**Module 1**

Introduction to IoT

## What is IoT?

The Internet of Things (IoT) is the network of physical objects or "things" embedded with electronics, software, sensors, and network connectivity, which enables these objects to collect and exchange data.

IoT allows objects to be sensed and controlled remotely across existing network infrastructure, creating opportunities for more direct integration between the physical world and computer-based systems, and resulting in improved efficiency, accuracy and economic benefit. "Things," in the IoT sense, can refer to a wide variety of devices such as heart monitoring implants, biochip transponders on farm animals, electric clams in coastal waters, automobiles with built-in sensors, DNA analysis devices for environmental/food/pathogen monitoring or field operation devices that assist fire-fighters in search and rescue operations.

These devices collect useful data with the help of various existing technologies and then autonomously flow the data between other devices.

## Genesis of IoT

The age of IoT is often said to have started between the years 2008 and 2009. During this time period, the number of devices connected to the Internet eclipsed the world’s population. With more “things” connected to the Internet than people in the world, a new age was upon us, and the Internet of Things was born.

Kevin Ashton is the person who created the term “Internet of Things”. Kevin has subsequently explained that IoT now involves the addition of senses to computers. In the Twentieth century, Computers depended on humans to input data and knowledge through typing, bar codes, and so on.

In the Twenty-first century, Computers are sensing things for themselves. The evolution of Internet can be categorized into four phases as shown below Figure 1.1

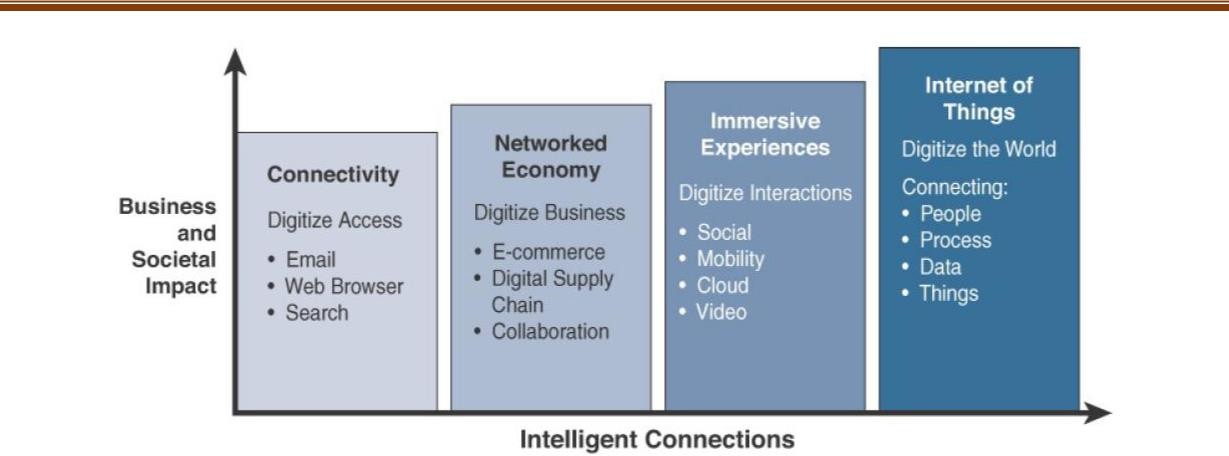
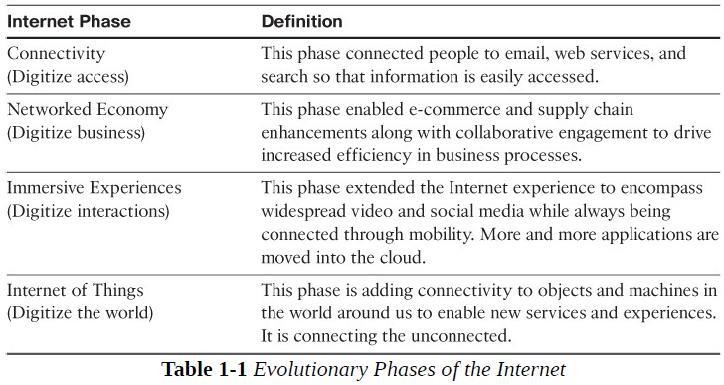


Figure 1.1 *Evolutionary Phases of the Internet*

The latest phase is the Internet of Things. Despite all the talk and media coverage of IoT, in many ways we are just at the beginning of this phase. When you think about the fact that 99% of “things” are still unconnected, you can better understand what this evolutionary phase is all about. Machines and objects in this phase connect with other machines and objects, along with humans.

Business and society have already started down this path and are experiencing huge increases in data and knowledge. In turn, this is now leading to previously unrecognized insights, along with increased automation and new process efficiencies. IoT is poised to change our world in new and exciting ways, just as the past Internet phases already have.



## IoT and Digitization

IoT focuses on connecting “things”, such as objects and machines, to a computer network, such as the Internet. IoT is a well-understood term used across the industry as a whole.

*For example:*

In a shopping mall where Wi-Fi location tracking has been deployed, the “things” are the Wi- Fi devices. Wi-Fi location tracking is simply the capability of knowing where a consumer is in a retail environment through his or her smart phone’s connection to the retailer’s Wi-Fi network.

While the value of connecting Wi-Fi devices or “things” to the Internet is obvious and appreciated by shoppers, tracking real-time location of Wi-Fi clients provides a specific benefit to the mall and shop owners.

On the other hand, Digitization can mean different things to different people but generally encompasses the connection of “things” with the data they generate and the business insights that result. Digitization is the conversion of information into a digital format.

*For example:*

The whole photography industry has been digitized. Pretty much everyone has digital cameras these days, either standalone devices or built into their mobile phones. Almost no one buys film and takes it to a retailer to get it developed. The digitization of photography has completely changed our experience when it comes to capturing images.

Example: Video rental industry, Transportation industry

##### IoT Impact

The following examples illustrate some of the benefits of IoT and their impact.

* + 1. Connected Roadways
    2. Connected Factory
    3. Smart Connected Buildings
    4. Smart Creatures

##### Connected Roadways

IoT is going to allow self-driving vehicles to better interact with the transportation system around them through bidirectional data exchanges while also providing important data to the riders. Connected roadways is the term associated with both the driver and driverless cars fully integrating with the surrounding transportation infrastructure



**Figure 1-2** *Google’s Self-Driving Car*

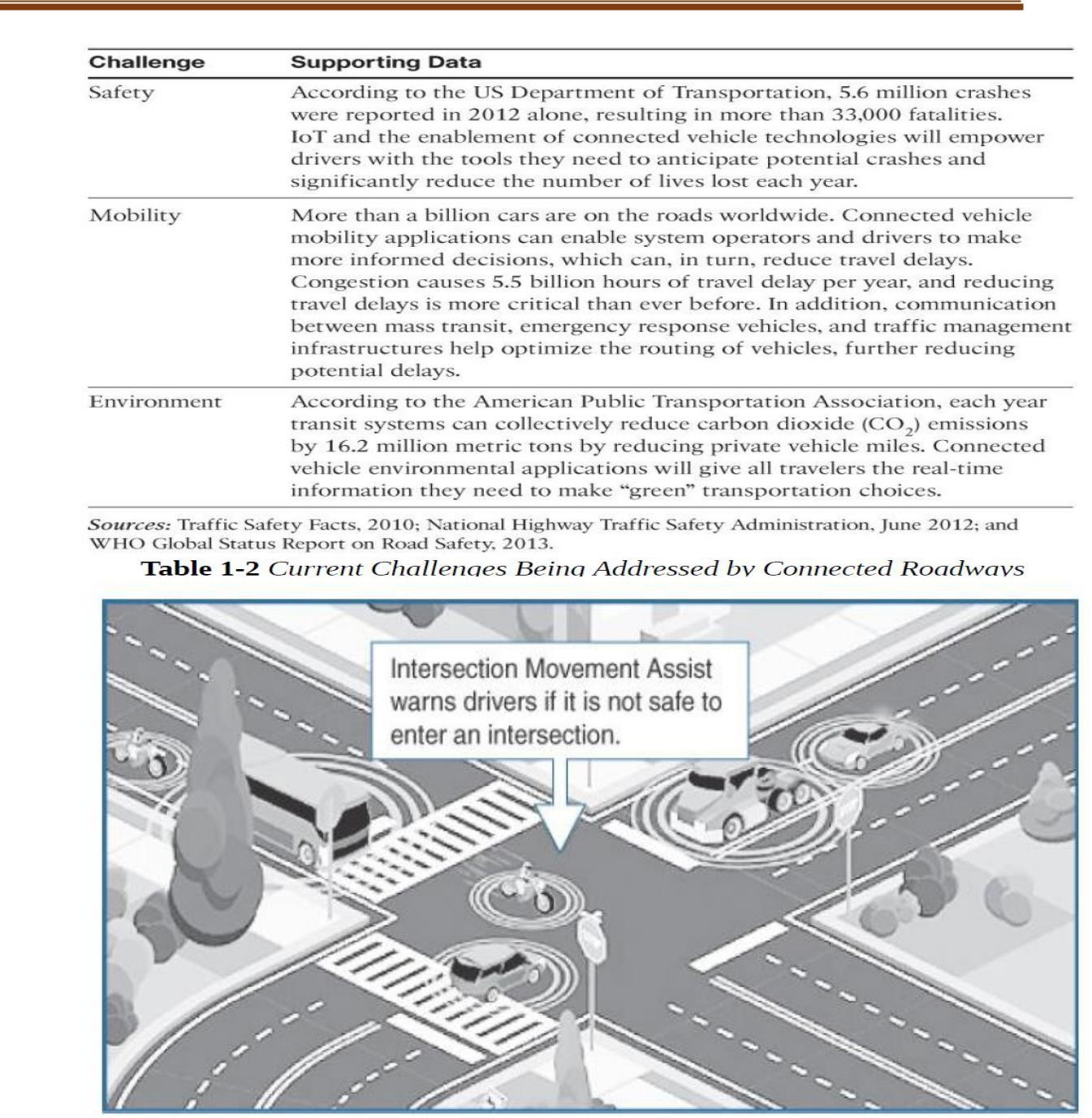
Most connected roadways solutions focus on resolving today’s transportation challenges such as

* 1. Safety 2. Mobility 3. Environment

By addressing the challenges in Table 1.2. Connected roadways will bring many benefits to society. These benefits include reduced traffic jams and urban congestion, decreased casualties and fatalities, increased response time for emergency vehicles, and reduced vehicle emissions.

For example:

With IoT-connected roadways, a concept known as Intersection Movement Assist (IMA) is possible. This application warns a driver (or triggers the appropriate response in a self-driving car) when it is not safe to enter an intersection due to a high probability of a collision—perhaps because another car has run a stop sign or strayed into the wrong lane. Thanks to the communications system between the vehicles and the infrastructure, this sort of scenario can be handled quickly and safely. Figure 1.4 for a graphical representation of IMA.



**Figure 1.3** Application of Intersection Movement Assist

##### Connected Factory

For years, traditional factories have been operating at a disadvantage, impeded by production environments that are “disconnected” or, at the very least, “strictly gated” to corporate business systems, supply chains, and customers and partners. Managers of these traditional factories are essentially “flying blind” and lack visibility into their

operations. These operations are composed of plant floors, front officers, and suppliers operating in independent silos.

Consequently, rectifying downtime issues, quality problems, and the root causes of various manufacturing inefficiencies is often difficult.

The main challenges facing manufacturing in a factory environment today include the following:



Accelerating new product and service introductions to meet customer and market opportunities

# 

Increasing plant production, quality, and uptime while decreasing cost

* Mitigating unplanned downtime
* Securing factories from cyber threats
* Decreasing high cabling and re-cabling costs
* Improving worker productivity and safety

For example 1:

Executive management is looking for new ways to manufacture in a more cost-effective manner while balancing the rising energy and material costs. Product development has time to market as the top priority. Plant managers are entirely focused on gains in plant efficiency and operational agility. The controls and automation department looks after the plant networks, controls, and applications and therefore requires complete visibility into all these systems. Industrial enterprises around the world are retooling their factories with advanced technologies and architectures to resolve these problems and boost manufacturing flexibility and speed. These improvements help them achieve new levels of overall equipment effectiveness, supply chain responsiveness, and customer satisfaction. A convergence of factory-based operational technologies and architectures with global IT networks is starting to occur, and this is referred to as the *connected factory*.

For example 2:

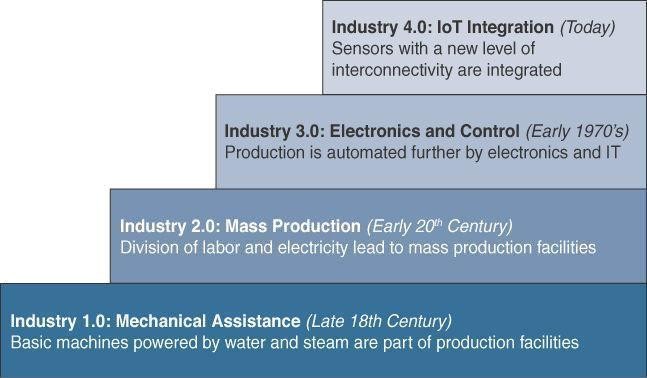
A smelting facility extracts metals from their ores. The facility uses both heat and chemicals to decompose the ore, leaving behind the base metal. This is a multistage process, and the data and controls are all accessed via various control rooms in a facility. Operators must go to a control room that is often hundreds of meters away for data and production changes. Hours of operator time are often lost to the multiple trips to the control room needed during a shift. With IoT and a connected factory solution, true “machine-to-people” connections are implemented to bring

sensor data directly to operators on the floor via mobile devices. Time is no longer wasted moving back and forth between the control rooms and the plant floor.

In addition, because the operators now receive data in real time, decisions can be made immediately to improve production and fix any quality problems.

##### Another example 3:

Connected factory solution involves a real-time location system (RTLS). An RTLS utilizes small and easily deployed Wi-Fi RFID tags that attach to virtually any material and provide real-time location and status. These tags enable a facility to track production as it happens. These IoT sensors allow components and materials on an assembly line to “talk” to the network. If each assembly line’s output is tracked in real time, decisions can be made to speed up or slow production to meet targets, and it is easy to determine how quickly employees are completing the various stages of production. Bottlenecks at any point in production and quality problems are also quickly identified. While we tend to look at IoT as an evolution of the Internet, it is also sparking an evolution of industry. In 2016 the World Economic Forum referred to the evolution of the Internet and the impact of IoT as the “fourth Industrial



**Figure 1.6** The Four Industrial Revolutions

##### Smart Connected Buildings

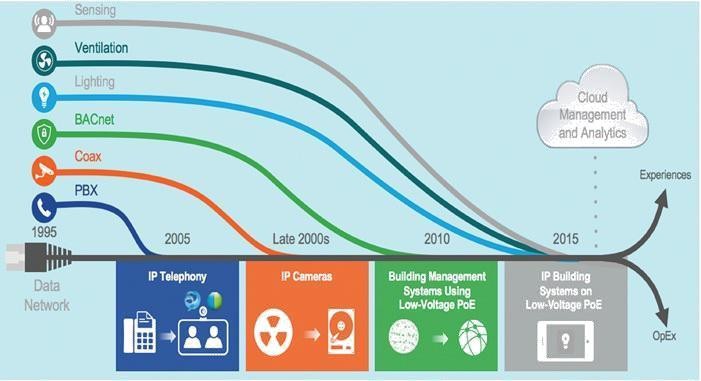
The function of a building is to provide a work environment that keeps the workers comfortable, efficient, and safe. Work areas need to be well lit and kept at a comfortable temperature. To keep workers safe, the fire alarm and suppression system needs to be carefully managed, as do the door and physical security alarm systems. While intelligent systems for modern buildings are being deployed and improved for each of these functions. Sensors are often used to control the heating, ventilation, and air-conditioning (HVAC)

system. Temperature sensors are spread throughout the building and are used to influence the building management system’s (BMS’s) control of air flow into a room.

Another interesting aspect of the smart building is that it makes them easier and cheaper to manage. Considering the massive costs involved in operating such complex structures, not to mention how many people spend their working lives inside a building, managers have become increasingly interested in ways to make buildings more efficient and cheaper to manage.

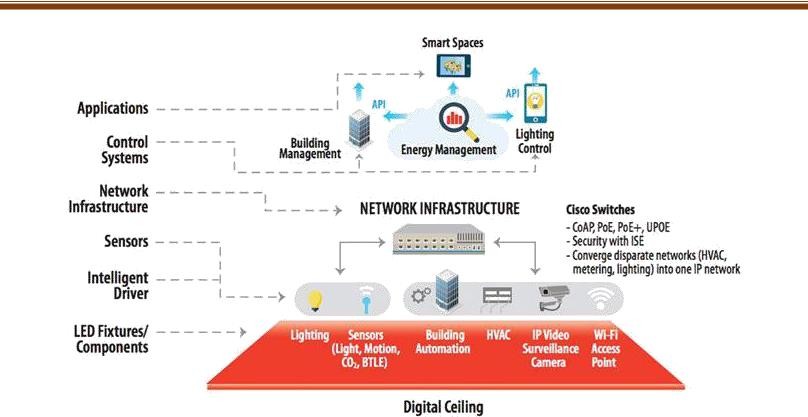
For example:

The de facto communication protocol responsible for building automation is known as BACnet (Building Automation and Control Network). In a nutshell, the BACnet protocol defines a set of services that allow Ethernet-based communication between building devices such as HVAC, lighting, access control, and fire detection systems. The same building Ethernet switches used for IT may also be used for BACnet. This standardization also makes possible an intersection point to the IP network (which is run by the IT department) through the use of a gateway device. In addition, BACnet/IP has been defined to allow the “things” in the building network to communicate over IP, thus allowing closer consolidation of the building management system on a single network.



**Figure 1.7** Convergence of Building Technologies to IP

Another promising IoT technology in the smart connected building, and one that is seeing widespread adoption, is the “digital ceiling.” The digital ceiling is more than just a lighting control system. This technology encompasses several of the building’s different networks—including lighting, HVAC, blinds, CCTV (closed-circuit television), and security systems—and combines them into a single IP network. Figure 1.8 provides a framework for the digital ceiling.



**Figure 1.8** A Framework for the Digital Ceiling

Central to digital ceiling technology is the lighting system. As you are probably aware, the lighting market is currently going through a major shift toward light-emitting diodes (LEDs).

Compared to traditional lighting, LEDs offer lower energy consumption and far longer life

##### Smart Creatures

IoT also provides the ability to connect living things to the Internet. Sensors can be placed on animals and even insects just as easily as on machines, and the benefits can be just as impressive. One of the most well-known applications of IoT with respect to animals focuses on what is often referred to as the “connected cow.” Sparked, a Dutch company, developed a sensor that is placed in a cow’s ear.

The sensor monitors various health aspects of the cow as well as its location and transmits the data wirelessly for analysis by the farmer. The data from each of these sensors is approximately 200 MB per year, and you obviously need a network infrastructure to make the connection with the sensors and store the information.

Once the data is being collected, however, you get a complete view of the herd, with statistics on every cow. You can learn how environmental factors may be affecting the herd as a whole and about changes in diet.

This enables early detection of disease as cows tend to eat less days before they show symptoms. These sensors even allow the detection of pregnancy in cows.

##### Convergence of IT and OT

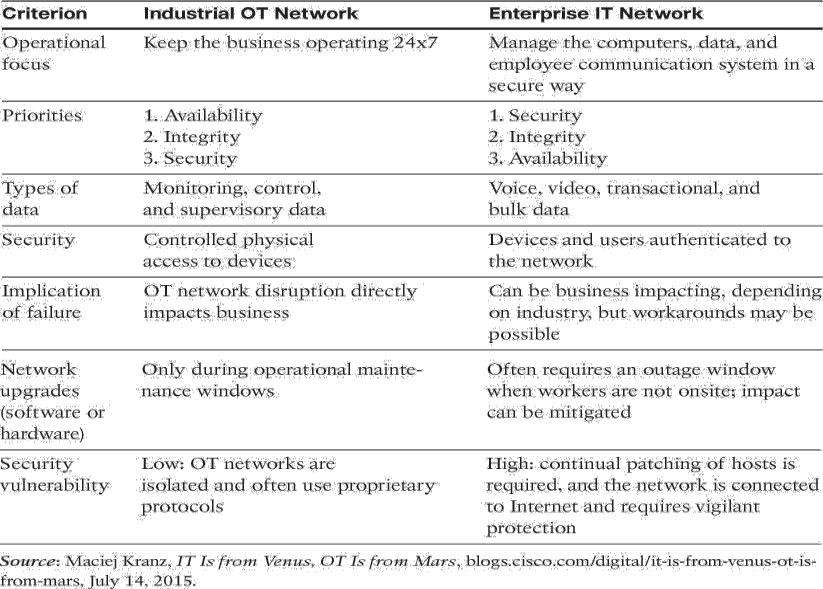
Until recently, information technology (IT) and operational technology (OT) have for the most part lived in separate worlds. IT supports connections to the Internet along with related data and technology systems and is focused on the secure flow of data across an organization.

OT monitors and controls devices and processes on physical operational systems. These systems include assembly lines, utility distribution networks, production facilities, roadway systems, and many more.

Typically, IT did not get involved with the production and logistics of OT environments. Specifically, the IT organization is responsible for the information systems of a business, such as email, file and print services, databases, and so on.

In comparison, OT is responsible for the devices and processes acting on industrial equipment, such as factory machines, meters, actuators, electrical distribution automation devices, SCADA (supervisory control and data acquisition) systems, and so on.

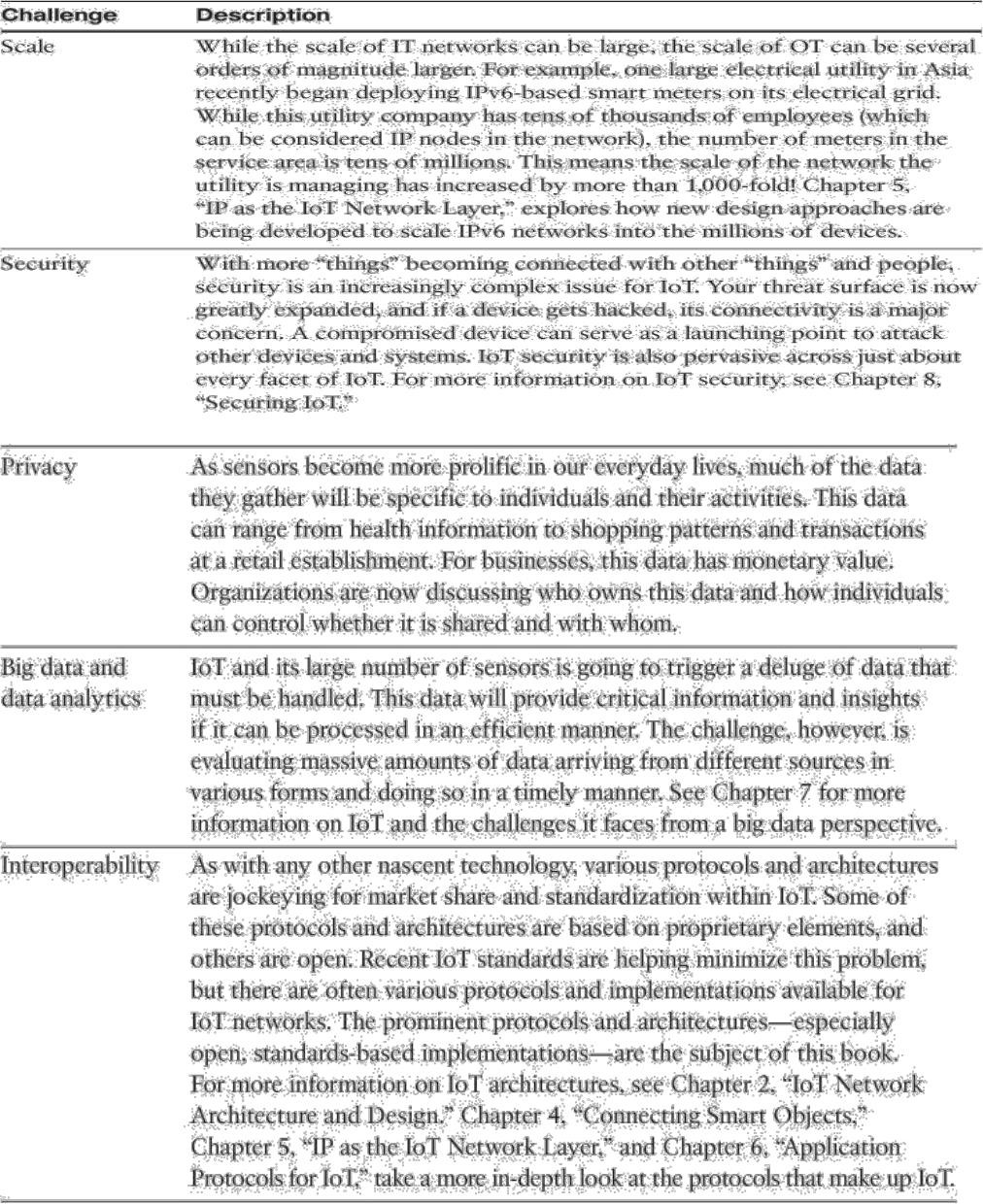
**Table 1.3** Comparing Operational Technology (OT) and Information Technology (IT)



* 1. **IoT Challenges**

While an IoT-enabled future paints an impressive picture, it does not come without significant challenges. Many parts of IoT have become reality, but certain obstacles need to be overcome for IoT to become ubiquitous throughout industry and our everyday life.

Table 1.4 highlights a few of the most significant challenges and problems that IoT is currently facing.



**Chapter 2**

**IoT Network Architecture and Design**

Enterprise IT network architecture has matured significantly over the past two decades and is generally well understood; however, the discipline of IoT network architecture is new and requires a fresh perspective.

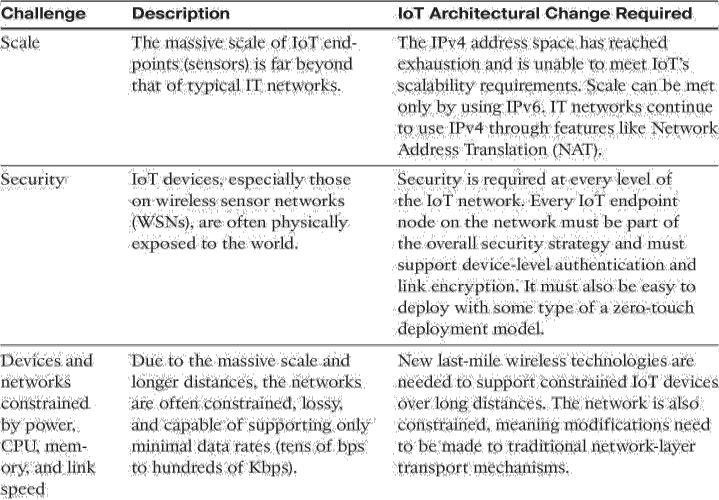
It is important to note that while some similarities between IT and IoT architectures do exist, for the most part, the challenges and requirements of IoT systems are radically different from those of traditional IT networks.

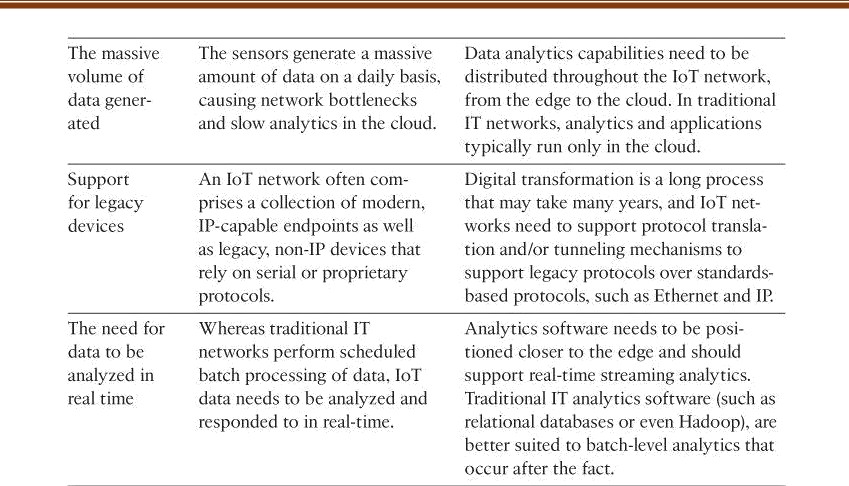
The terminology is also different to the point where IoT networks are often under the umbrella of OT, which is responsible for the management and state of operational systems. In contrast, IT networks are primarily concerned with the infrastructure that transports flows of data, regardless of the data type.

##### Drivers Behind New Network Architectures

While IT systems are mostly concerned with reliable and continuous support of business applications such as email, web, databases, CRM systems, and so on. IoT is all about the data generated by sensors and how that data is used. The essence of IoT architectures thus involves how the data is transported, collected, analyzed, and ultimately acted upon.

Table 2.1 takes a closer look at some of the differences between IT and IoT networks, with a focus on the IoT requirements that are driving new network architectures, and considers what adjustments are needed.

**Table 2.1** IoT Architectural Drivers



* 1. **Comparing IoT Architectures**

Two of the best-known architectures are

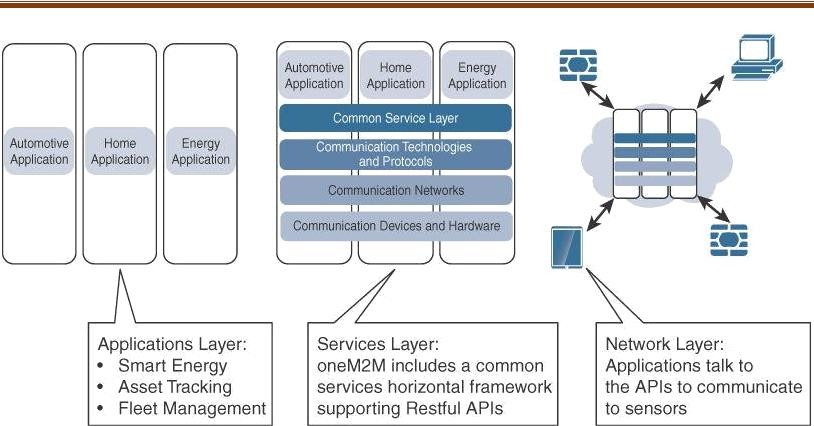
1. oneM2M and
2. The IoT World Forum (IoTWF)

##### The oneM2M IoT Standardized Architecture

In an effort to standardize the rapidly growing field of machine-to-machine (M2M) communications, the European Telecommunications Standards Institute (ETSI) created the M2M Technical Committee in 2008.

The goal of this committee was to create a common architecture that would help accelerate the adoption of M2M applications and devices. Over time, the scope has expanded to include the Internet of Things.

The goal of oneM2M is to create a common services layer, which can be readily embedded in field devices to allow communication with application servers. oneM2M’s framework focuses on IoT services, applications, and platforms. These include smart metering applications, smart grid, smart city automation, e-health, and connected vehicles.



**Figure 2.1** The Main Elements of the oneM2M IoT Architecture The oneM2M architecture divides IoT functions into three major domains:

* 1. The application layer,
  2. The services layer, and
  3. The network layer.

##### Applications layer:

The oneM2M architecture gives major attention to connectivity between devices and their applications. This domain includes the application-layer protocols and attempts to standardize northbound API definitions for interaction with business intelligence (BI) systems. Applications tend to be industry-specific and have their own sets of data models, and thus they are shown as vertical entities.

1. **Services layer:**

This layer is shown as a horizontal framework across the vertical industry applications. At this layer, horizontal modules include the physical network that the IoT applications run on, the underlying management protocols, and the hardware.

**Examples** include backhaul communications via cellular, MPLS networks, VPNs, and so on.

Riding on top is the common services layer. This conceptual layer adds APIs and middleware supporting third-party services and applications.

One of the stated goals of oneM2M is to “Develop technical specifications which address the need for a common M2M Service Layer that can be readily embedded within various hardware and software nodes, and rely upon connecting the myriad of devices in the field area network to M2M application servers, which typically reside in a cloud or data center.”

A critical objective of oneM2M is to attract and actively involve organizations from M2M-related business domains, including telematics and intelligent transportation, healthcare, utility, industrial automation, and smart home applications, to name just a few

##### Network layer:

This is the communication domain for the IoT devices and endpoints. It includes the devices themselves and the communications network that links them. Embodiments of this communications infrastructure include wireless mesh technologies, such as IEEE 802.15.4, and wireless point-to-multipoint systems, such as IEEE 801.11ah. Also included are wired device connections, such as IEEE 1901 power line communications.

##### The IoT World Forum (IoTWF) Standardized Architecture

In 2014 the IoTWF architectural committee (led by Cisco, IBM, Rockwell Automation, and others) published a seven-layer IoT architectural reference model. While various IoT reference models exist, the one put forth by the IoT World Forum offers a clean, simplified perspective on IoT and includes edge computing, data storage, and access.

It provides a succinct way of visualizing IoT from a technical perspective. Each of the seven layers is broken down into specific functions, and security encompasses the entire model.

As shown in Figure 2.2, the IoT Reference Model defines a set of levels with control flowing from the center (this could be either a cloud service or a dedicated data center), to the edge, which includes sensors, devices, machines, and other types of intelligent end nodes.

In general, data travels up the stack, originating from the edge, and goes northbound to the center. Using this reference model, we are able to achieve the following:



Decompose the IoT problem into smaller parts.

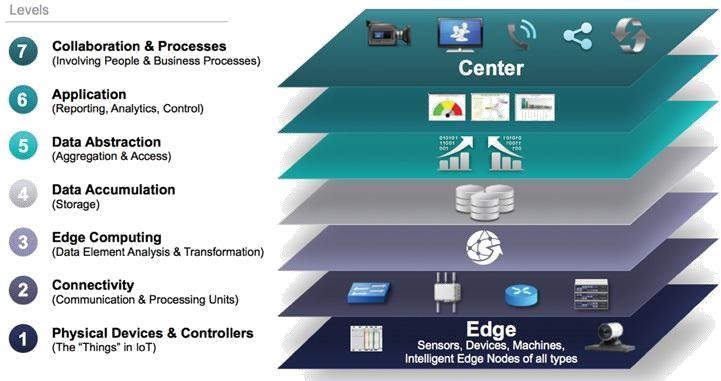
* Identify different technologies at each layer and how they relate to one another.
* Define a system in which different parts can be provided by different vendors.

# 

Have a process of defining interfaces that leads to interoperability

#### 

Define a tiered security model that is enforced at the transition points between levels



**Figure 2.2** IoT Reference Model Published by the IoT World Forum

##### Layer 1: Physical Devices and Controllers Layer

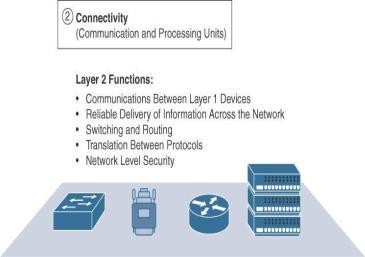
The first layer of the IoT Reference Model is the physical devices and controllers layer. This layer is home to the “things” in the Internet of Things, including the various endpoint devices and sensors that send and receive information. The size of these “things” can range from almost microscopic sensors to giant machines in a factory. Their primary function is generating data and being capable of being queried and/or controlled over a network.

##### Layer 2: Connectivity Layer

In the second layer of the IoT Reference Model, the focus is on connectivity. The most important function of this IoT layer is the reliable and timely transmission of data. More specifically, this includes transmissions between Layer 1 devices and the network and between the network and information processing that occurs at Layer 3 (the edge computing layer).

As we notice, the connectivity layer encompasses all networking elements of IoT and doesn’t really distinguish between the last-mile network (the network between the sensor/endpoint and the IoT gateway, discussed later in this chapter), gateway, and backhaul networks.

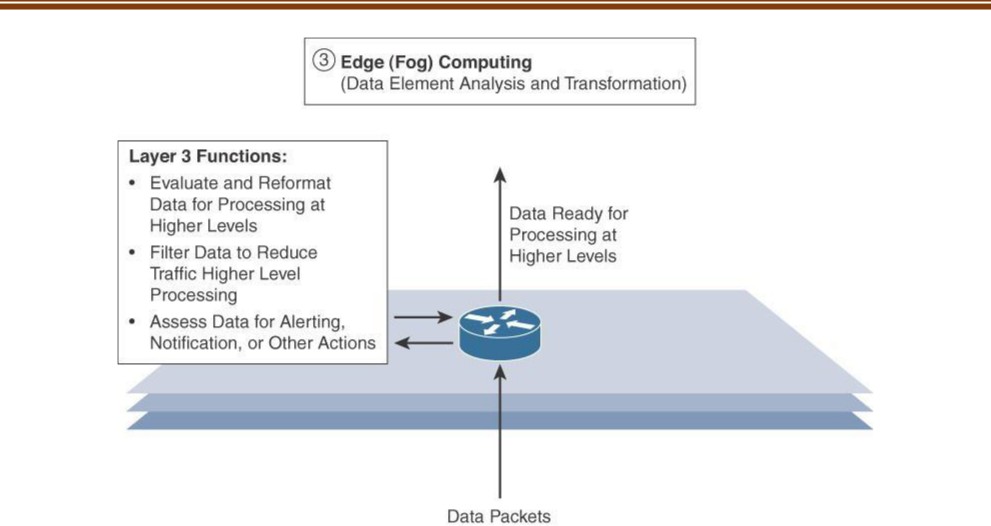
Functions of the connectivity layer are detailed in Figure 2-3.



**Figure 2-3** IoT Reference Model Connectivity Layer Functions

##### Layer 3: Edge Computing Layer

At this layer, the emphasis is on data reduction and converting network data flows into information that is ready for storage and processing by higher layers. One of the basic principles of this reference model is that information processing is initiated as early and as close to the edge of the network as possible.

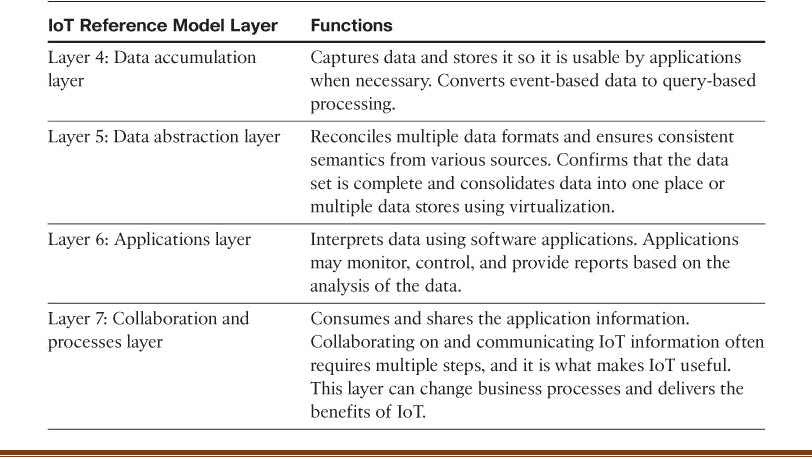


**Figure 2-4** IoT Reference Model Layer 3 Functions

**Upper Layers: Layers 4–7**

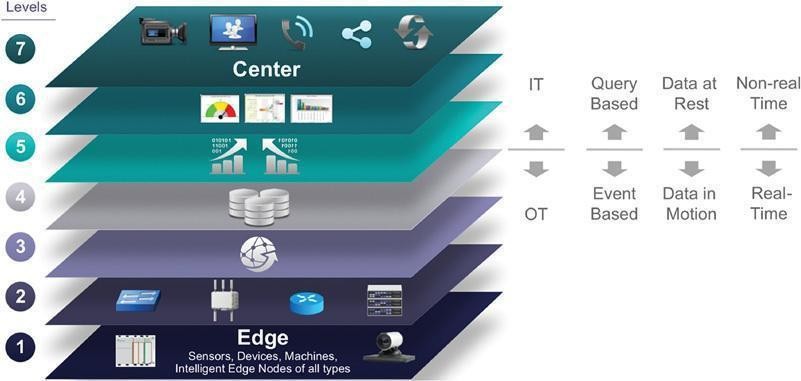
The upper layers deal with handling and processing the IoT data generated by the bottom layer. For the sake of completeness.

**Table 2.2** Summary of Layers 4–7 of the IoTWF Reference Model



##### IT and OT Responsibilities in the IoT Reference Model

An interesting aspect of visualizing an IoT architecture this way is that you can start to organize responsibilities along IT and OT lines. Figure 2-5 illustrates a natural demarcation point between IT and OT in the IoT Reference Model framework.



**Figure 2-5** IoT Reference Model Separation of IT and OT

As demonstrated in Figure 2-5, IoT systems have to cross several boundaries beyond just the functional layers. The bottom of the stack is generally in the domain of OT. For an industry like oil and gas, this includes sensors and devices connected to pipelines, oil rigs, refinery machinery, and so on.

The top of the stack is in the IT area and includes things like the servers, databases, and applications, all of which run on a part of the network controlled by IT. In the past, OT and IT have generally been very independent and had little need to even talk to each other. IoT is changing that paradigm.

At the bottom, in the OT layers, the devices generate real-time data at their own rate—sometimes vast amounts on a daily basis. Not only does this result in a huge amount of data transiting the IoT network, but the sheer volume of data suggests that applications at the top layer will be able to ingest that much data at the rate required.

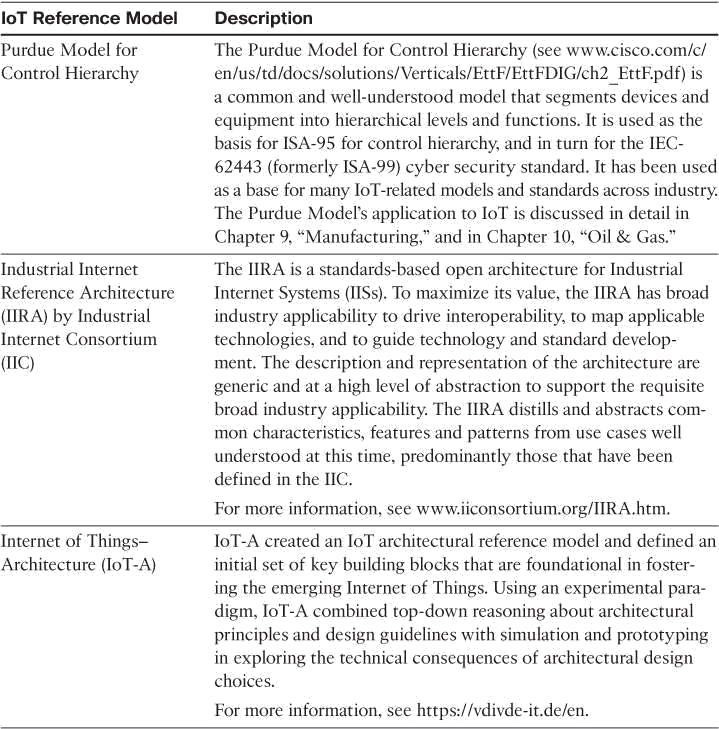
To meet this requirement, data has to be buffered or stored at certain points within the IoT stack. Layering data management in this way throughout the stack helps the top four layers handle data at their own speed.

As a result, the real-time “data in motion” close to the edge has to be organized and stored so that it becomes “data at rest” for the applications in the IT tiers. The IT and OT organizations need to work together for overall data management.

##### Additional IoT Reference Models

In addition to the two IoT reference models already presented in this chapter, several other reference models exist. These models are endorsed by various organizations and standards bodies and are often specific to certain industries or IoT applications. Table 2-3 highlights these additional IoT reference models.

**Table 2.3** Alternative IoT Reference Models

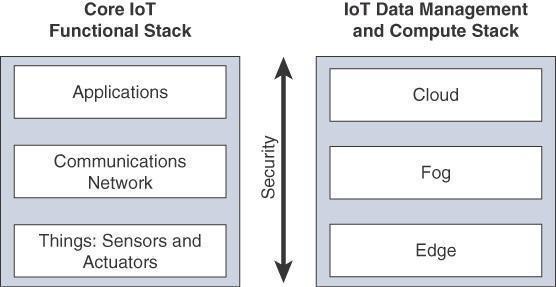


## A Simplified IoT Architecture

The framework is presented as two parallel stacks:

* + 1. The IoT Data Management and Compute Stack and
    2. The Core IoT Functional Stack.

Reducing the framework down to a pair of three-layer stacks in no way suggests that the model lacks the detail necessary to develop a sophisticated IoT strategy. Rather, the intention is to simplify the IoT architecture into its most basic building blocks and then to use it as a foundation to understand key design and deployment principles that are applied to industry-specific use cases. All the layers of more complex models are still covered, but they are grouped here in functional blocks that are easy to understand. Figure 2-6 illustrates the simplified IoT model.



**Figure 2.6** Simplified IoT Architecture

This separation gives you better visibility into the functions of each layer. The presentation of the Core IoT Functional Stack in three layers is meant to simplify your understanding of the IoT architecture into its most foundational building blocks. Figure 2-7 highlights an expanded view of the IoT architecture.Core IoT Functional Stack can be expanded into sublayers containing greater detail and specific network functions.

**For example,** the communications layer is broken down into four separate sublayers: the access network, gateways and backhaul, IP transport, and operations and management sublayers.

The applications layer of IoT networks is quite different from the application layer of a typical enterprise network. Instead of simply using business applications, IoT often involves a strong big

data analytics component. The applications layer typically has both analytics and industry- specific IoT control system components.

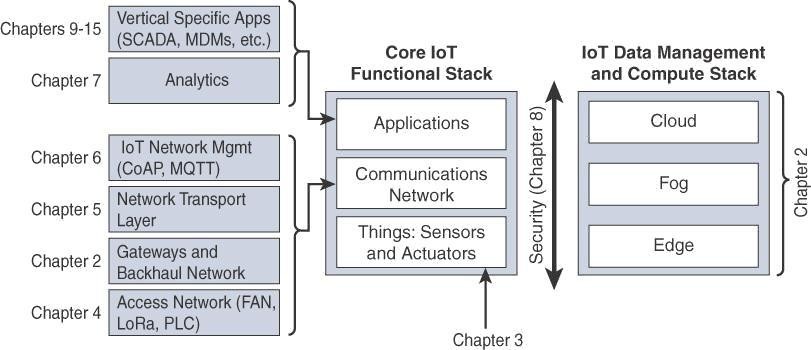


Figure 2-7 Expanded View of the Simplified IoT Architecture

## The Core IoT Functional Stack

From an architectural standpoint, several components have to work together for an IoT network to be operational:

**“Things” layer:** At this layer, the physical devices need to fit the constraints of the environment in which they are deployed while still being able to provide the information needed.

**Communications network layer:** When smart objects are not self-contained, they need to communicate with an external system. In many cases, this communication uses a wireless technology. This layer has four sub layers:

1. **Access network sub layer:** The last mile of the IoT network is the access network. This is typically made up of wireless technologies such as 802.11ah, 802.15.4g, and LoRa. The sensors connected to the access network may also be wired.
2. **Gateways and backhaul network sub layer:** A common communication system organizes multiple smart objects in a given area around a common gateway. The gateway communicates directly with the smart objects. The role of the gateway is to forward the collected information through a longer-range medium (called the backhaul) to a headend central station where the information is processed. This information exchange is a Layer 7 (application) function, which is the reason this object is called a gateway. On IP networks, this gateway also forwards packets from one IP network to another, and it therefore acts as a router.
3. **Network transport sublayer:** For communication to be successful, network and transport layer protocols such as IP and UDP must be implemented to support the variety of devices to connect and media to use.
4. **IoT network management sublayer:** Additional protocols must be in place to allow the headend applications to exchange data with the sensors. Examples include CoAP and MQTT.

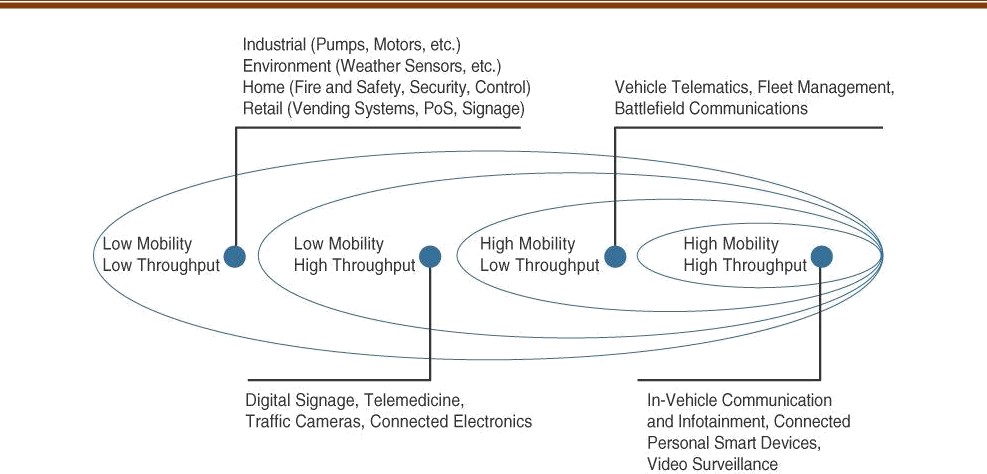
**Application and analytics layer:** At the upper layer, an application needs to process the collected data, not only to control the smart objects when necessary, but to make intelligent decision based on the information collected and, in turn, instruct the “things” or other systems to adapt to the analyzed conditions and change their behaviors or parameters.

##### Layer 1: Things: Sensors and Actuators Layer

Most IoT networks start from the object, or “thing,” that needs to be connected. There are myriad ways to classify smart objects.

1. **Battery-powered or power-connected:** This classification is based on whether the object carries its own energy supply or receives continuous power from an external power source.
2. **Mobile or static:** This classification is based on whether the “thing” should move or always stay at the same location. A sensor may be mobile because it is moved from one object to another.
3. **Low or high reporting frequency:** This classification is based on how often the object should report monitored parameters. A rust sensor may report values once a months
4. **Simple or rich data:** This classification is based on the quantity of data exchanged at each report cycle. A humidity sensor in a field may report a simple daily index value, while an engine sensor may report hundreds of parameters, from temperature to pressure, gas velocity, compression speed, carbon index, and many others. Richer data typically drives higher power consumption.
5. **Report range:** This classification is based on the distance at which the gateway is located.
6. **Object density per cell:** This classification is based on the number of smart objects over a given area, connected to the same gateway. An oil pipeline may utilize a single sensor at key locations every few miles.

Figure 2-8 provides some examples of applications matching the combination of mobility and throughput requirements.



**Figure 2-8** Example of Sensor Applications Based on Mobility and Throughput

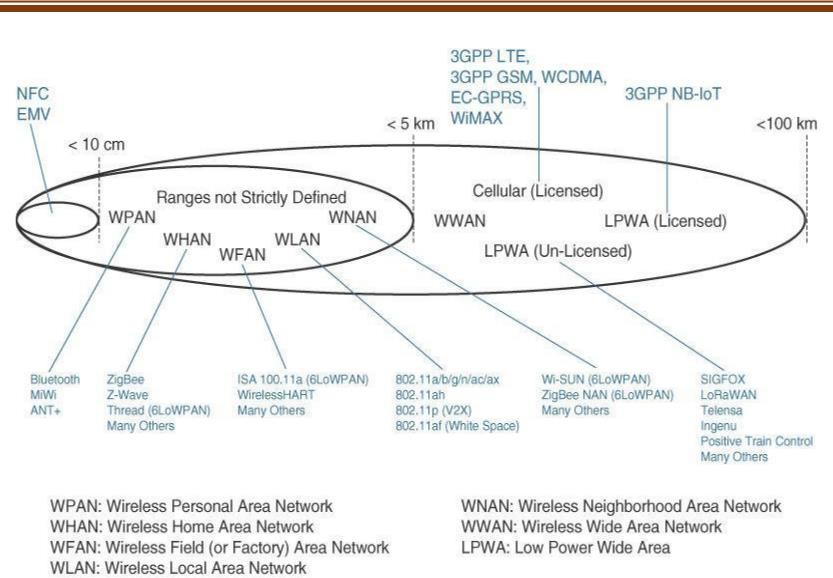
##### Layer 2: Communications Network Layer

Once you have determined the influence of the smart object form factor over its transmission capabilities (transmission range, data volume and frequency, sensor density and mobility), you are ready to connect the object and communicate.

##### Access Network Sublayer

There is a direct relationship between the IoT network technology you choose and the type of connectivity topology this technology allows. Each technology was designed with a certain number of use cases in mind (what to connect, where to connect, how much data to transport at what interval and over what distance).

These use cases determined the frequency band that was expected to be most suitable, the frame structure matching the expected data pattern (packet size and communication intervals), and the possible topologies that these use cases illustrate. One key parameter determining the choice of access technology is the range between the smart object and the information collector. Figure 2-9 lists some access technologies you may encounter in the IoT world and the expected transmission distances.



**Figure 2-9** Access Technologies and Distances

For example, cellular is indicated for transmissions beyond 5 km, but you could achieve a successful cellular transmission at shorter range (for example, 100 m). By contrast, ZigBee is expected to be efficient over a range of a few tens of meters, but you would not expect a successful ZigBee transmission over a range of 10 km.

Range estimates are grouped by category names that illustrate the environment or the vertical where data collection over that range is expected. Common groups are as follows:

* + **PAN (personal area network):** Scale of a few meters. This is the personal space around a person. A common wireless technology for this scale is Bluetooth.
  + **HAN (home area network):** Scale of a few tens of meters. At this scale, common wireless technologies for IoT include ZigBee and Bluetooth Low Energy (BLE).
  + **NAN (neighborhood area network):** Scale of a few hundreds of meters. The term NAN is often used to refer to a group of house units from which data is collected.
  + **FAN (field area network):** Scale of several tens of meters to several hundred meters. FAN typically refers to an outdoor area larger than a single group of house units. The FAN is often seen as “open space” (and therefore not secured and not controlled).
  + **LAN (local area network):** Scale of up to 100 m. This term is very common in networking, and it is therefore also commonly used in the IoT space when standard networking technologies (such as Ethernet or IEEE 802.11) are used. Other networking classifications, such as MAN (metropolitan area network, with a range of up to a few kilometers) and WAN (wide area network, with a range of more than a few kilometers), are also commonly used.

##### Gateways and Backhaul Sub layer

Data collected from a smart object may need to be forwarded to a central station where data is processed. As this station is often in a different location from the smart object, data directly received from the sensor through an access technology needs to be forwarded to another medium (the backhaul) and transported to the central station. The gateway is in charge of this inter-medium communication.

**For example,** dedicated short-range communication (DSRC) allows vehicle-to-vehicle and vehicle-to-infrastructure communication. In this model, the smart object’s position relative to the gateway is static. The car includes sensors and one gateway. Communication between the sensors and the gateway may involve wired or wireless technologies. Sensors may also be integrated into the road infrastructure and connect over a wired or wireless technology to a gateway on the side of the road. A wireless technology is used for backhaul communication, peer-to-peer, or mesh communication between vehicles.

##### Network Transport Sublayer

This communication structure thus may involve peer-to-peer (for example, meter to meter), point-to-point (meter to headend station), point-to-multipoint (gateway or head- end to multiple meters), unicast and multicast communications (software update to one or multiple systems).

In a multitenant environment (for example, electricity and gas consumption management), different systems may use the same communication pathways. This communication occurs over multiple media (for example, power lines inside your house or a short-range wireless system like indoor Wi-Fi and/or ZigBee), a longer-range wireless system to the gateway, and yet another wireless or wired medium for backhaul transmission.

##### IoT Network Management Sublayer

IP, TCP, and UDP bring connectivity to IoT networks. Upper-layer protocols need to take care of data transmission between the smart objects and other systems. Multiple protocols have been leveraged or created to solve IoT data communication problems. Some networks rely on a push model (that is, a sensor reports at a regular interval or based on a local trigger), whereas others rely on a pull model (that is, an application queries the sensor over the network), and multiple hybrid approaches are also possible.

Reusing well-known methods, Extensible Messaging and Presence Protocol (XMPP) was created. XMPP is based on instant messaging and presence. It allows the exchange of data between two or more systems and supports presence and contact list maintenance. It can also handle publish/subscribe, making it a good choice for distribution of information to multiple devices. A limitation of XMPP is its reliance on TCP, which may force subscribers to maintain open sessions to other systems and may be a limitation for memory-constrained objects.

##### Layer 3: Applications and Analytics Layer

Once connected to a network, your smart objects exchange information with other systems. As soon as your IoT network spans more than a few sensors, the power of the Internet of Things appears in the applications that make use of the information exchanged with the smart objects.

**Analytics Versus Control Applications**

* + **Analytics application:** This type of application collects data from multiple smart objects, processes the collected data, and displays information resulting from the data that was processed. The display can be about any aspect of the IoT network, from historical reports, statistics, or trends to individual system states.
  + **Control application:** This type of application controls the behavior of the smart object or the behavior of an object related to the smart object.

**For example,** a pressure sensor may be connected to a pump. A control application increases the pump speed when the connected sensor detects a drop in pressure. Control applications are very useful for controlling complex aspects of an IoT network with a logic that cannot be programmed inside a single IoT object.

**Data Versus Network Analytics**

Analytics is a general term that describes processing information to make sense of collected data

##### Data analytics:

This type of analytics processes the data collected by smart objects and combines it to provide an intelligent view related to the IoT system. At a very basic level, a dashboard can display an alarm when a weight sensor detects that a shelf is empty in a store. In a more complex case, temperature, pressure, wind, humidity, and light levels collected from thousands of sensors may be combined and then processed to determine the likelihood of a storm and its possible path. In this case, data processing can be very complex and may combine multiple changing values over complex algorithms.

##### Network analytics:

Most IoT systems are built around smart objects connected to the network. A loss or degradation in connectivity is likely to affect the efficiency of the system. Such a loss can have dramatic effects.

**For example,** open mines use wireless networks to automatically pilot dump trucks. A lasting loss of connectivity may result in an accident or degradation of operations efficiency (automated dump trucks typically stop upon connectivity loss).

**Data Analytics Versus Business Benefits**

Data analytics is undoubtedly a field where the value of IoT is booming. Almost any object can be connected, and multiple types of sensors can be installed on a given object. Collecting and interpreting the data generated by these devices is where the value of IoT is realized.

A smarter architectural choice may be to allow for an open system where the network is engineered to be flexible enough that other sensors may be added in the future, and where both upstream and downstream operations are allowed.

This flexibility allows for additional processing of the existing sensors and also deeper and more efficient interaction with the connected objects. This enhanced data processing can result in new added value for businesses that are not envisioned at the time when the system is initially deployed.

**An example of a flexible analytics and control application is Cisco Jasper,** which provides a turnkey cloud-based platform for IoT management and monetization. Consider the case of vending machines deployed throughout a city. At a basic level, these machines can be connected, and sensors can be deployed to report when a machine is in an error state. A repair person can be sent to address the issue when such a state is identified. This type of alert is a time saver and avoids the need for the repair team to tour all the machines in turn when only one may be malfunctioning.

## IoT Data Management and Compute Stack

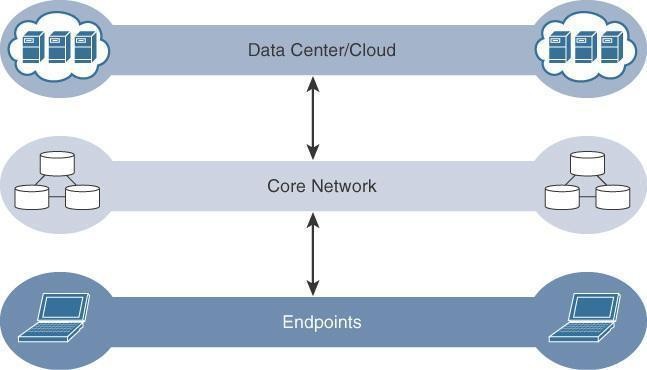
The data generated by IoT sensors is one of the single biggest challenges in building an IoT system. In the case of modern IT networks, the data sourced by a computer or server is typically generated by the client/server communications model, and it serves the needs of the application.

In sensor networks, the vast majority of data generated is unstructured and of very little use on its own. These requirements include the following:

1. **Minimizing latency:** Milliseconds matter for many types of industrial systems, such as when you are trying to prevent manufacturing line shutdowns or restore electrical service. Analyzing data close to the device that collected the data can make a difference between averting disaster and a cascading system failure.
2. **Conserving network bandwidth:** Offshore oil rigs generate 500 GB of data weekly. Commercial jets generate 10 TB for every 30 minutes of flight. It is not practical to transport vast amounts of data from thousands or hundreds of thousands of edge devices to the cloud. Nor is it necessary because many critical analyses do not require cloud-scale processing and storage.
3. **Increasing local efficiency:** Collecting and securing data across a wide geographic area with different environmental conditions may not be useful. The environmental conditions

in one area will trigger a local response independent from the conditions of another site hundreds of miles away. Analyzing both areas in the same cloud system may not be necessary for immediate efficiency.

As illustrated in Figure 2-14, data management in traditional IT systems is very simple. The endpoints (laptops, printers, IP phones, and so on) communicate over an IP core network to servers in the data center or cloud. Data is generally stored in the data center, and the physical links from access to core are typically high bandwidth, meaning access to IT data is quick.



**Figure 2-14** The Traditional IT Cloud Computing Model

IoT systems function differently. Several data-related problems need to be addressed:

# 

**Bandwidth in last-mile IoT networks is very limited**. When dealing with thousands/millions of devices, available bandwidth may be on order of tens of Kbps per device or even less.

#### 

**Latency can be very high**. Instead of dealing with latency in the milliseconds range, large IoT networks often introduce latency of hundreds to thousands of milliseconds.

# 

**Network backhaul from the gateway can be unreliable** and often depends on 3G/LTE or even satellite links. Backhaul links can also be expensive if a per-byte data usage model is necessary.

#### 

**The volume of data transmitted over the backhaul can be high**, and much of the data may not really be that interesting (such as simple polling messages).

# 

**Big data is getting bigger**. The concept of storing and analyzing all sensor data in the cloud is impractical. The sheer volume of data generated makes real-time analysis and response to the data almost impossible.

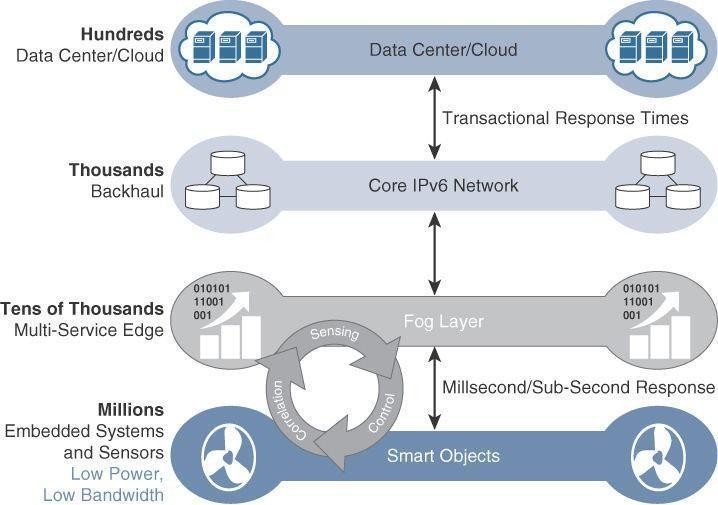
## Fog Computing

The best-known embodiment of edge services in IoT is fog computing. Any device with computing, storage, and network connectivity can be a fog node.

**Examples include** industrial controllers, switches, routers, embedded servers, and IoT gateways. Analyzing IoT data close to where it is collected minimizes latency, offloads gigabytes of network traffic from the core network, and keeps sensitive data inside the local network.

An advantage of this structure is that the fog node allows intelligence gathering (such as analytics) and control from the closest possible point, and in doing so, it allows better performance over constrained networks. In one sense, this introduces a new layer to the traditional IT computing model, one that is often referred to as the “fog layer.”

Figure 2-15 shows the placement of the fog layer in the IoT Data Management and Compute Stack.



**Figure 2-15** The IoT Data Management and Compute Stack with Fog Computing

Fog services are typically accomplished very close to the edge device, sitting as close to the IoT endpoints as possible. One significant advantage of this is that the fog node has contextual awareness of the sensors it is managing because of its geographic proximity to those sensors.

For example, there might be a fog router on an oil derrick that is monitoring all the sensor activity at that location. Because the fog node is able to analyze information from all the sensors on that derrick, it can provide contextual analysis of the messages it is receiving and may decide to send back only the relevant information over the backhaul network to the cloud. In this way, it is

performing distributed analytics such that the volume of data sent upstream is greatly reduced and is much more useful to application and analytics servers residing in the cloud.

In addition, having contextual awareness gives fog nodes the ability to react to events in the IoT network much more quickly than in the traditional IT compute model.The fog layer thus provides a distributed edge control loop capability, where devices can be monitored, controlled, and analyzed in real time without the need to wait for communication from the central analytics and application servers in the cloud.

##### For example,

Tire pressure sensors on a large truck in an open-pit mine might continually report measurements all day long. There may be only minor pressure changes that are well within tolerance limits, making continual reporting to the cloud unnecessary. Is it really useful to continually send such data back to the cloud over a potentially expensive backhaul connection? With a fog node on the truck, it is possible to not only measure the pressure of all tires at once but also combine this data with information coming from other sensors in the engine, hydraulics, and so on.

With this approach, the fog node sends alert data upstream only if an actual problem is beginning to occur on the truck that affects operational efficiency. IoT fog computing enables data to be preprocessed and correlated with other inputs to produce relevant information.

This data can then be used as real-time, actionable knowledge by IoT-enabled applications. Longer term, this data can be used to gain a deeper understanding of network behavior and systems for the purpose of developing proactive policies, processes, and responses. Fog applications are as diverse as the Internet of Things itself.

What they have in common is data reduction, monitoring or analyzing real-time data from network-connected things and then initiating an action, such as locking a door, changing equipment settings, applying the brakes on a train, zooming a video camera, opening a valve in response to a pressure reading, creating a bar chart, or sending an alert to a technician to make a preventive repair.

# 

The defining characteristic of fog computing are as follows:

#### 

**Contextual location awareness and low latency:** The fog node sits as close to the IoT endpoint as possible to deliver distributed computing.

# 

**Geographic distribution:** In sharp contrast to the more centralized cloud, the

services and applications targeted by the fog nodes demand widely distributed deployments.

### 

**Deployment near IoT endpoints:** Fog nodes are typically deployed in the presence of a large number of IoT endpoints.

# 

##### Wireless communication between the fog and the IoT endpoint: Although it is

possible to connect wired nodes, the advantages of fog are greatest when dealing with a large number of endpoints, and wireless access is the easiest way to achieve such scale.



**Use for real-time interactions:** Important fog applications involve real-time interactions rather than batch processing.

##### Edge Computing

In recent years, the concept of IoT computing has been pushed even further to the edge, and in some cases it now resides directly in the sensors and IoT devices.

IoT devices and sensors often have constrained resources, however, as compute capabilities increase. Some new classes of IoT endpoints have enough compute capabilities to perform at least low-level analytics and filtering to make basic decisions.

**For example,** consider a water sensor on a fire hydrant. While a fog node sitting on an electrical pole in the distribution network may have an excellent view of all the fire hydrants in a local neighborhood, a node on each hydrant would have clear view of a water pressure drop on its own line and would be able to quickly generate an alert of a localized problem. The fog node, on the other hand, would have a wider view and would be able to ascertain whether the problem was more than just localized but was affecting the entire area.

**Another example** is in the use of smart meters. Edge compute–capable meters are able to communicate with each other to share information on small subsets of the electrical distribution grid to monitor localized power quality and consumption, and they can inform a fog node of events that may pertain to only tiny sections of the grid. Models such as these help ensure the highest quality of power delivery to customers.

##### The Hierarchy of Edge, Fog, and Cloud

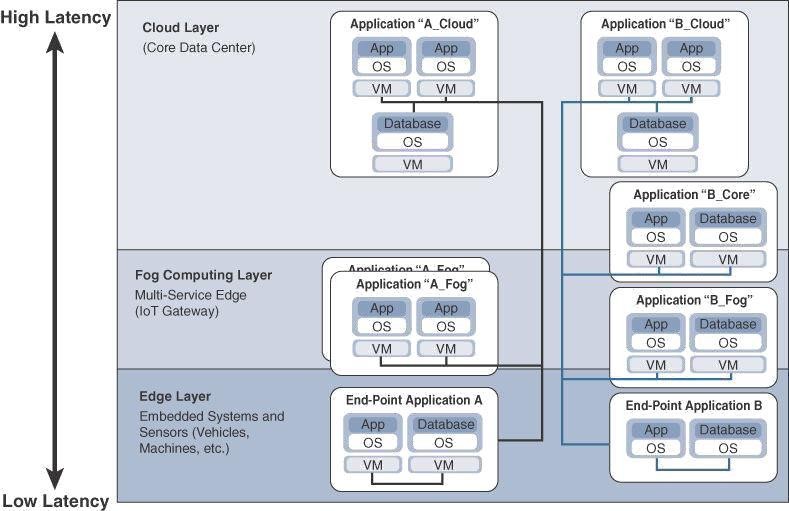
It is important to stress that edge or fog computing in no way replaces the cloud. In the same way that lower courts do not replace the supreme court of a country, edge and fog computing layers simply act as a first line of defense for filtering, analyzing, and otherwise managing data endpoints. This saves the cloud from being queried by each and every node for each event.

This model suggests a hierarchical organization of network, compute, and data storage resources.

At each stage, data is collected, analyzed, and responded to when necessary, according to the capabilities of the resources at each layer. As data needs to be sent to the cloud, the latency becomes higher. It is important to note that the heterogeneity of IoT devices also means a heterogeneity of edge and fog computing resources.

While cloud resources are expected to be homogenous, it is fair to expect that in many cases both edge and fog resources will use different operating systems, have different CPU and data storage capabilities, and have different energy consumption profiles. Edge and fog thus require an abstraction layer that allows applications to communicate with one another.

The abstraction layer exposes a common set of APIs for monitoring, provisioning, and controlling the physical resources in a standardized way. The abstraction layer also requires a mechanism to support virtualization, with the ability to run multiple operating systems or service containers on physical devices to support multitenancy and application consistency across the IoT system. Figure 2-16 illustrates the hierarchical nature of edge, fog, and cloud.



**Figure 2-16** Distributed Compute and Data Management Across an IoT System

From an architectural standpoint, fog nodes closest to the network edge receive the data from IoT devices.

The fog IoT application then directs different types of data to the optimal place for analysis:

1. The most time-sensitive data is analyzed on the edge or fog node closest to the things generating the data. Data that can wait seconds or minutes for action is passed along to an aggregation node for analysis and action.
2. Data that is less time sensitive is sent to the cloud for historical analysis, big data analytics, and long-term storage. For example, each of thousands or hundreds of thousands of fog nodes might send periodic summaries of data to the cloud for historical analysis and storage.