INTERNET OF THINGS

Theory – 410343

WSN & Cloud Computing

WSN Basics

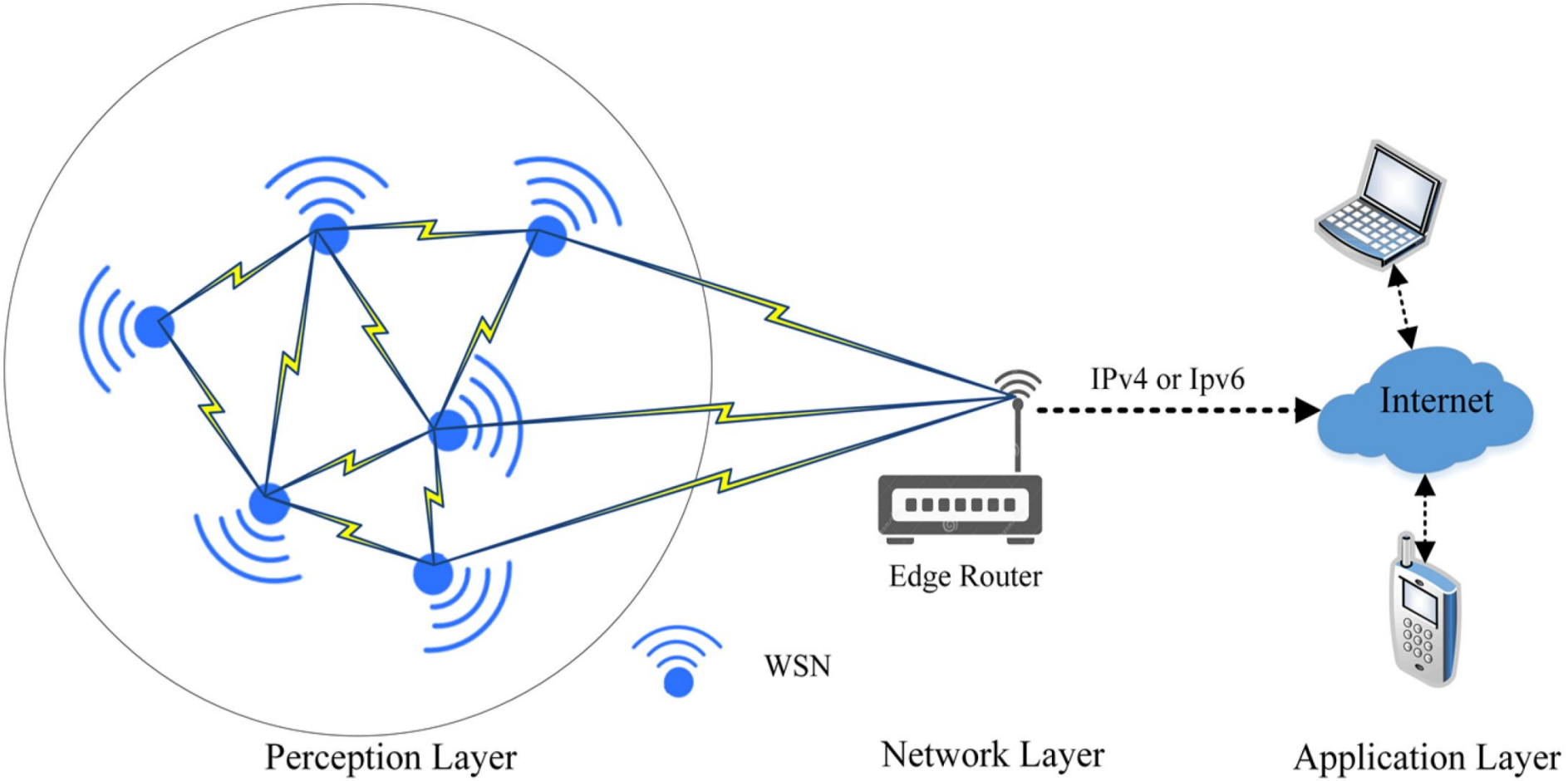
- Wireless Sensor Network - a network of spatially distributed autonomous sensors that monitor physical/environmental conditions and communicate the collected data to central location or other nodes in the network.

  sensors -temperature, humidity, light, pressure, motion.

- Features of WSNs-
  - Wireless Communication
  - Autonomy of nodes
  - Distributed Deployment of nodes
  - Energy Efficiency of nodes
  - Some edge computing

- Typical Application fields –
- Environmental monitoring, industrial automation, healthcare, agriculture, surveillance, disaster management

A WSN schematic-



Features of WSNs

- Wireless Sensor Networks (WSNs) possess several distinctive features that set them apart from other types of networks.

- **Distributed Sensing:** numerous small, autonomous sensor nodes that are spread out in a particular geographical area. nodes collaboratively gather data.

- **Wireless Communication:** Communication is accomplished through wireless radio frequency links. Eliminates the need for physical connections/ flexibility.

- **Energy Constraints:** Sensor nodes powered by batteries or other limited energy sources. Energy efficiency is critical/ sleep modes.

- **Dynamic Topology:** Network topology of a WSN can change due to node mobility, failures, and environmental changes. Adaptive routing protocols are used to handle dynamic topology changes.

Cont.

- **Self-Organization:** Sensor nodes autonomously configure themselves into a network. They can form ad-hoc networks based on the available nodes and communication links.

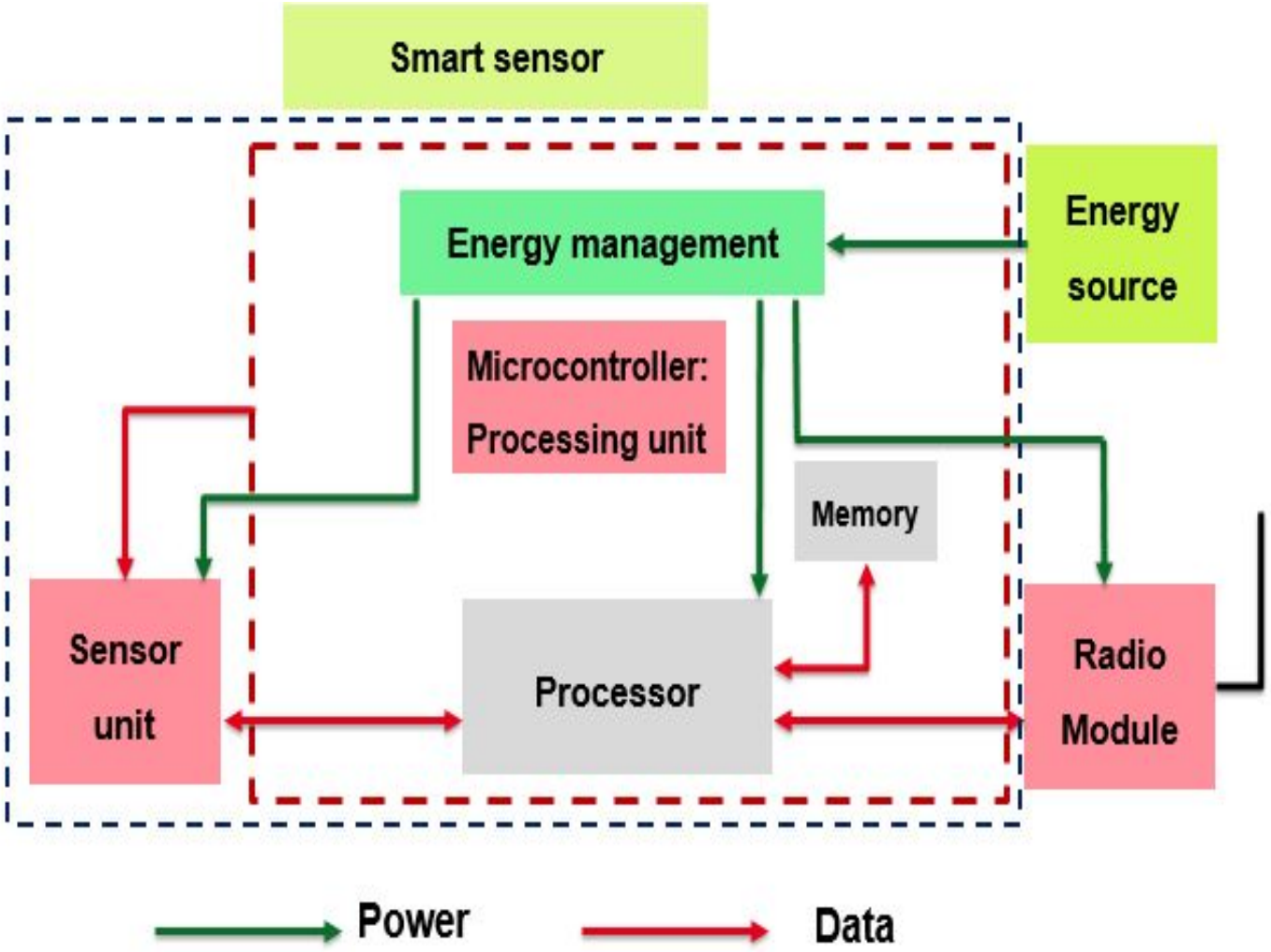
- **Adaptive Communication:** Sensor nodes adjust their communication strategies based on energy levels, channel conditions, and network congestion. helps in optimizing energy consumption and communication reliability.

- **Data Localization:** WSNs often focus on localized data collection and processing. Data is collected and processed locally before transmission, reducing the need for centralized processing.

- **Scalability & Heterogeneity:** WSNs can be scaled up or down to cover dynamic geographical areas. Sensor nodes can have varying capabilities, such as different types, processing power, and communication ranges.

Concept of Sensor Node

- A typical sensor node in a WSN



Typical challenges for designing WSNs

- These challenges arise from technological, operational, and environmental factors and can impact WSN design, deployment, and operation.

- **Limited Energy Resources:** Sensor nodes powered by batteries and have limited energy capacity. Energy-efficient design, power management, and energy harvesting techniques.
- **Communication Reliability:** Wireless communication is susceptible to signal interference, fading, and path loss, which can lead to unreliable data transmission. Robust communication protocols and error correction mechanisms.
- **Security and Privacy:** WSNs are susceptible to security threats such as eavesdropping, data tampering, and node impersonation. secure communication protocols, encryption techniques, and intrusion detection systems.
- **Dynamic Network Topology:** Sensor nodes in WSNs may be mobile or placed in dynamic environments, leading to frequent changes in network topology.

Cont.

- **Deployment and Maintenance:** often deployed in remote or harsh environments, making initial deployment and maintenance challenging. Accessing, repairing, and updating nodes in such locations can be difficult.

- **Data Processing and Storage Constraints:** Sensor nodes have limited processing power and memory. efficient data processing and storage mechanisms that meet application requirements within these constraints.

- **Interference and Coexistence:** In environments with multiple WSNs or other wireless technologies, interference and coexistence issues can arise. Proper channel allocation, interference avoidance, and spectrum management are critical.

- **Data Aggregation and Fusion:** Aggregating and fusing data from multiple sensor nodes is important for reducing redundant transmissions and conserving energy. Designing efficient data aggregation algorithms that balance accuracy and energy savings can be complex.

WSN-Basic components

- Components that ensure that the network can collect, transmit, and process data effectively.

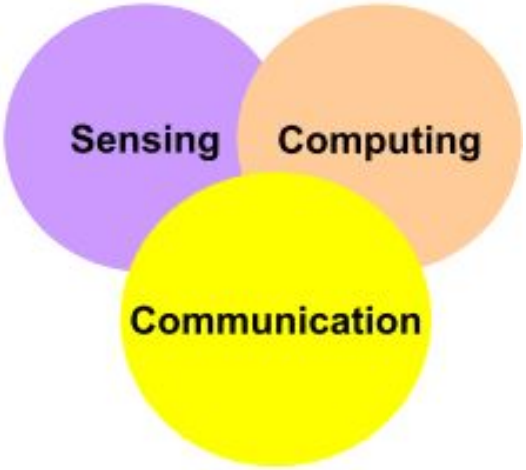
- **Sensor Nodes –**

  Each node - one or more sensors -measure specific environmental parameters like temperature, humidity, light, motion.
- Battery-powered, limited processing /communication capabilities.
- Transreceiver , Microcont., Power Unit , Memory, Edge processing unit.

- **Networking Protocol-**
- communication protocol for sensor nodes to exchange data.
- Protocols define how nodes communicate, handle collisions, establish connections, and manage data flow.
- Examples - Zigbee, Bluetooth Low Energy (BLE), and IEEE 802.15.4.

Cont.

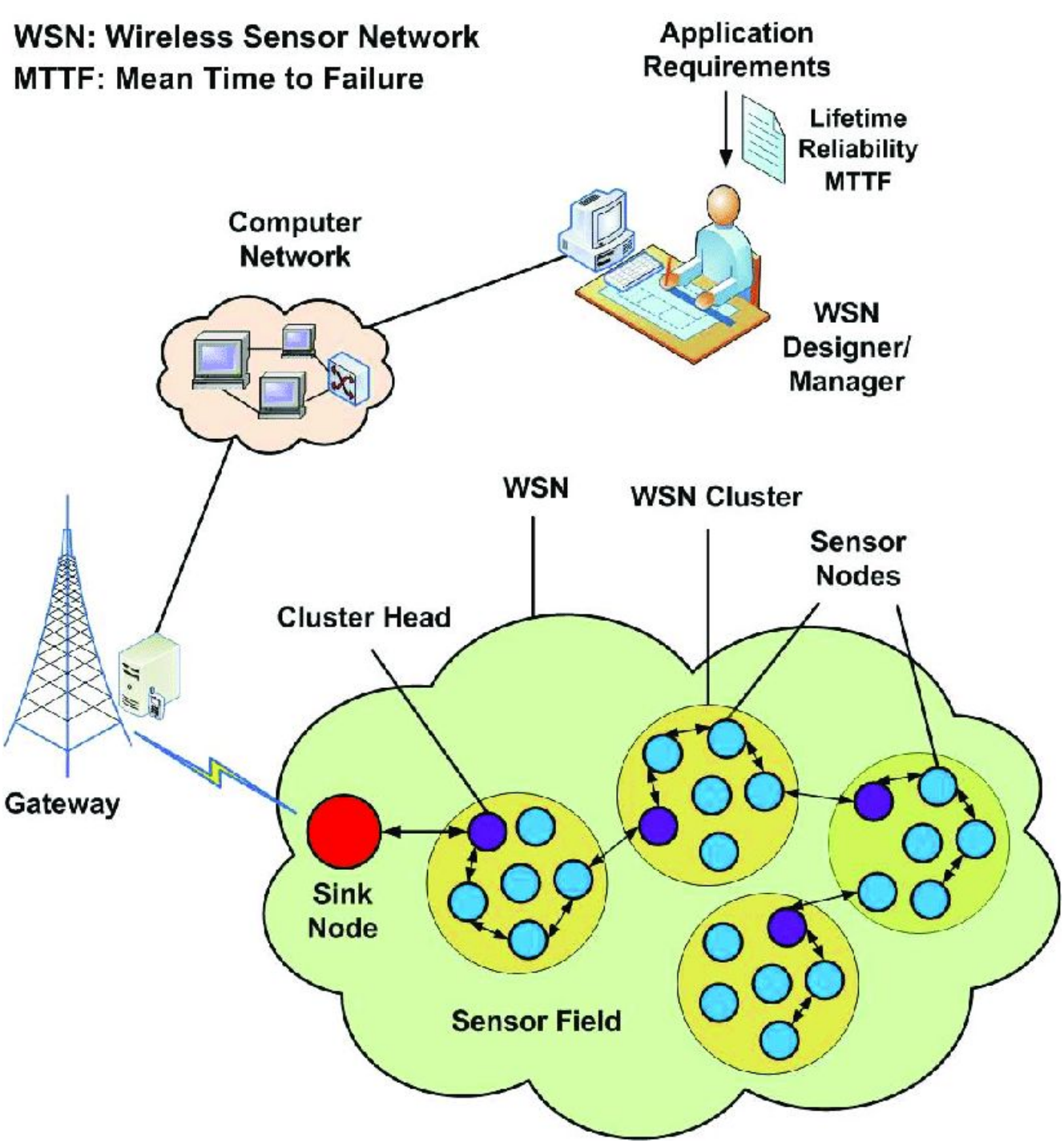
- **Base Station/Sink Node:**
- The base station or sink node collects data from sensor nodes and serves as a gateway to higher-level networks or the internet.
- It performs more substantial data processing acts as coordinator for managing the network.



- **Security Mechanisms:**
- WSNs handle sensitive data, making security crucial. Encryption, authentication, and intrusion detection mechanisms.

- **Energy management –**
- duty cycling, sleep modes, and adaptive data transmission.

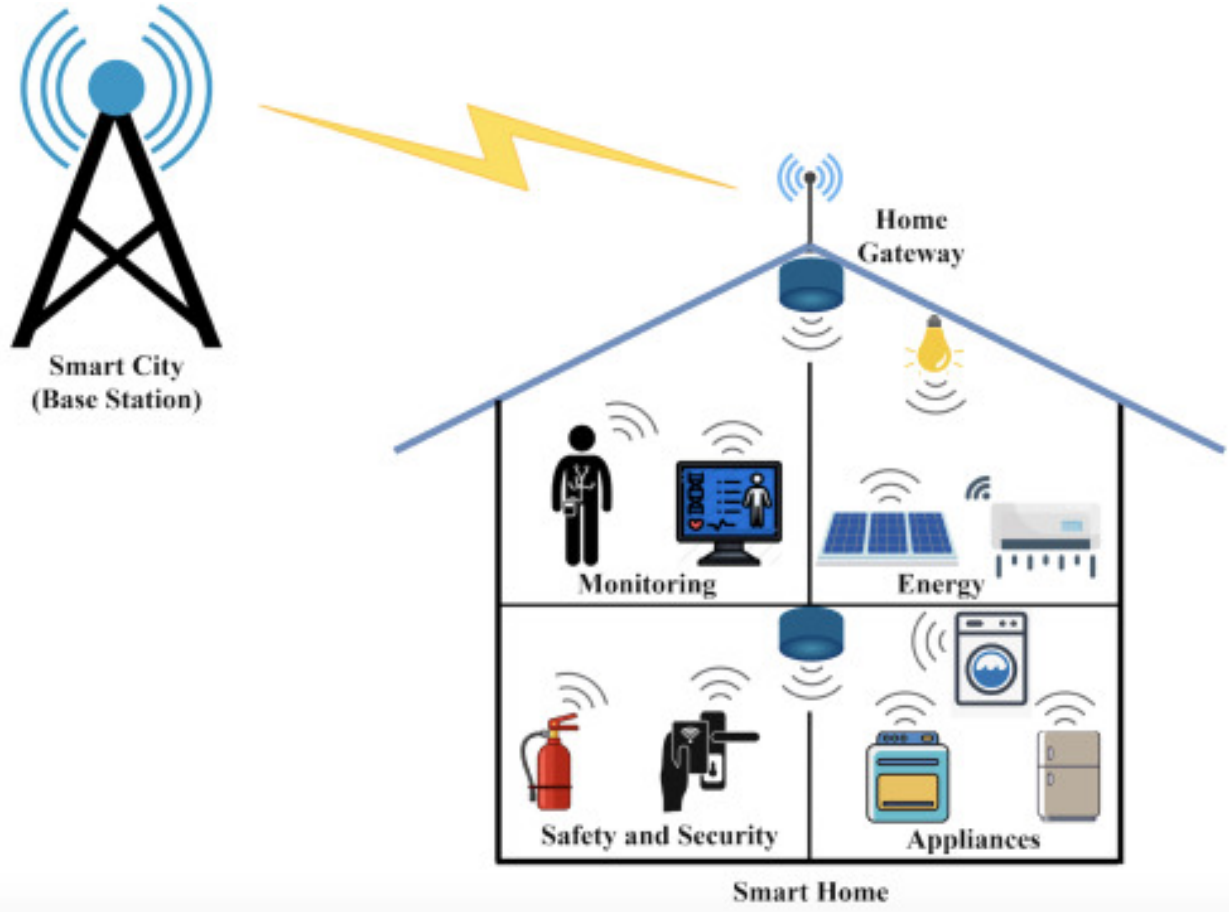
- **User Interface:** Depending on the application, a user interface to interact with and visualize collected data.


WSN: Wireless Sensor Network
MTTF: Mean Time to Failure
Application Requirements
Lifetime
Reliability
MTTF
Computer Network
WSN Designer/ Manager
WSN
WSN Cluster
Sensor Nodes
Cluster Head
Gateway
Sink Node
Sensor Field

Case study – Applications of WSN at homes

- A collection of interconnected sensors that communicate wirelessly to monitor and control various aspects of a residential environment.
- Enable automation/intelligent decision-making to enhance the comfort, security, energy efficiency, and convenience of the home.

- Focus Areas –
- Monitoring
- Energy
- Safety
- Comfort



Case study – Components of WSN at homes

- **Sensors:** detect /measure specific parameters.
  - Temperature and humidity sensors monitor climate conditions.
  - Motion sensors detect movement within rooms.
  - Light sensors assess ambient lighting.
  - Proximity sensors -Door/window sensors track openings and closures.
  - Smoke and gas sensors provide safety alerts.

- **Wireless Communication:** Sensors communicate with each other and a central controller using wireless protocols such as Wi-Fi, Zigbee, Z-Wave, Bluetooth, or Thread. Factors like range, power consumption, and data transfer rate.

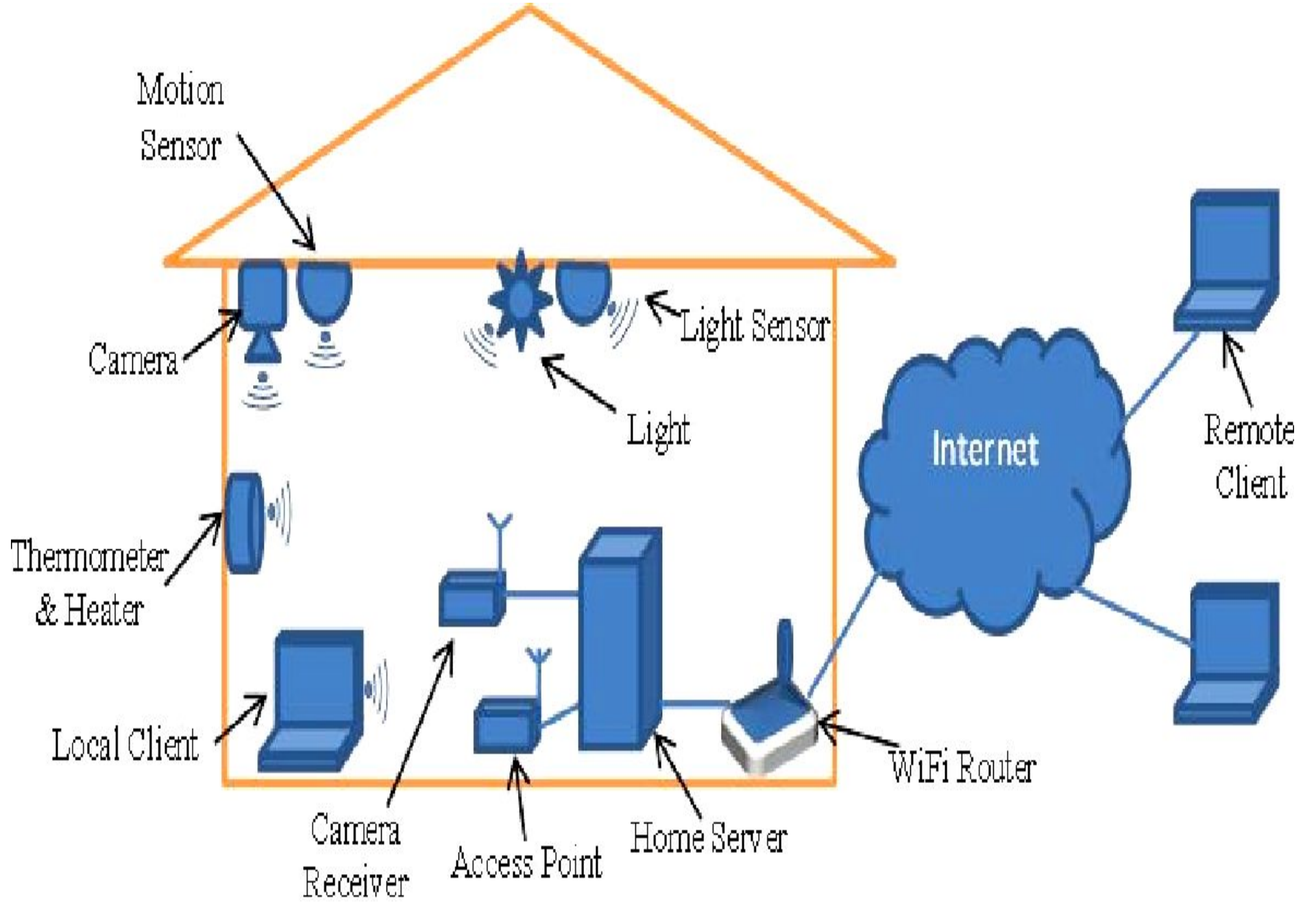
- **Central Hub/Controller:** Receives data from sensors and can process, analyze, and trigger actions based on that data. It can also serve as a user interface, allowing homeowners to monitor and control their smart home devices through a mobile app or a web interface.

Case study – Applications of WSN at homes

- **Data Processing and Analysis:** The central hub processes the data received from sensors and may apply algorithms for data analysis. Identify patterns, anomalies, and trends in the collected data to make informed decisions.

- **Automation and Control Algo:** Based on the processed data, the smart home system can automatically control various devices and systems.
  - Adjusting thermostats and HVAC systems based on temperature and occupancy.
  - Turning lights on or off based on room occupancy or ambient light levels.
  - Sending alerts or notifications in case of security breaches or unusual events.
  - Initiating actions like locking doors, shutting off water supply, or triggering alarms based on time and RTC.

- **User Interface:** interaction with the smart home system using a mobile app, a web portal, or voice-controlled virtual assistants like Amazon Alexa or Google Assistant. monitor sensor data, control devices remotely, and set up custom automation routines.


Motion
Sensor
Camera
Light Sensor
Light
Internet
Remote
Client
Thermometer
& Heater
Local Client
WiFi Router
Camera
Receiver
Access Point
Home Server

Case study – Applications of WSN in Transportation

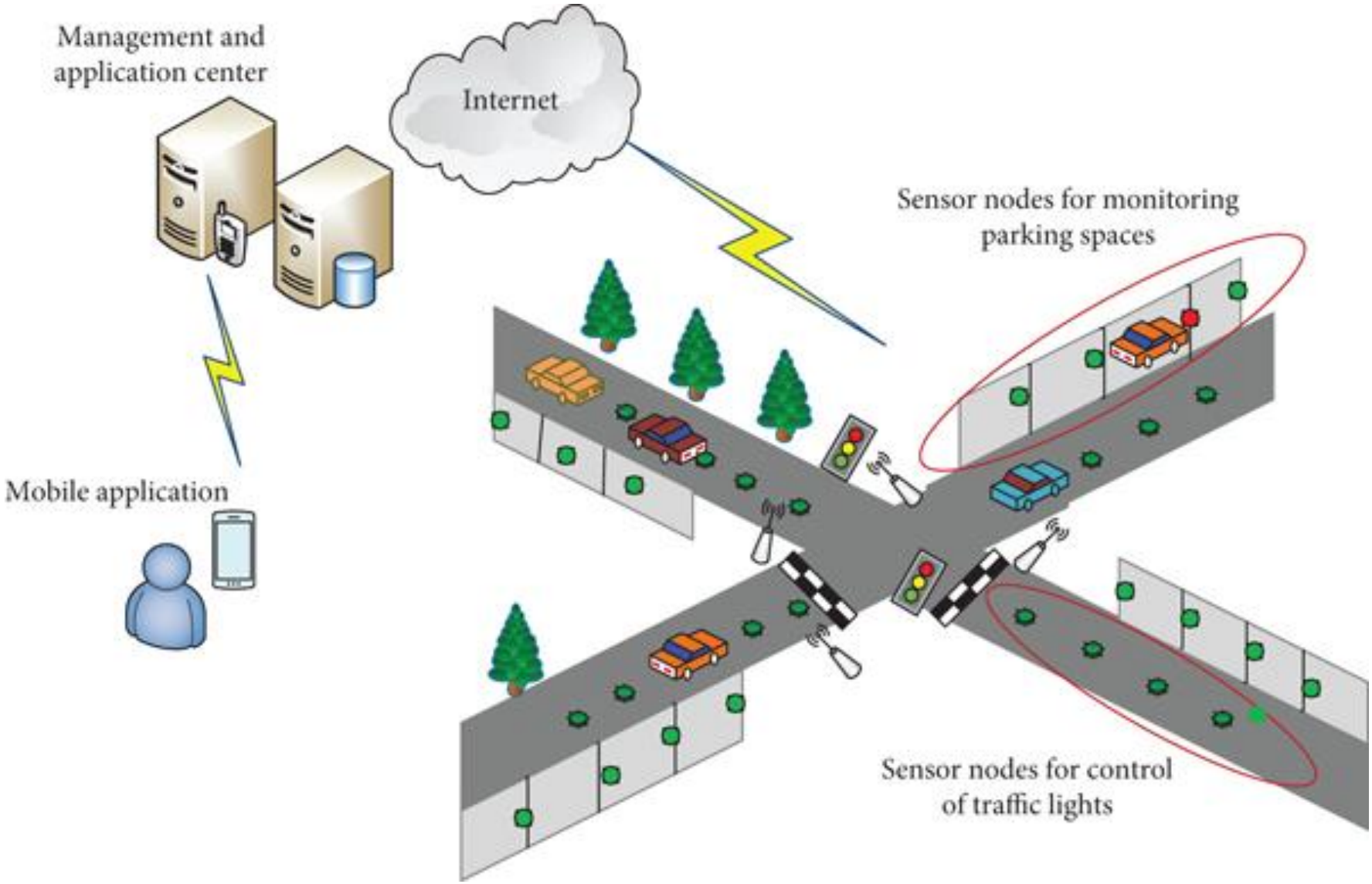
- WSN in the context of transportation involves deploying interconnected sensors and communication technologies to monitor and manage various aspects of transportation systems.

1. Vehicle Monitoring and Management:

- **Vehicle Health Monitoring:** Sensors installed in vehicles can continuously monitor their health, including engine performance, tire pressure, fuel consumption, and emission levels. This data can be used for maintenance scheduling and early detection of mechanical issues.

- **Fleet Management:** In logistics and fleet management, WSNs can track the location and condition of vehicles in real time. This aids in optimizing routes, improving fuel efficiency, and ensuring timely deliveries.

- **Traffic Management:** WSNs can monitor traffic flow and congestion in urban areas. This data can be used to optimize traffic signal timings, manage traffic incidents, and reduce congestion.


Management and application center
Internet
Sensor nodes for monitoring parking spaces
Mobile application
Sensor nodes for control of traffic lights

Case study – Applications of WSN in Transportation

2. Infrastructure Monitoring:

- **Bridge and Road Condition Monitoring:** WSNs can be deployed on bridges and roads to monitor their structural health and detect signs of wear and tear. This information helps prioritize maintenance and prevent accidents.

- **Railway Track Monitoring:** Sensors placed along railway tracks can monitor track conditions, identify defects, and help prevent derailments and accidents.

3. Public Transportation Systems:

- **Real-time Information:** WSNs can provide real-time information to passengers about bus, train, and subway schedules, delays, and available seats.

- **Passenger Counting:** Sensors can be used to count passengers boarding and disembarking public transportation vehicles. This data aids in capacity planning and optimizing services.

Case study – Applications of WSN in Transportation

4. Environmental Monitoring:

- **Air Quality Monitoring:** WSNs can measure air pollution levels along transportation routes and in urban areas. This data can be used to develop pollution mitigation strategies.

- **Noise Level Monitoring:** Sensors can measure noise levels near transportation hubs and busy roadways. This information can help identify noise pollution hotspots and implement noise reduction measures.

5. Emergency Response:

- **Accident Detection:** WSNs can detect sudden changes in vehicle speed, impact, or rollovers, signaling potential accidents. This information can trigger emergency response systems.

- **Emergency Call Systems:** In the event of an accident, vehicles equipped with WSNs can automatically send distress signals to emergency services, providing them with accurate location information.

Case study – Applications of WSN in Transportation

6. Parking Management:

- **Smart Parking:** WSNs can help drivers find available parking spaces in crowded areas, reducing traffic congestion and emissions caused by circling for parking.

- **Parking Payment:** Sensors can enable cashless and automatic payment systems for parking, improving convenience for users.

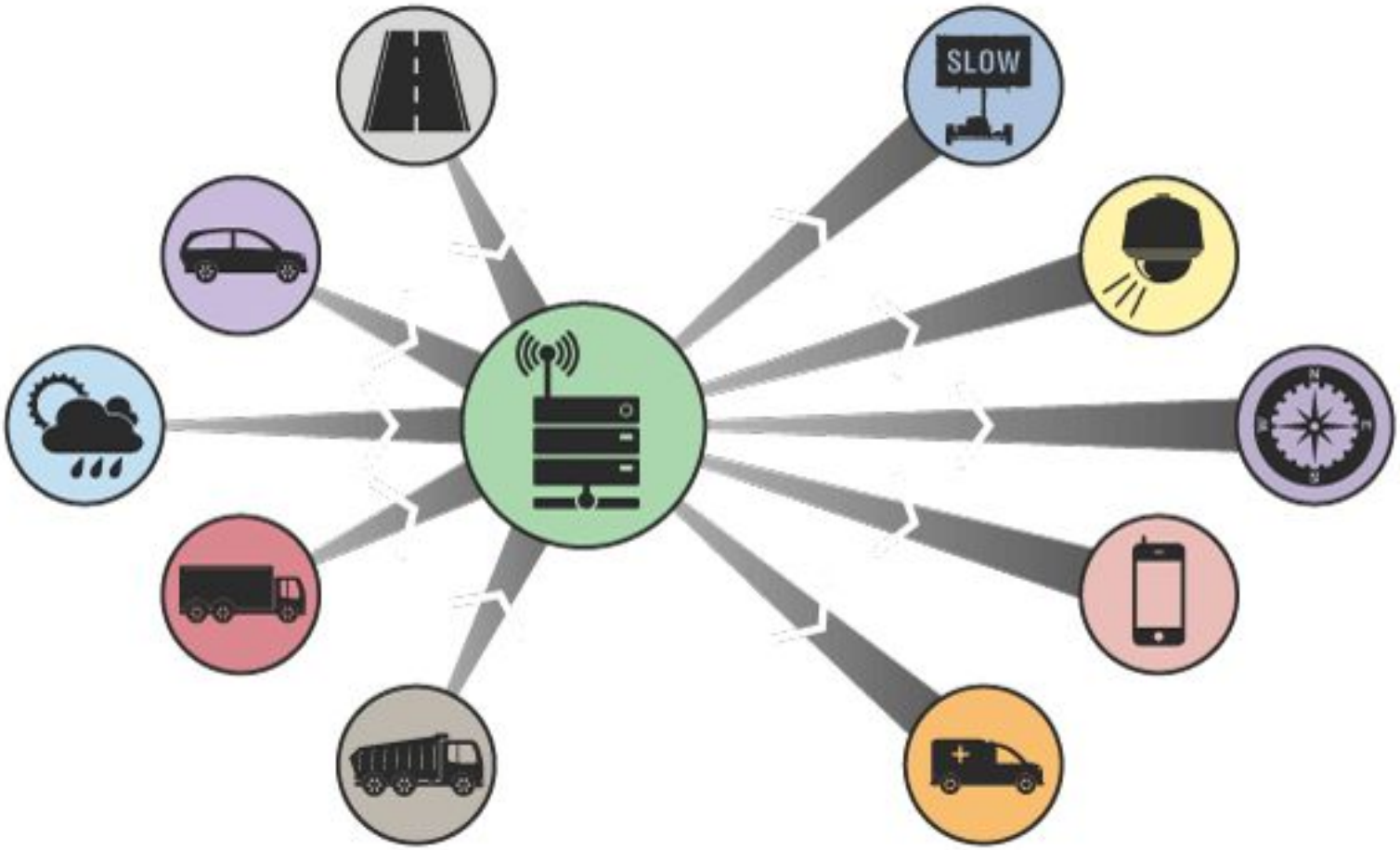
7. Connectivity and Communication:

- **Vehicle-to-Vehicle (V2V) Communication:** WSNs enable vehicles to communicate with each other, sharing information about speed, location, and other parameters. This helps prevent collisions and improve overall traffic flow.

- **Vehicle-to-Infrastructure (V2I) Communication:** Vehicles can communicate with traffic signals and road infrastructure to optimize traffic management and reduce congestion.

- WSNs are integral to the development of connected and autonomous vehicles (CAVs).

Case study – Applications of WSN in Transportation



Case study – Applications of WSN in Transportation

- **Freight Tracking and Cold Chain Monitoring:** WSNs are used to track the movement and conditions of cargo, especially in the case of perishable goods. Temperature and humidity sensors help maintain the quality of goods during transportation.

- **Border Security and Customs Management:** WSNs can enhance border security by monitoring vehicle movements and detecting unauthorized border crossings. They can also streamline customs processes through improved cargo tracking.

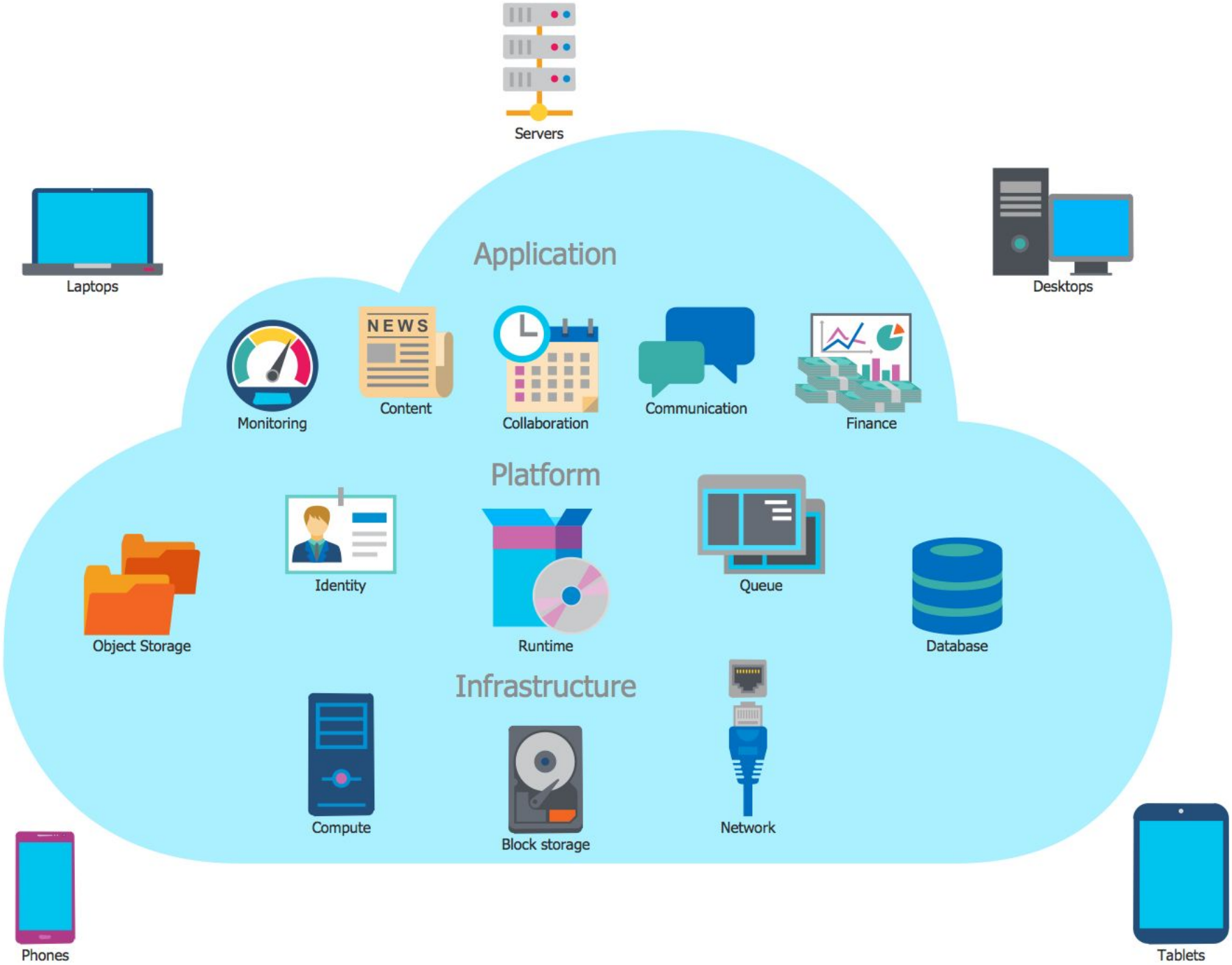
- **Toll Collection and Traffic Enforcement:** WSNs can be used in electronic toll collection systems, automatically deducting toll fees as vehicles pass through toll stations. They can also assist in enforcing traffic regulations through automatic license plate recognition.

Cloud Computing Basics

- **Cloud computing-** Is a collection of integrated and networked hardware, software and internet infrastructure (platform) that provide variety of services (computing/storage)to users and organizations over the internet.

- Provider -> Data center -> virtualizing center -> distributed center -> clients.

- **Roles and Components** of cloud computing –
- Cloud provider – makes service available, manages, configures, have IT resource
- Cloud consumer- paid user of service, hierarchy/roles/previlages of consumers.
- Cloud service owner- legal owner of service, provider or consumer.
- Resource admin- administrating a cloud based IT resource.
- Cloud auditor-Independent assessor of service,complience,performance,security.
- Cloud broker-negotiates between consumer-provider, supports consumers.
- Cloud carrier-provides connectivity and transport of service, internet Serv. P.


Servers
Laptops
Desktops
Application
Monitoring
NEWS
Content
Collaboration
Communication
Finance
Platform
Identity
Object Storage
Runtime
Queue
Database
Infrastructure
Compute
Block storage
Network
Phones
Tablets

Characteristics of cloud computing

- Cloud computing is characterized by several key features that distinguish it from traditional on-premises computing models.

- **On-Demand Self-Service:** Users can provision and manage computing resources, such as virtual machines, storage, and applications, without needing to interact with human administrators, reducing the need for manual intervention.

- **Broad Network Access:** Cloud services are accessible over the internet from a variety of devices, including laptops, smart phones, tablets, and desktop computers anywhere with an internet connection.

- **Resource Pooling:** Cloud providers pool computing resources to serve multiple users. Resources such as processing power, storage, and memory are dynamically allocated and reassigned based on demand ,enhances efficiency and scalability.

- **Rapid Elasticity:** Cloud resources can be quickly scaled up or down to accommodate changes in workload or demand. This elasticity enables users to handle traffic spikes or variations in resource requirements without manual intervention.

Characteristics of cloud computing

- **Measured Service:** Cloud usage is monitored, measured, and tracked by the cloud provider. Users are billed based on their consumption of resources, allowing for cost optimization and pay-as-you-go pricing models.

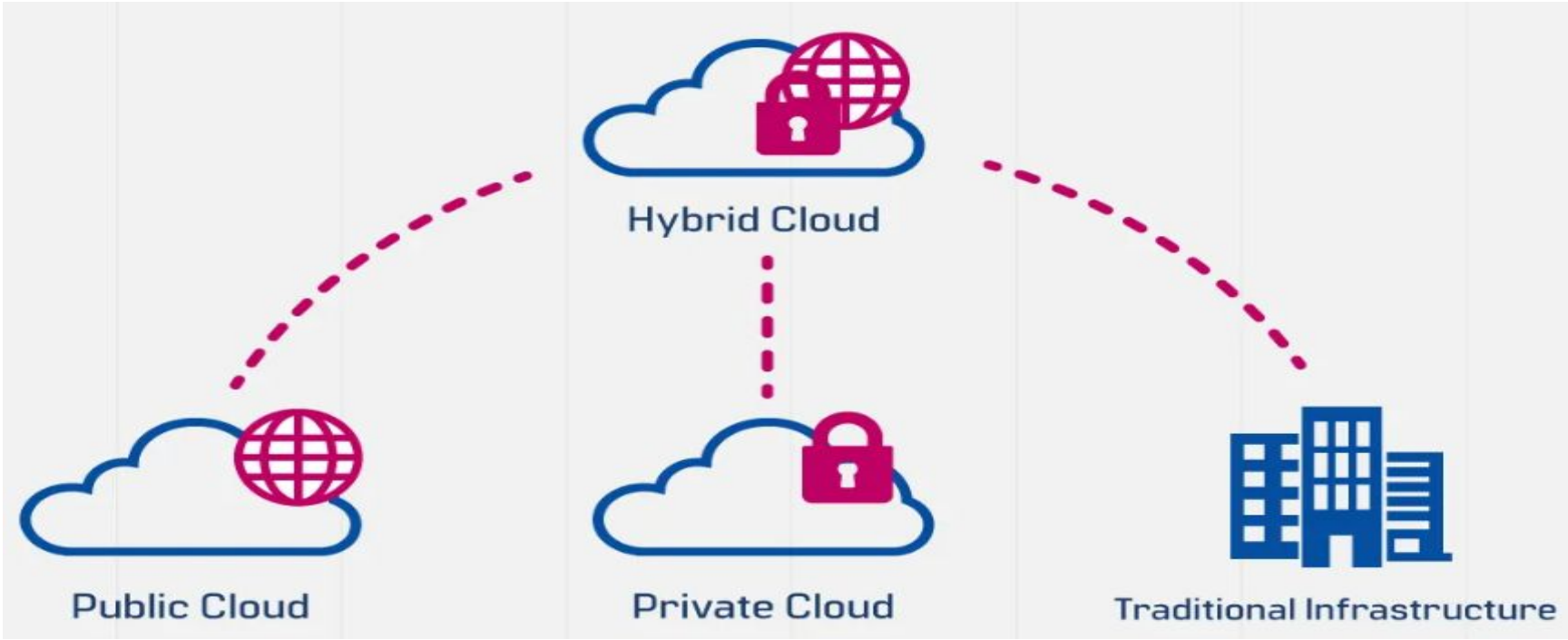
- **Multi-Tenancy:** Cloud resources are shared among multiple users or tenants, while maintaining isolation between them. Tenants may have different levels of access, security, and customization, but they share the underlying infrastructure.

- **Ubiquitous Access:** Cloud services are accessible from virtually anywhere with an internet connection, enabling users to work and collaborate remotely.

- **Resilience and Redundancy:** Cloud providers often offer redundancy and failover mechanisms to ensure high availability and minimize downtime in case of hardware failures or other issues.

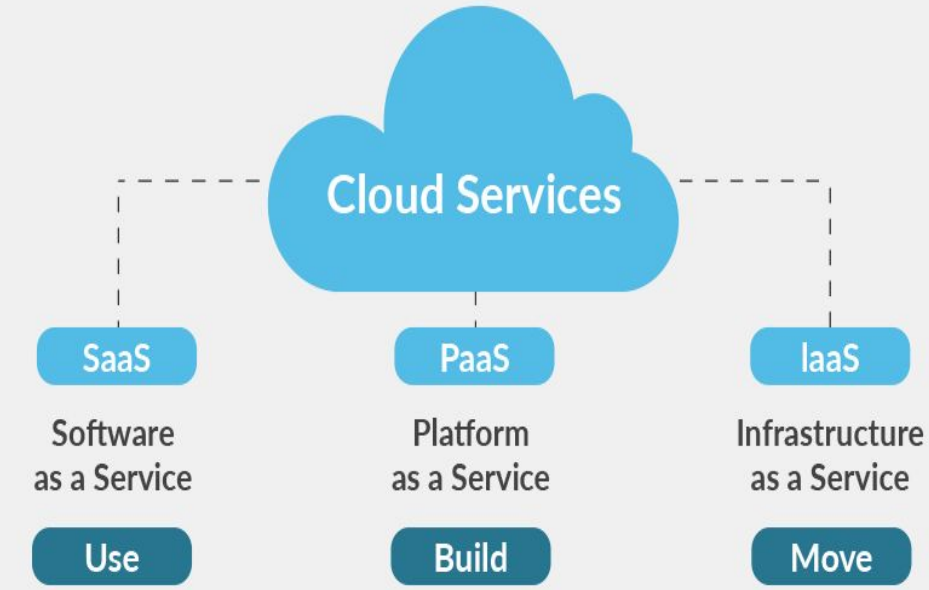
Deployment model of cloud computing

- Based on location and management of cloud's infrastructure.
- Services provided by third-party cloud providers to the general public over the internet are public cloud services.
- Services that are available to the organization's internal infrastructure is private cloud service. These are not available to any customer or subscriber.
- In a hybrid cloud environment , when the sensitive data is stored in private cloud but the app and resources are on the public cloud to be accessible for day to day communication.



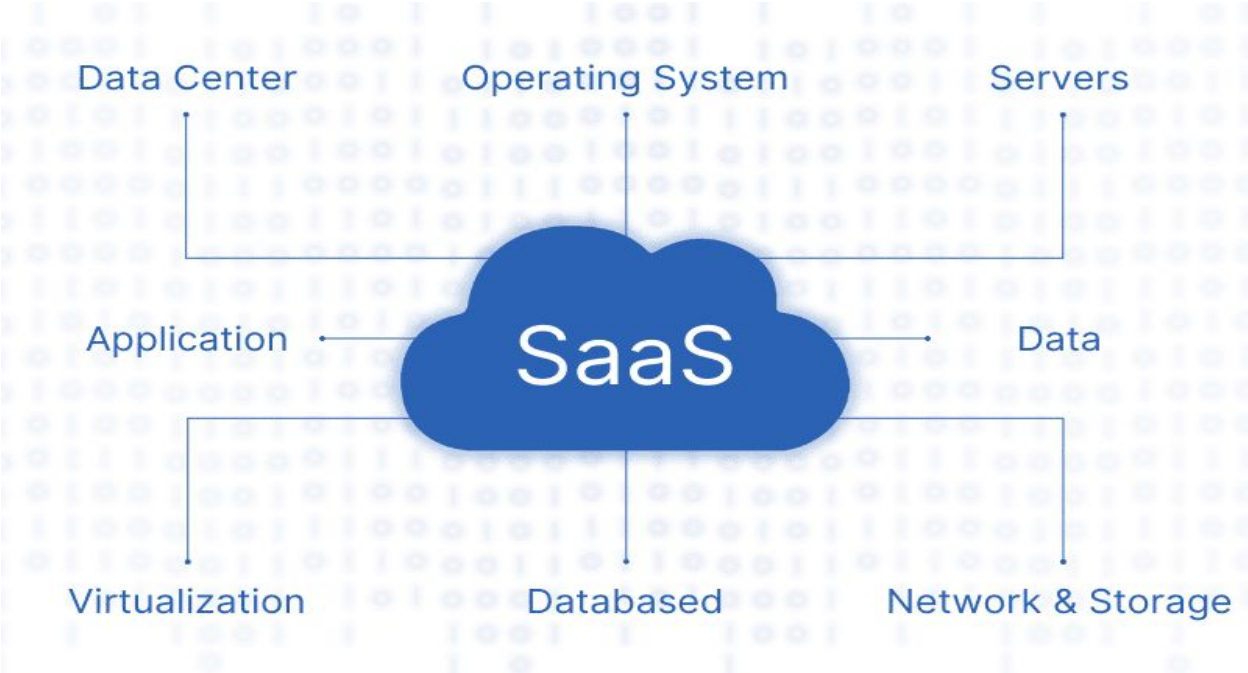
Service Models of cloud computing

- Based on the type of services that the provider is offering to the end client. The consumer is subscribing to the cloud platform to cater to the computing needs. On-demand, any-time, any-where service.
- Services are managed by cloud service provider and venders jointly. These services can be broad complexity range and be hierarchal in nature and interdependent.
- Three primary service models, catering to different levels of control and responsibility for users and organizations and determine the extent to which users manage their infrastructure, platforms, and applications.



Service Models - SaaS

- Software as a Service (SaaS):
- It delivers software applications over the internet on a subscription basis. Users access applications through web browsers, eliminating the need for local installation and maintenance. These applications are hosted and maintained by the cloud provider.
- In traditional model software is purchased and installed on PC.
- In this model the user client and consumer runs an application from cloud infrastructure through web browser. The complete application is offered as on demand, this saves the customer from buying software licenses and server front ends.



Service Models – SaaS - Characterstics

- **Accessibility:** SaaS applications are accessible from any device with an internet connection and a web browser, making it suitable for remote work and collaboration.

- **Subscription-Based:** SaaS is typically offered on a subscription basis, where users pay a recurring fee to access. This subscription model often includes updates, maintenance, and support.

- **Managed by Provider:** The cloud provider is responsible for managing the entire software stack, including infrastructure, hardware, software updates, security, and maintenance.

- **Multi-Tenancy:** SaaS applications are designed to serve multiple customers (tenants) using a shared infrastructure. Each customer's data and configuration are logically separated, ensuring data isolation and security.

- **Rapid Deployment:** Users can quickly start using SaaS applications without the need for lengthy installation processes. Rapid deployment accelerates time-to-value for users and businesses.

- **Reduced IT Overhead:** cloud provider handles infrastructure management, software updates, and security, users can focus on using the software and on their core business activities rather than managing IT operations.

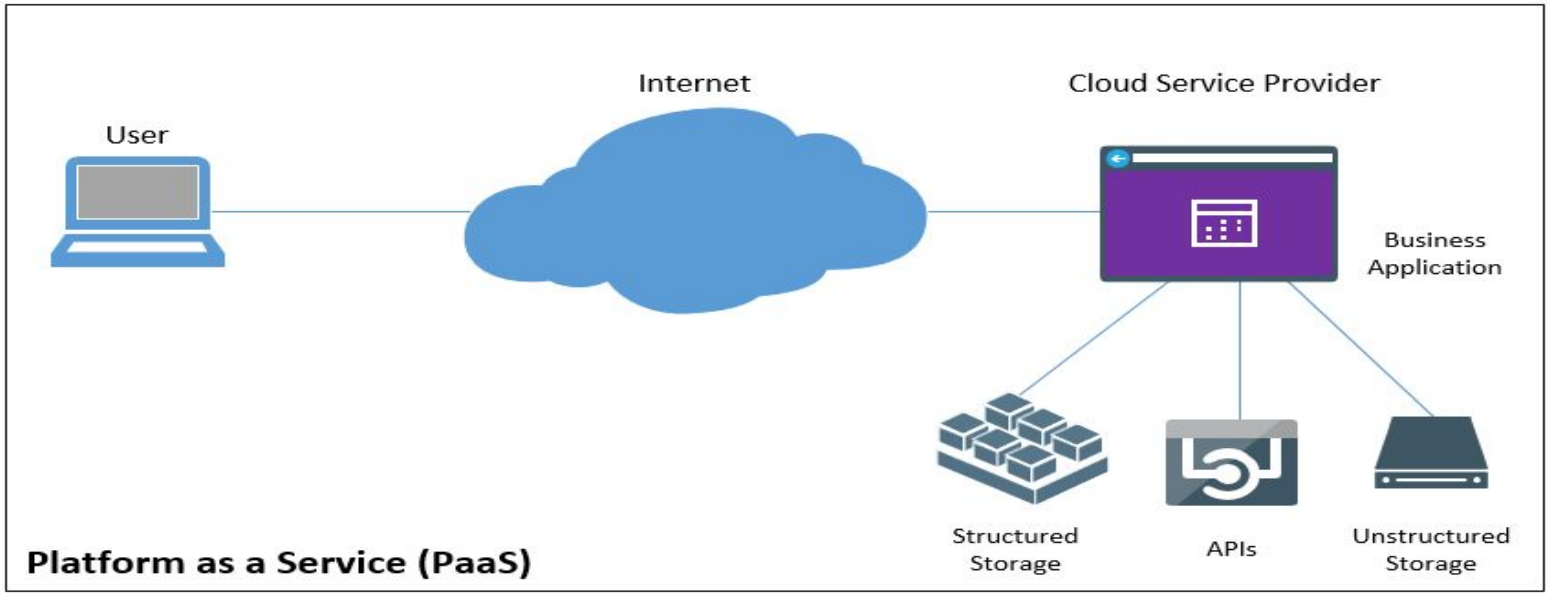
Examples of SaaS Applications:

- **Customer Relationship Management (CRM):** Salesforce, HubSpot, Microsoft Dynamics 365.
- **Productivity and Collaboration:** Google Workspace (formerly G Suite), Microsoft Office 365, Dropbox.
- **Enterprise Resource Planning (ERP):** Oracle NetSuite, SAP Business ByDesign.
- **Human Resources Management:** Workday, BambooHR, Zenefits.
- **Content Management Systems:** WordPress, Wix, Squarespace.
- **Video Conferencing and Communication:** Zoom, Microsoft Teams, Slack.
- **Financial Management:** QuickBooks Online, Xero, FreshBooks.


SaaS Providers
Microsoft office365
Google Apps
GoTo Metting
Constant Contact
Oracle CRM
NetSuite
Workday, Inc
Salesforce .com

Service Models - PaaS

- Platform as a Service (PaaS) is a cloud computing service model that provides a platform and environment for developers to build, deploy, and manage applications without the complexities of managing the underlying infrastructure.
- PaaS offers tools, services, and development frameworks that streamline the application development process and enable developers to focus on writing code and creating features, rather than managing servers or runtime environments.
- PaaS is related to SaaS but it does not provide an application, but a platform, supplies all the resources required to develop an application on web.
- It consist of browser based development studio, management and super visioning tools.



Service Models – PaaS - Characteristics

- **Development Frameworks:** PaaS offers pre-configured development frameworks and runtime environments that simplify the coding process. Developers can choose from a range of languages, libraries, and tools to build applications.

- **Deployment and Scaling:** PaaS platforms automate deployment and scaling processes, allowing applications to be easily deployed to production environments and scaled up or down based on demand.

- **Database and Storage:** PaaS provides managed database services, making it easy to set up, configure, and manage databases without dealing with the underlying infrastructure. It also offers storage solutions for managing application data.

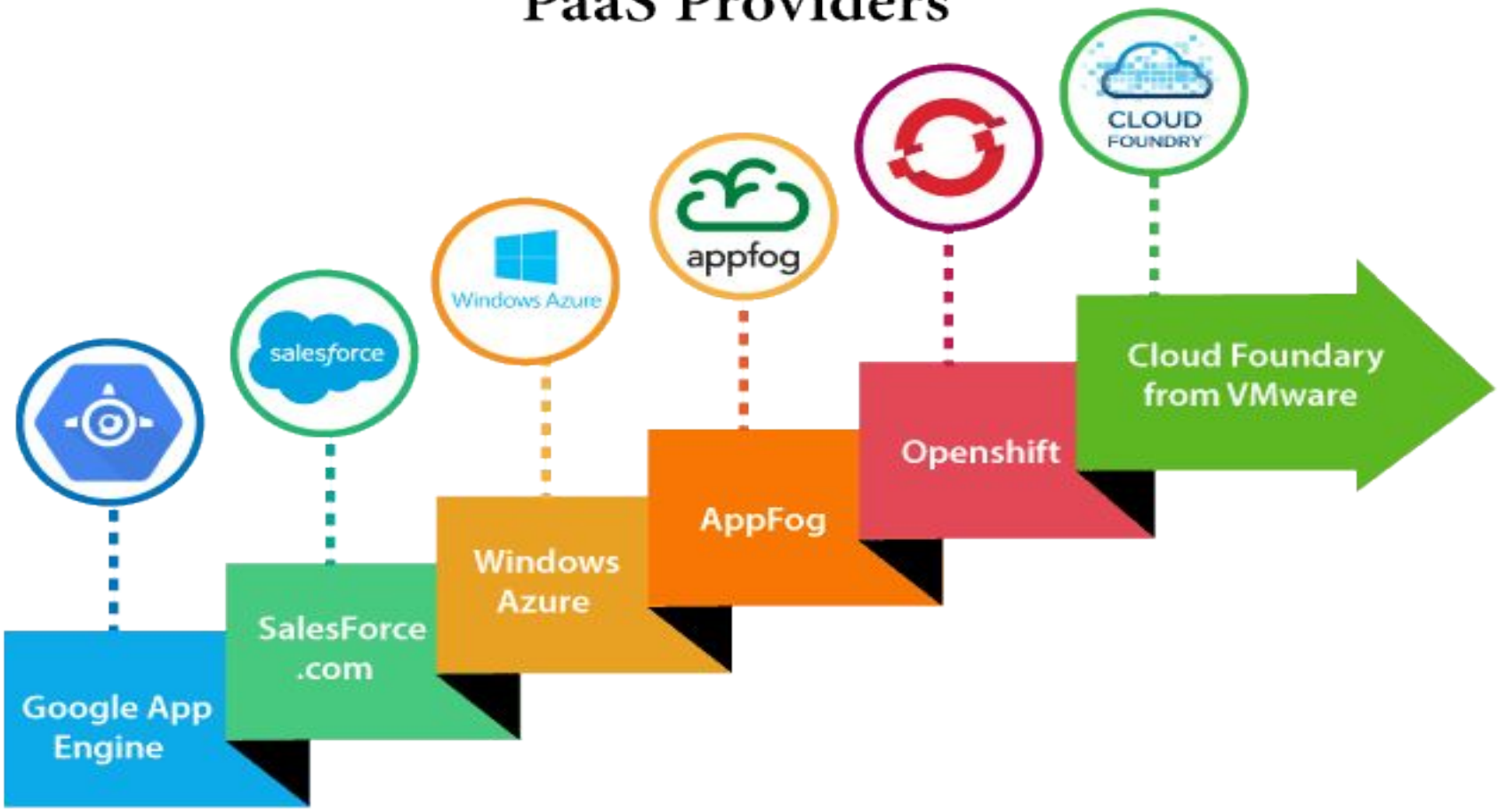
- **Middleware:** PaaS includes middleware services that facilitate communication between different components of an application. These services can include messaging, caching, and integration capabilities.

- **Vendor Lock-In:** While PaaS abstracts the underlying infrastructure, it may create some level of vendor lock-in due to the specific tools and services offered by the platform. Porting applications to another platform might require adjustments.

Examples of PaaS Applications:

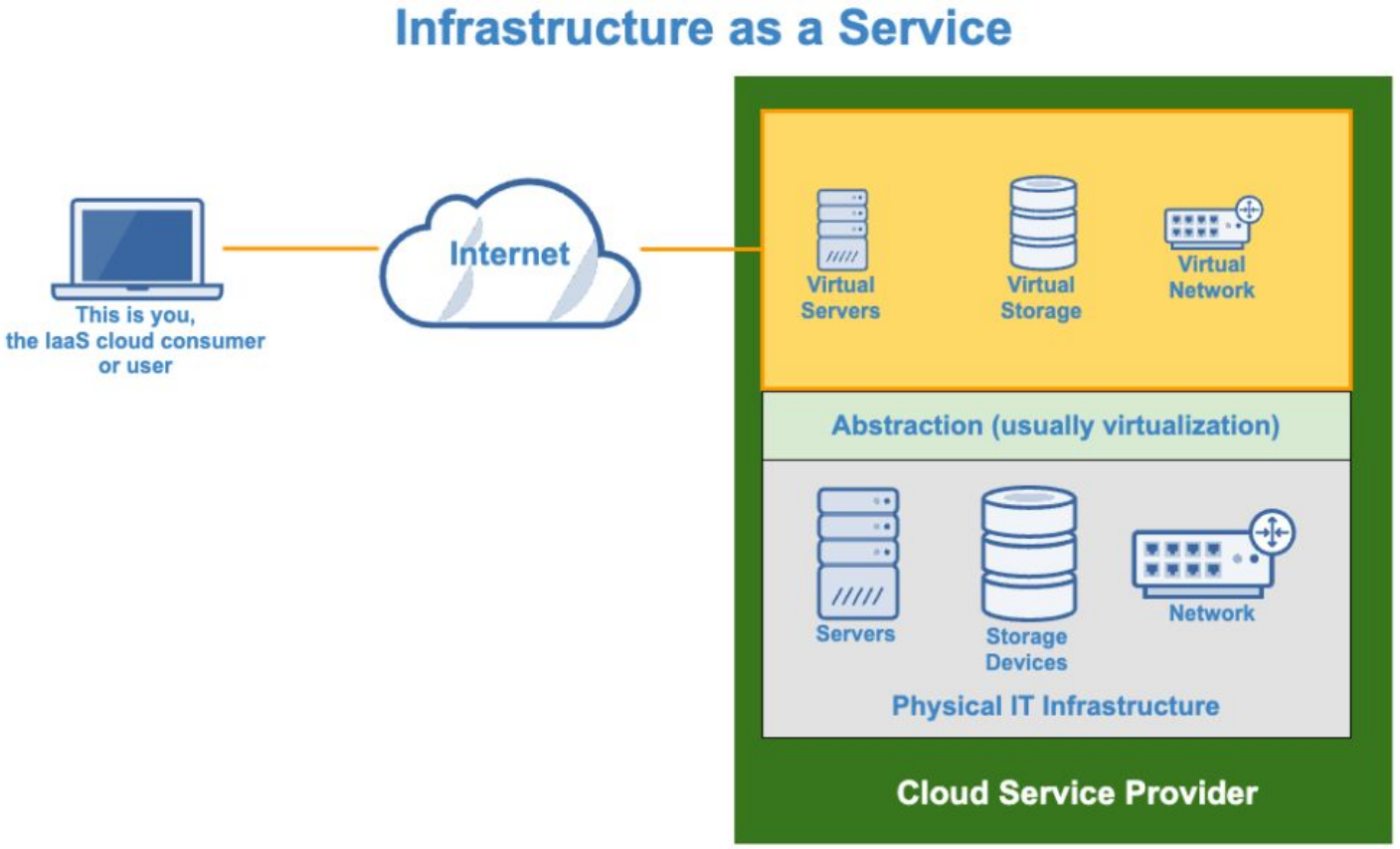
- **Google App Engine:** Offers a fully managed platform for building, deploying, and scaling applications using languages like Java, Python, and Go.
- **Microsoft Azure App Service:** Provides a platform for building and hosting web applications, APIs, and mobile backends with support for various programming languages and frameworks.
- **Heroku:** A cloud platform that simplifies deployment and management of applications by providing pre-configured environments and tools.
- **Red Hat OpenShift:** An open-source PaaS platform that enables developers to build, deploy, and manage applications across cloud environments.

PaaS Providers



Service Models - IaaS

- Infrastructure as a service -provides virtualized computing resources over the internet. With IaaS, users can rent and manage virtual machines, storage, and networking components without the need to invest in and maintain physical hardware and infrastructure.
- It offers hardware such as server space, routers, gateways, Memory, CPU, storage space which is managed by provider. Users have control over the operating systems, applications, and configurations of the virtual machines they deploy.



Service Models – IaaS - Characteristics

- **Virtualization:** IaaS uses virtualization technology to create virtual instances of computing resources, such as virtual machines (VMs) and storage. These virtual resources can be managed, configured, and used like physical hardware.

- **Resource Management:** Users have control over the allocation, configuration, and management of virtual machines and storage. They can choose operating systems, applications, and configurations according to their requirements.

- **Network Management:** IaaS platforms offer networking components such as virtual networks, subnets, load balancers, and firewalls, allowing users to create complex network configurations.
- **Storage Options:** including block storage, object storage, and file storage. Users can choose the storage type that best suits their application's needs.

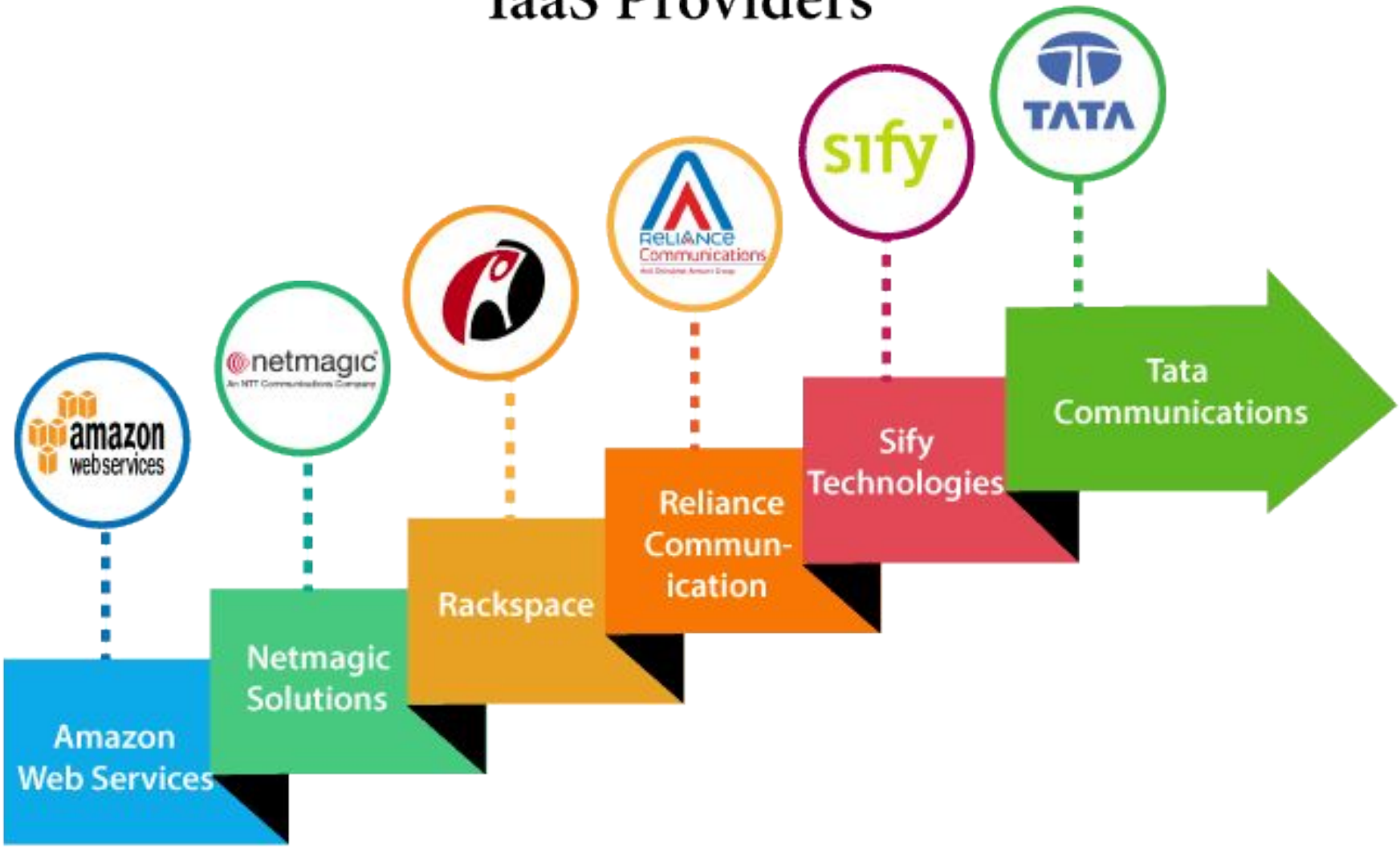
- **Security and Compliance:** features such as firewalls, encryption, and identity and access management (IAM) to help users secure their virtual resources and data.

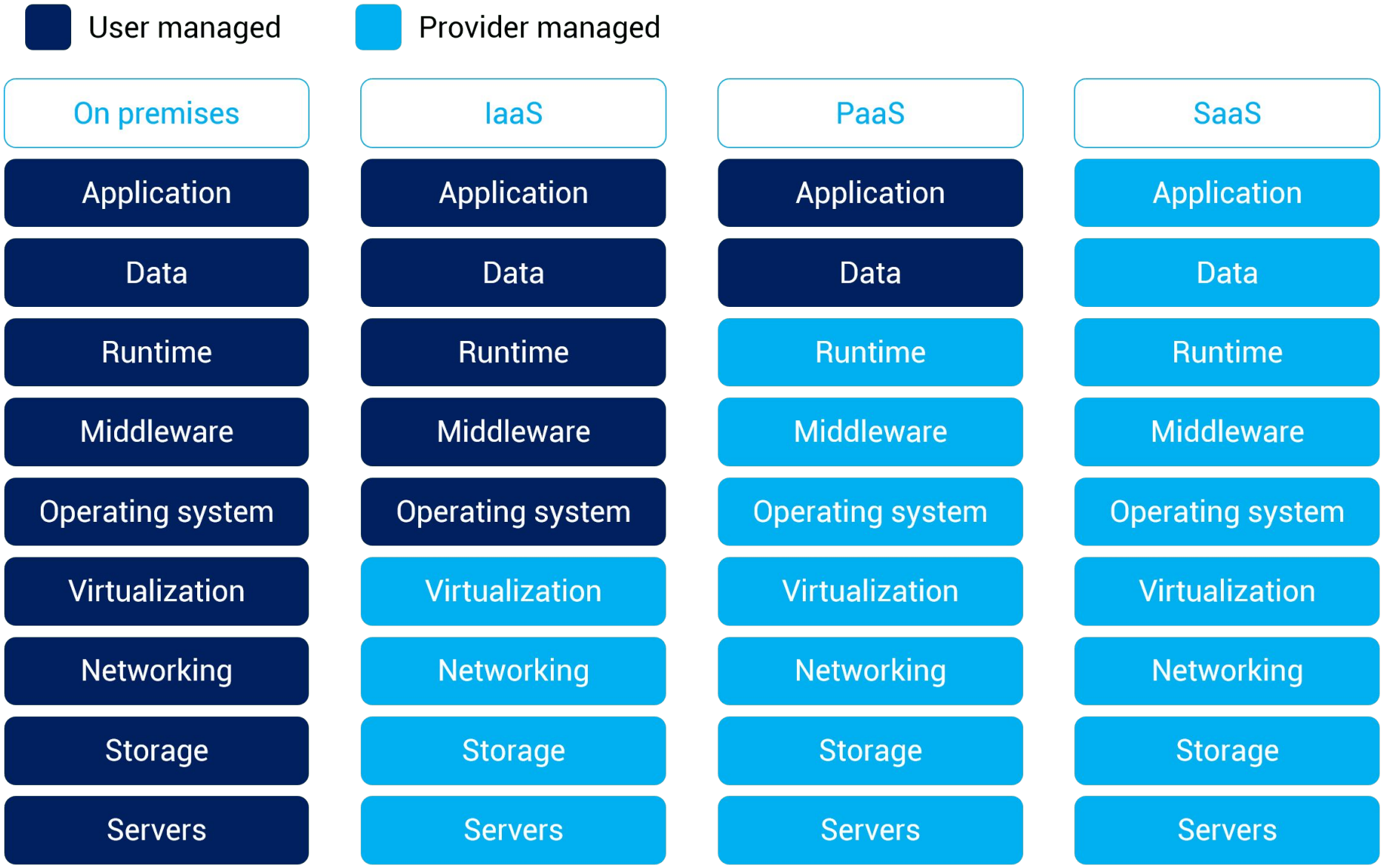
- **Backup and Disaster Recovery:** allowing users to create snapshots of their virtual machines and data for data protection and business continuity.

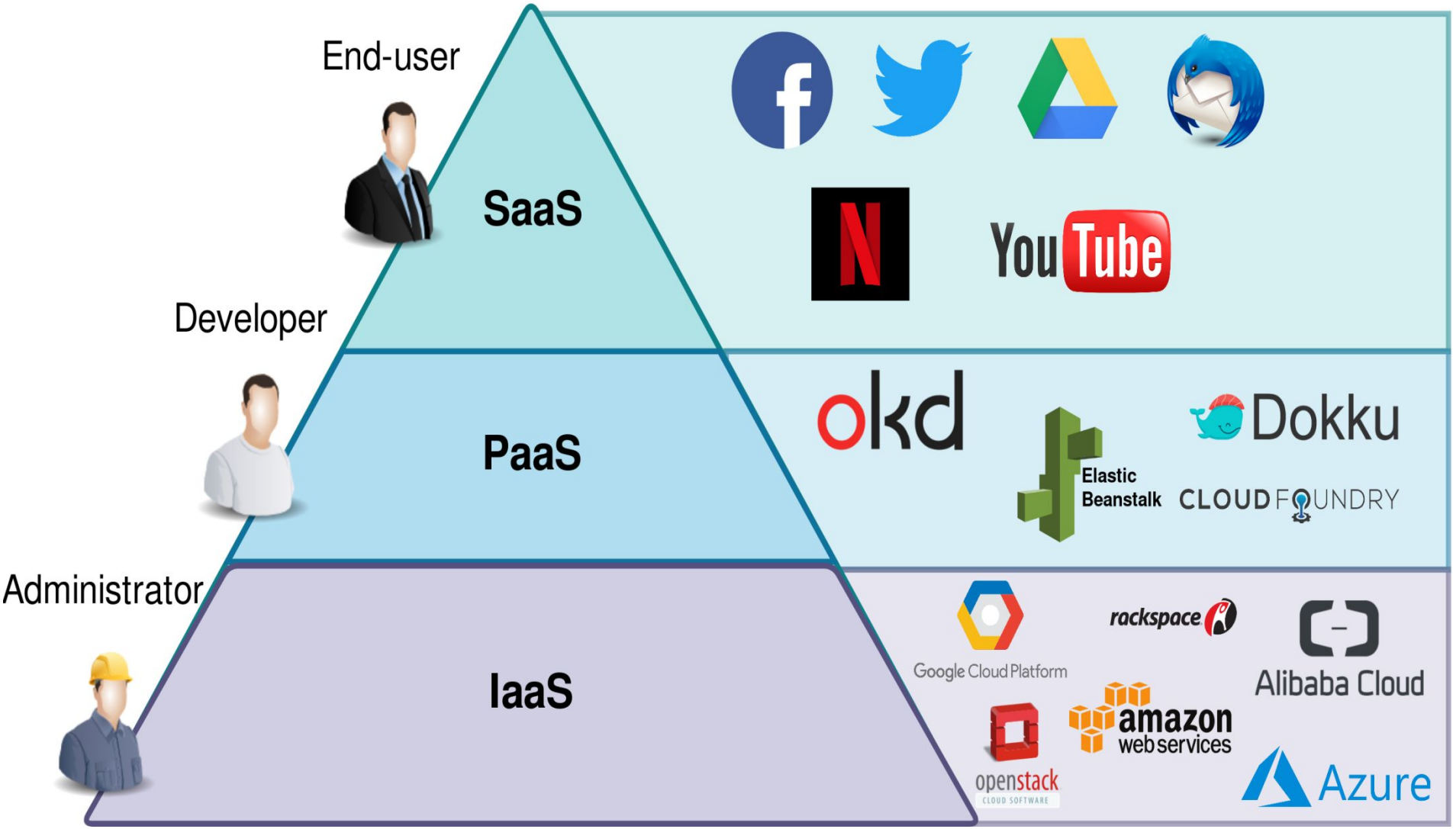
Examples of IaaS Applications:

- **Amazon Web Services (AWS) Elastic Compute Cloud (EC2):** Offers virtual machine instances in a variety of configurations.
- **Microsoft Azure Virtual Machines:** Provides scalable virtual machine options with Windows and Linux operating systems.
- **Google Cloud Compute Engine:** Offers virtual machine instances with flexible configuration options.
- **IBM Cloud Infrastructure:** Provides virtual servers, storage, and networking components on demand.

IaaS Providers




User managed
Provider managed
On premises
IaaS
PaaS
SaaS
Application
Data
Runtime
Middleware
Operating system
Virtualization
Networking
Storage
Servers


End-user
SaaS
Developer
PaaS
Administrator
IaaS
okd
Dokku
Elastic Beanstalk
CLOUD FOUNDRY
rackspace
Google Cloud Platform
Alibaba Cloud
amazon web services
openstack
CLOUD SOFTWARE
Azure

Cloud computing challenges

- **Security and Privacy:** service providers need to ensure the confidentiality, integrity, and availability of their data in the cloud. Data breaches, unauthorized access, and data loss are potential risks. Encrypting data, implementing strong access controls, and adhering to security best practices are crucial.

- **Data Location and Sovereignty:** Data stored in the cloud might be physically located in various jurisdictions, which can complicate compliance efforts and raise concerns about data sovereignty.

- **Vendor Lock-In:** Moving data and applications between cloud providers or back to on-premises infrastructure can be complex and costly. Vendor-specific APIs, services, and architecture might result in vendor lock-in, limiting flexibility.

- **Performance and Latency:** Cloud resources are shared among multiple users. Performance may vary due to resource contention, leading to latency and unpredictable performance.

Cloud computing challenges

- **Lack of Control:** Cloud services abstract infrastructure details from users, which can result in less control over underlying hardware, networking, and security configurations. This may pose challenges for organizations with specific requirements.

- **Sustainability and Environmental Impact:** The energy consumption of data centers that power cloud services can contribute to environmental concerns. Organizations need to consider the environmental impact of their cloud usage and evaluate providers' sustainability efforts.

- **Cost Management:** Monitoring usage, optimizing resource allocation, and selecting the right pricing models are essential to avoid unexpected expenses.

Development Environments for services