypervisor overhead. Also, container capacity auto-scales dynamically with computing load, which eliminates the problem of over-provisioning and enables usage-based billing.[64] IaaS clouds often offer additional resources such as a virtual-machine disk-image library, raw block storage, file or object storage, firewalls, load balancers, IP addresses, virtual local area networks (VLANs), and software bundles.

The NIST's definition of cloud computing describes IaaS as "where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications; and possibly limited control of select networking components (e.g., host firewalls)."

IaaS-cloud providers supply these resources on-demand from their large pools of equipment installed in data centers. For wide-area connectivity, customers can use either the Internet or carrier clouds (dedicated virtual private networks). To deploy their applications, cloud users install operating-system images and their application software on the cloud infrastructure. In this model, the cloud user patches and maintains the operating systems and the application software. Cloud providers typically bill IaaS services on a utility computing basis: cost reflects the amount of resources allocated and consumed.

1.15.1. Benefits of using IaaS platform

- **Minimize Costs.** Deploying an IaaS cloud model eliminates the need to deploy on-premise hardware that reduces the costs.
- **Enhanced Scalability.** As the most flexible cloud computing model, IaaS allows you to scale the computing resources up or down based on demand.
- **Simple Deployment.** IaaS lets you easily deploy the servers, processing, storage, and networking to make it up and running in no time.

1.15.2. Why Should One Opt IaaS?

IaaS being the most flexible of cloud models gives the best option when it comes to IT hardware infrastructure. IaaS is the right option if you need control over the hardware infrastructure such as in managing and customizing according to your requirements.

Whether you are running a startup or a large enterprise, IaaS gives access to computing resources without the need to invest in them separately. However, the only downside with IaaS is that it is much costlier than SaaS or PaaS cloud models.

1.16. PLATFORM AS A SERVICE (PAAS)

The NIST's definition of cloud computing defines Platform as a Service as:

The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages, libraries, services, and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment.

PaaS vendors offer a development environment to application developers. The provider typically develops toolkit and standards for development and channels for distribution and payment. In the PaaS models, cloud providers deliver a computing platform, typically including operating system, programming-language execution environment, database, and web server. Application developers develop and run their software on a cloud platform instead of directly buying and managing the underlying hardware and software layers. With some PaaS, the underlying computer and storage resources scale automatically to match application demand so that the cloud user does not have to allocate resources manually.

Some integration and data management providers also use specialized applications of PaaS as delivery models for data. Examples include **iPaaS (Integration Platform as a Service)** and **dPaaS (Data Platform as a Service)**. iPaaS enables customers to develop, execute and govern integration flows. Under the iPaaS integration model, customers drive the development and deployment of integrations without installing or managing any hardware or middleware. dPaaS delivers integration—and data-management—products as a fully managed service. Under the dPaaS model, the PaaS provider, not the customer, manages the development and execution of programs by building data applications for the customer. dPaaS users access data through data-visualization tools. Platform as a Service (PaaS) consumers do not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but have control over the deployed applications and possibly configuration settings for the application-hosting environment.

1.16.1. Why Should One Opt PaaS?

PaaS is the preferred option if your project involves multiple developers and vendors. With PaaS, it is easy to create customized applications as it leases all the essential computing and networking resources. Being a different model, PaaS simplifies the app development process that minimizes your organizational costs.

Besides, it is flexible and delivers the necessary speed in the process, which will rapidly improve your development times. A typical disadvantage with PaaS is that since it is built on virtualized technology, you will have less control over the data processing. In addition, it is also less flexible compared to the IaaS cloud model.

1.16.2. Benefits of using PaaS Platform

- **Minimal Development Time.** PaaS reduces the development time since the vendor provides all computing resources like server-side components, which simplifies the process and improve the focus of the development team.
- **Multiple Programming Language Support.** PaaS offers support for multiple programming languages, which a software development company can utilize to build applications for different projects.
- **Enhanced Collaboration.** With PaaS, your business can benefit from having enhanced collaboration, which will help integrate your team dispersed across various locations.

1.17. SOFTWARE AS A SERVICE (SAAS)

SaaS is a model that gives quick access to cloud-based web applications. The vendor controls the entire computing stack, which you can access using a web browser. These applications run on the cloud and you can use them by a paid licensed subscription or for free with limited access.

SaaS does not require any installations or downloads in your existing computing infrastructure. This eliminates the need for installing applications on each of your computers with the maintenance and support taken over by the vendor. Some known example of SaaS includes Google G Suite, Microsoft Office 365, Dropbox etc.

The NIST's definition of cloud computing defines Software as a Service as:

The capability provided to the consumer is to use the provider's applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface.

The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.

In the software as a service (SaaS) model, users gain access to application software and databases. Cloud providers manage the infrastructure and platforms that run the applications. SaaS is sometimes referred to as "on-demand software" and is usually priced on a pay-per-use basis or using a subscription fee. In the SaaS model, cloud providers install and operate application software in the cloud and cloud users access the software from cloud clients. Cloud users do not manage the cloud infrastructure and platform where the application runs. This eliminates the need to install and run the application on the cloud user's own computers, which simplifies maintenance and support. Cloud applications differ from other applications in their scalability—which can be achieved by cloning tasks onto multiple virtual machines at run-time to meet changing work demand. Load balancers distribute the work over the set of virtual machines. This process is transparent to the cloud user, who sees only a single access-point. To accommodate a large number of cloud users, cloud applications can be *multitenant*, meaning that any machine may serve more than one cloud-user organization.

The pricing model for SaaS applications is typically a monthly or yearly flat fee per user, so prices become scalable and adjustable if users are added or removed at any point. It may also be free. Proponents claim that SaaS gives a business the potential to reduce IT operational costs by outsourcing hardware and software maintenance and support to the cloud provider. This enables the business to reallocate IT operations costs away from hardware/software spending and from personnel expenses, towards meeting other goals. In addition, with applications hosted centrally, updates can be released without the need for users to install new software. One drawback of SaaS comes with storing the users' data on the cloud provider's server. As a result, there could be unauthorized access to the data.

Examples of applications offered as SaaS are games and productivity software like Google Docs and Word Online. SaaS applications may be integrated with cloud storage or File hosting services, which is the case with Google Docs being integrated with Google Drive and Word Online being integrated with Onedrive.

1.17.1. Benefits of using SaaS

- **Affordable.** SaaS is affordable as it eliminates the costs involved in the purchase, installation, maintenance and upgrades of computing hardware.
- **Anywhere Accessibility.** With SaaS, you can access the services from anywhere using any device such as smart phones, which eliminates the constraints set by on-premise software.
- **Ready to Use** – You can quickly set up SaaS services so that they become functional in no time. All it takes is that you sign up for the service to get access to fast and powerful computing resources.

1.17.2. Why Should One Opt SaaS?

SaaS has its own disadvantages since it leaves you no control over the hardware allotted to you as only the vendor can manage the software. With SaaS, communication, transferring of content and scheduling meetings are made easy.

SaaS is the ideal choice for small-scale businesses who do not have the necessary budget and resources to deploy on-premise hardware. Besides, companies that require frequent collaboration on their projects will find SaaS platforms useful.

1.18. CHOOSING A PROPER CLOUD SERVICE MODELS

Growing adoption of cloud services is a sign of the rapidly changing business environment. The forecasts and reports shed light on how the cloud is going to become the primary computing resource for enterprises in times to come. So, that suggests that your business should quickly adopt a cloud platform to leverage its wide-reaching benefits and in turn help you grow.

But, what cloud model would be apt as a solution that delivers the results that you are looking for. The above-mentioned details about SaaS, IaaS and PaaS may have provided you with a peek into the nature of these cloud models. Each of them differs and it is up to you to address your business requirements and select one that you find apt for your needs.

Summarizing, SaaS would suit your business well if you need a cloud-based software like email, CRM, and productivity tools. IaaS is the perfect option if you require a complete virtual computing platform with powerful resources. If your requirement is a platform to develop and test your software and applications, then it is better to opt PaaS.

1.19. CHALLANGES TO IOT APPLICATIONS

At present IoT is faced with many challenges, such as:

1. Insufficient testing and updating
2. Concern regarding data Tecurity and privacy
3. Software complexity
4. Data volumes and interpretation
5. Integration with AI and automation
6. Devices require a constant power supply which is difficult
7. Interaction and short-range communication
8. Advantages of IoT

1.20. KEY BENEFITS OF IOT

Key benefits of IoT technology are as follows:

1. **Technical Optimization.** IoT technology helps a lot in improving technologies and making them better. Example, with IoT, a manufacturer is able to collect data from various car sensors. The manufacturer analyzes them to improve its design and make them more efficient.
2. **Improved Data Collection.** Traditional data collection has its limitations and its design for passive use. IoT facilitates immediate action on data.
3. **Reduced Waste.** IoT offers real-time information leading to effective decision making & management of resources. For example, if a manufacturer finds an issue in multiple car engines, he can track the manufacturing plan of those engines and solves this issue with the manufacturing belt.
4. **Improved Customer Engagement.** IoT allows you to improve customer experience by detecting problems and improving the process.

Disadvantages IOT

1. **Security.** IoT technology creates an ecosystem of connected devices. However, during this process, the system may offer little authentication control despite sufficient security measures.
2. **Privacy.** The use of IOT, exposes a substantial amount of personal data, in extreme detail, without the user's active participation. This creates lots of privacy issues.
3. **Flexibility.** There is a huge concern regarding the flexibility of an IoT system. It is mainly regarding integrating with another system as there are many diverse systems involved in the process.

4. **Complexity.** The design of the IOT system is also quite complicated. Moreover, it's deployment and maintenance also not very easy.

5. **Compliance.** IOT has its own set of rules and regulations. However, because of its complexity, the task of compliance is quite challenging. Internet of Things (IOT) Sectors

1.21. IOT DEVELOPMENT PHASES

The growth of the IOT is expected to go through several stages of development:

1. *Passive*– RFID sensors etc –
2. *Active*– Responds to sensor data
3. *Aware*– can make choices based on data.
4. *Autonomous*– e.g. self driving cars

We are currently in the early stages of development (Passive phase) were we are receiving data from objects and manually taking action.

1.22. SECURITY ASPECTS IN IOT

1.22.1. User Management

Alongside with device management, it's important to provide control over the users having access to an IoT system.

User management involves identifying users, their roles, access levels and ownership in a system. It includes such options as adding and removing users, managing user settings, controlling access of various users to certain information, as well as the permission to perform certain operations within a system, controlling and recording user activities and more.

1.22.2. Security Monitoring

Security is one of the top concerns in the Internet of things. Connected things produce huge volumes of data, which need to be securely transmitted and protected from cyber-criminals. Another side is that the things connected to the Internet can be entry points for villains. What is more, cyber-criminals can get the access to the "brain" of the whole IoT system and take control of it.

To prevent such problems, it makes sense to log and analyze the commands sent by control applications to things, monitor the actions of users and store all these data in the cloud. With such an approach, it's possible to address security breaches at the earlies stages and take measures to reduce their influence on an IoT system (for example, block certain commands coming from control applications).

Also, it's possible to identify the patterns of suspicious behavior, store these samples and compare them with the logs generated by an IoT systems to prevent potential penetrations and minimize their impact on an IoT system.

Due to the interconnectivity of the IoT, a cyber incident could result in an information breach which affects multiple levels of your business, from the head office, to your customers, and to the supply chain in between. Whether targeted or indirect, cyber incidents could weaken your entire IT security infrastructure. An information breach could cause a loss of revenue and time, could damage your business's reputation and credibility, and could lead to legal challenges. To protect the information on your network, you should control who and what connects to it.

1.22.3. Privacy

With every connected device comes some vulnerability. IoT-related cyber incidents can put business, employee, and client information at risk of being destroyed, altered, stolen and exposed, or even held for ransom. Another concern with IoT data collection is over the confidentiality, privacy and integrity of business data. It is important to understand the data collection and privacy policies of IoT devices, before you buy or download them. You should find information on the device's website about what data is gathered by, how long it is kept, and what your data is used for, such as marketing research. Educate and train your employees on the potential risks of connecting their personal IoT devices to the business network. Make sure staff are adhering to your Bring Your Own Device policy, and consider updating it with the introduction of IoT into the workplace.

1.22.4. Safety

For many devices, their operational roles can be far more important than the data they store. Consider the legal and financial impact of a device like a smart vehicle or insulin pump failing or being manipulated. The malfunction or unauthorized control of an IoT device could cause damage to data and equipment, or physical harm to staff, customers or the public. Cyber attacks can be costly, as you seek to recover your systems, information, and your company's reputation.

1.22.5 Security Considerations

1. **Trustworthy and secure communication**

All information received from and sent to a device must be trustworthy. Unless a device can support the following cryptographic capabilities, it should be constrained to local networks and all internetwork communication should go through a field gateway:

- Data encryption with a provably secure, publicly analyzed, and broadly implemented symmetric-key encryption algorithm.
- Digital signature with a provably secure, publicly analyzed, and broadly implemented symmetric-key signature algorithm.
- Support for either TLS 1.2 for TCP or other stream-based communication paths or DTLS 1.2 for datagram-based communication paths. Support of X.509 certificate handling is optional and can be replaced by the more compute-efficient and wire-efficient pre-shared key mode for TLS, which can be implemented with support for the AES and SHA-2 algorithms.
- Updateable key-store and per-device keys. Each device must have unique key material or tokens that identify it toward the system. The devices should store the key securely on the device (for example, using a secure key-store). The device should be able to update the keys or tokens periodically, or reactively in emergency situations such as a system breach.
- The firmware and application software on the device must allow for updates to enable the repair of discovered security vulnerabilities.

However, many devices are too constrained to support these requirements. In that case, a field gateway should be used. Devices connect securely to the field gateway through a local area network, and the gateway enables secure communication to the cloud.

2. **Physical taper-proofing**

It is strongly recommended that device design incorporates features that defend against physical manipulation attempts, to help ensure the security integrity and trustworthiness of the overall system.

For example:

- Choose microcontrollers/microprocessors or auxiliary hardware that provide secure storage and use of cryptographic key material, such as trusted platform module (TPM) integration.
- Secure boot loader and secure software loading, anchored in the TPM.
- Use sensors to detect intrusion attempts and attempts to manipulate the device environment with alerting and potentially "digital self-destruction" of the device.

3. **Monitoring and logging**

Logging and monitoring systems are used to determine whether the solution is functioning and to help troubleshoot problems. Monitoring and logging systems help answer the following operational questions:

- Are devices or systems in an error condition?
- Are devices or systems correctly configured?
- Are devices or systems generating accurate data?
- Are systems meeting the expectations of both the business and end customers?

Logging and monitoring tools are typically comprised of the following four components:

- System performance and timeline visualization tools to monitor the system and for basic troubleshooting.
- Buffered data ingestion, to buffer log data.
- Persistence store to store log data.
- Search and query capabilities, to view log data for use in detailed troubleshooting.

Monitoring systems provide insights into the health, security, and stability, and performance of an IoT solution. These systems can also provide a more detailed view, recording component configuration changes and providing extracted logging data that can surface potential security vulnerabilities, enhance the incident management process, and help the owner of the system troubleshoot problems. Comprehensive monitoring solutions include the ability to query information for specific subsystems or aggregating across multiple subsystems.

Monitoring system development should begin by defining healthy operation, regulatory compliance, and audit requirements. Metrics collected may include:

- Physical devices, edge devices, and infrastructure components reporting configuration changes.
- Applications reporting configuration changes, security audit logs, request rates, response times, error rates, and garbage collection statistics for managed languages.
- Databases, persistence stores, and caches reporting query and write performance, schema changes, security audit log, locks or deadlocks, index performance, CPU, memory, and disk usage.
- Managed services (IaaS, PaaS, SaaS, and FaaS) reporting health metrics and configuration changes that impact dependent system health and performance.

Visualization of monitoring metrics alert operators to system instabilities and facilitate incident response.

1.23. HISTORY OF IOT

The history of the IIoT begins with the invention of the programmable logic controller (PLC) by Dick Morley in 1968, which was used by General Motors in their automatic transmission manufacturing division. These PLCs allowed for fine control of individual elements in the manufacturing chain. In 1975, Honeywell and Tokugawa introduced the world's first DCSs, the TDC 2000 and the CENTUM system, respectively. These DCSs were the next step in allowing flexible process control throughout a plant, with the added benefit of backup redundancies by distributing control across the entire system, eliminating a singular point of failure in a central control room.

With the introduction of Ethernet in 1980, people began to explore the concept of a network of smart devices as early as 1982, when a modified Coke machine at Carnegie Mellon University became the first internet-connected appliance, able to report its inventory and whether newly loaded drinks were cold. As early as in 1994, greater industrial applications were envisioned, as Reza Raji described the concept in IEEE Spectrum as "[moving] small packets of data to a large set of nodes, so as to integrate and automate everything from home appliances to entire factories".

The concept of the internet of things first became popular in 1999, through the Auto-ID Center at MIT and related market-analysis publications.[18] Radio-frequency identification (RFID) was seen by Kevin Ashton (one of the founders of the original Auto-ID Center) as a prerequisite for the internet of things at that point. If all objects and people in daily life were equipped with identifiers, computers could manage and inventory them. Besides using RFID, the tagging of things may be achieved through such technologies as near field communication, barcodes, QR codes and digital watermarking.

The current conception of the IIoT arose after the emergence of cloud technology in 2002, which allows for the storage of data to examine for historical trends, and the development of the OPC Unified Architecture protocol in 2006, which enabled secure, remote communications between devices, programs, and data sources without the need for human intervention or interfaces.

One of the first consequences of implementing the industrial internet of things (by equipping objects with minuscule identifying devices or machine-readable identifiers) would be to create instant and ceaseless inventory control. Another benefit of implementing an IIoT system is the ability to create a digital twin of the system. Utilizing this digital twin allows for further optimization of the system by allowing for experimentation with new data from the cloud without having to halt production or sacrifice safety, as the new processes can be refined virtually until they are ready to be implemented. A digital twin can also serve as a training ground for new employees who won't have to worry about real impacts on the live system.

Milestones in history of development of IOT

1. **1970.** The actual idea of connected devices was proposed
2. **1990.** John Romkey created a toaster which could be turned on/off over the Internet
3. **1995.** Siemens introduced the first cellular module built for M2M
4. **1999.** The term "Internet of Things" was used by Kevin Ashton during his work at P&G which became widely accepted
5. **2004.** The term was mentioned in famous publications like the Guardian, Boston Globe, and Scientific American
6. **2005.** UN's International Telecommunications Union (ITU) published its first report on this topic.
7. **2008.** The Internet of Things was born
8. **2011.** Gartner, the market research company, include "The Internet of Things" technology in their research

IoT − Disadvantages

1. Though– IoT delivers an impressive set of benefits, it also presents a significant set of challenges. Here is a list of some its major issues −

2. Security− IoT creates an ecosystem of constantly connected devices communicating over networks. The system offers little control despite any security measures. This leaves users exposed to various kinds of attackers.

3. Privacy− The sophistication of IoT provides substantial personal data in extreme detail without the user's active participation.

4. Complexity− Some find IoT systems complicated in terms of design, deployment, and maintenance given their use of multiple technologies and a large set of new enabling technologies.

5. Flexibility− Many are concerned about the flexibility of an IoT system to integrate easily with another. They worry about finding themselves with several conflicting or locked systems.

6. Compliance− IoT, like any other technology in the realm of business, must comply with regulations. Its complexity makes the issue of compliance seem incredibly challenging when many consider standard software compliance a battle.

References

1. *"What is Cloud Computing?". Amazon Web Services. 2013-03-19. Retrieved* 2013-03-20.
2. *"Gartner Says Cloud Computing Will Be As Influential As E-business". Gartner. Retrieved* 2010-08-22.
3. Duan, Yucong; Fu, Guohua; Zhou, Nianjun; Sun, Xiaobing; Narendra, Nanjangud; Hu, Bo (2015). "Everything as a Service (XaaS) on the Cloud: Origins, Current and Future Trends". *2015 IEEE 8th International Conference on Cloud Computing*. IEEE. pp. 621–628. doi:10.1109/CLOUD.2015.88. ISBN 978-1-4673-7287-9.
4. Mills, Elinor (2009-01-27). "Cloud computing security forecast: Clear skies". CNET News. Retrieved 2019-09-19.