BITS Pilani
Pilani Campus

BITS Pilani presentation

Paramananda Barik
CS&IS Department

innovate achieve lead

BITS Pilani
Pilani Campus

SEZG586/SSZG586, Edge Computing

Lecture No.4 : 23rd Aug 2025

Agenda

- Shortcomings of Cloud for IoT
- Driving factors of Edge Computing
- Why do we need Edge Computing?
- Key Techniques that Enable Edge Computing
  - VMs and Containers
  - SDN
  - CDN
  - Cloudlets/Micro Datacenters
- Basic Attributes of Edge
- "CROSS" Value of Edge Computing
- Edge Computing Enables Industry Intelligence
- Edge Computing Benefits
- Edge Computing Systems

Shortcomings of Cloud for IoT

- a safety critical control system operating an industrial machine might need to stop immediately if a human is too close – Speed, Reliability
- a temperature sensor reports a 20ºC reading every second might not be interesting until the sensor reports a 40ºC reading - Cost
- autonomous vehicles or augmented reality applications need a response time below 20ms – Speed

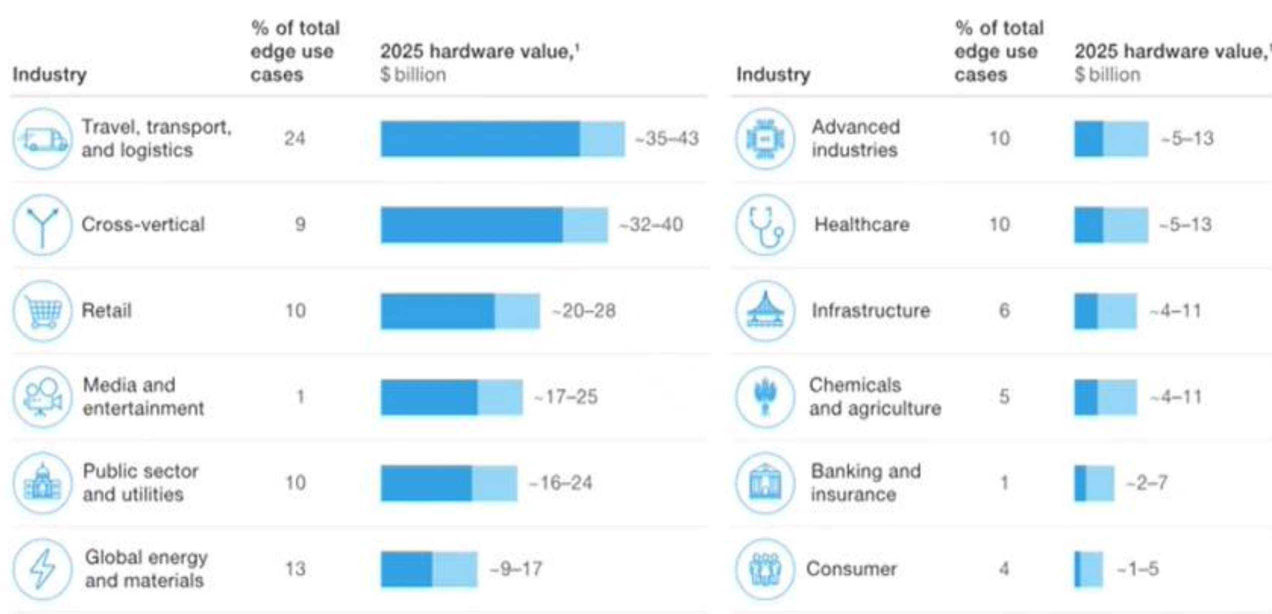
Need of the hour: IoT applications might require short response time, private data, and produce a large quantity of data needing large bandwidth

Study by McKinsey

Industries with **the most edge computing use cases are**

- Travel, transportation, and logistics
- Energy
- Retail
- Healthcare
- Utilities

Edge computing represents a potential value of $175 billion to $215 billion in hardware by 2025.



Industry	% of total edge use cases	2025 hardware value,[1] $ billion
Travel, transport, and logistics	24	~35–43
Cross-vertical	9	~32–40
Retail	10	~20–28
Media and entertainment	1	~17–25
Public sector and utilities	10	~16–24
Global energy and materials	13	~9–17
Advanced industries	10	~5–13
Healthcare	10	~5–13
Infrastructure	6	~4–11
Chemicals and agriculture	5	~4–11
Banking and insurance	1	~2–7
Consumer	4	~1–5

Total: ~$175 billion–$215 billion

Driving factors of Edge Computing

- Varied connectivity and data mobility
- Need for real-time decision making
- Localized compute power & storage

Characteristics of Edge Computing

- Proximity
- Ultra-low latency
- High bandwidth
- Reliability
- Real time access to radio network

Why do we need Edge Computing?

- Push(ed) from Cloud Services
  - Limited bandwidth over Internet
  - Response Time
  - Reliability of network
- Pull(ed) from Internet of Things
  - Enormous amount of Data generated by billions of devices
  - Leading to huge unnecessary bandwidth and computing resources usage
  - End devices in IoT are energy constraint things
  - Wireless communication module - drains battery

Why do we need Edge Computing?

- The end devices at the edge usually play as a data producer and consumer
- Example – Social networking platforms
  - Youtube – 72 hrs of content uploaded every single minute
  - Facebook – users share ~2.5 million pieces of content
  - Twitter – 300,000 tweets
  - Instagram – 220.000 new photos

Key Techniques that Enables Edge Computing

- VMs and Containers
- Software Defined Networking (SDN)
- Content Delivery/Distribution Network (CDN)
- Cloudlets and Micro Data Centers (MDC)

VMs and Containers

- What is Virtualization?
  - Virtualization is technology that lets users create useful IT services using resources that are **traditionally bound to hardware**.
  - It allows users to use a **physical machine's full capacity** by distributing its capabilities among many users or environments.

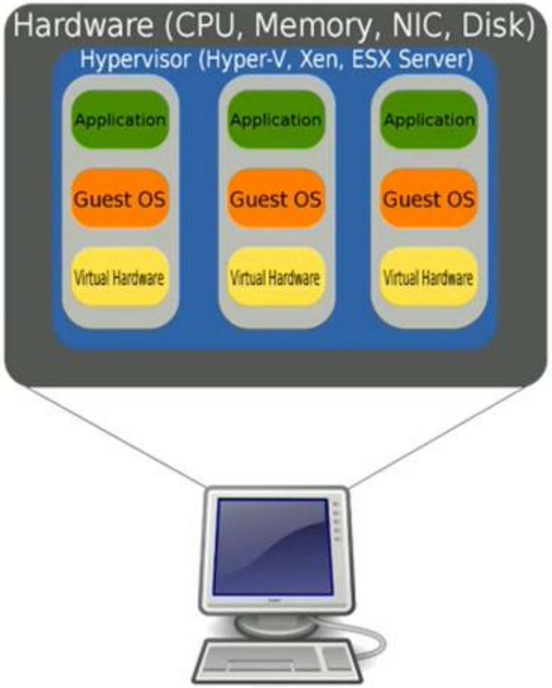
  - Virtualization and cloud computing are not interchangeable.
  - Virtualization is software that makes computing environments independent of physical infrastructure.

Before Virtualization

After Virtualization

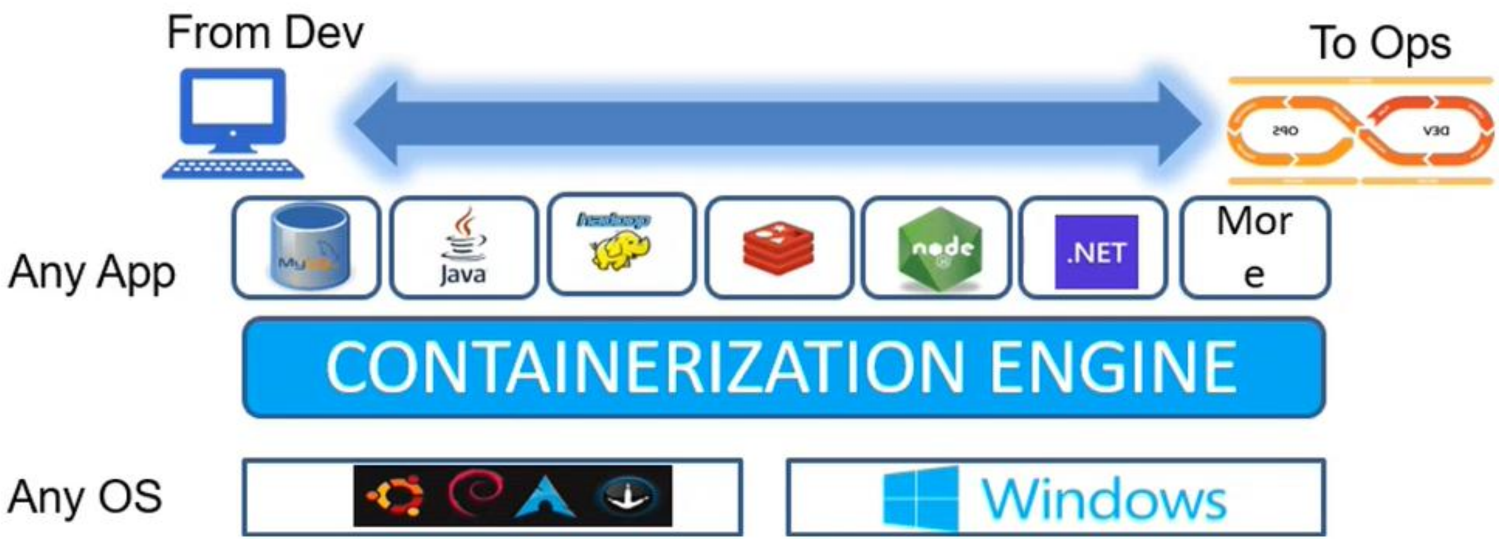
Virtual Machines

- A virtual computer system is known as a "virtual machine" (VM): a tightly isolated software container with an operating system and application inside.
- Each self-contained VM is completely independent.
- Putting multiple VMs on a single computer enables several operating systems and applications to run on just one physical server, or "host."

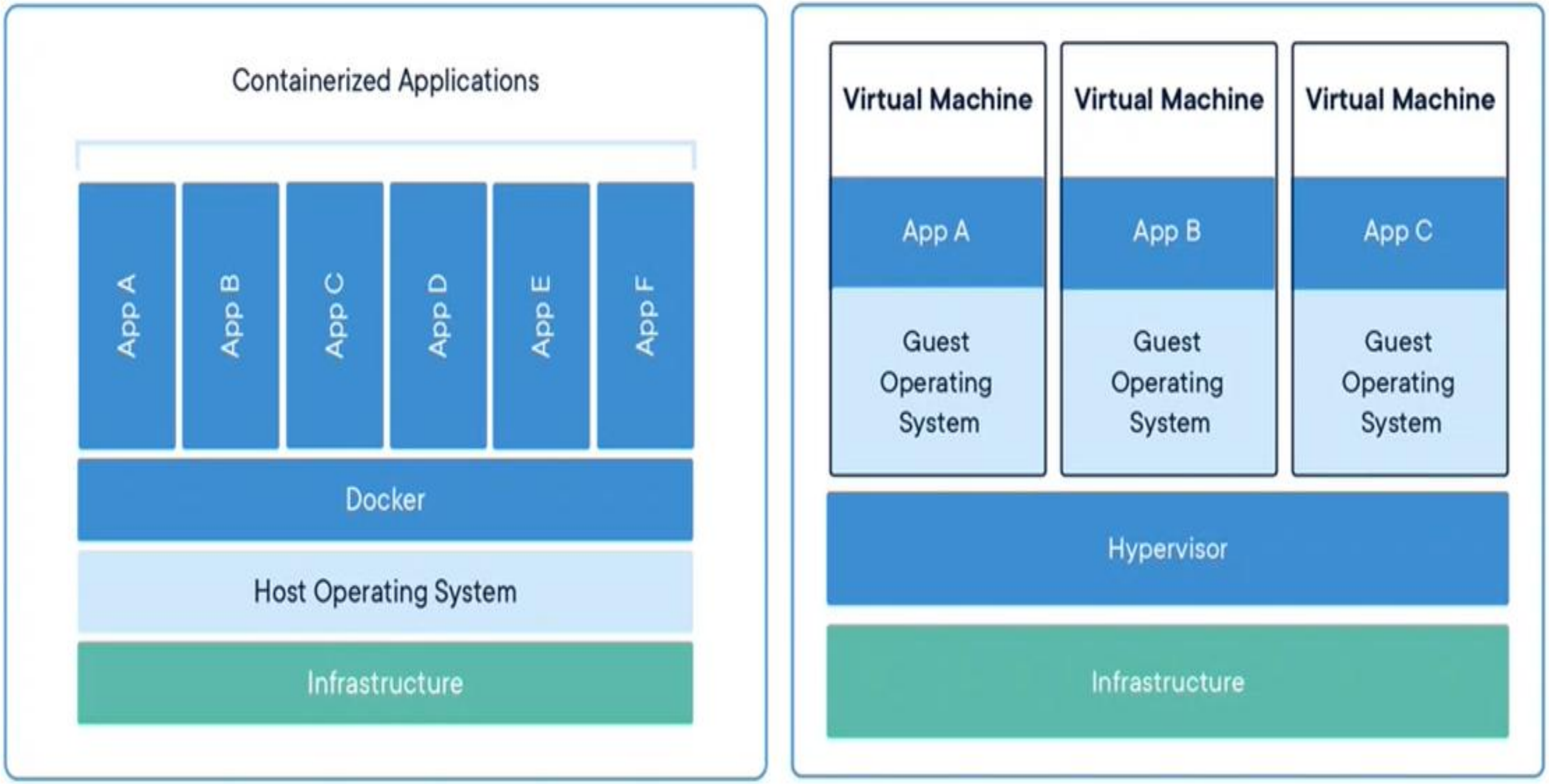


What are Containers?

- A software container is a standardized package of software.
- Everything needed for the software to run is inside the container.
- The software code, runtime, system tools, system libraries, and settings are all inside a single container



Virtual Machine vs Containers



Software Defined Networking (SDN)

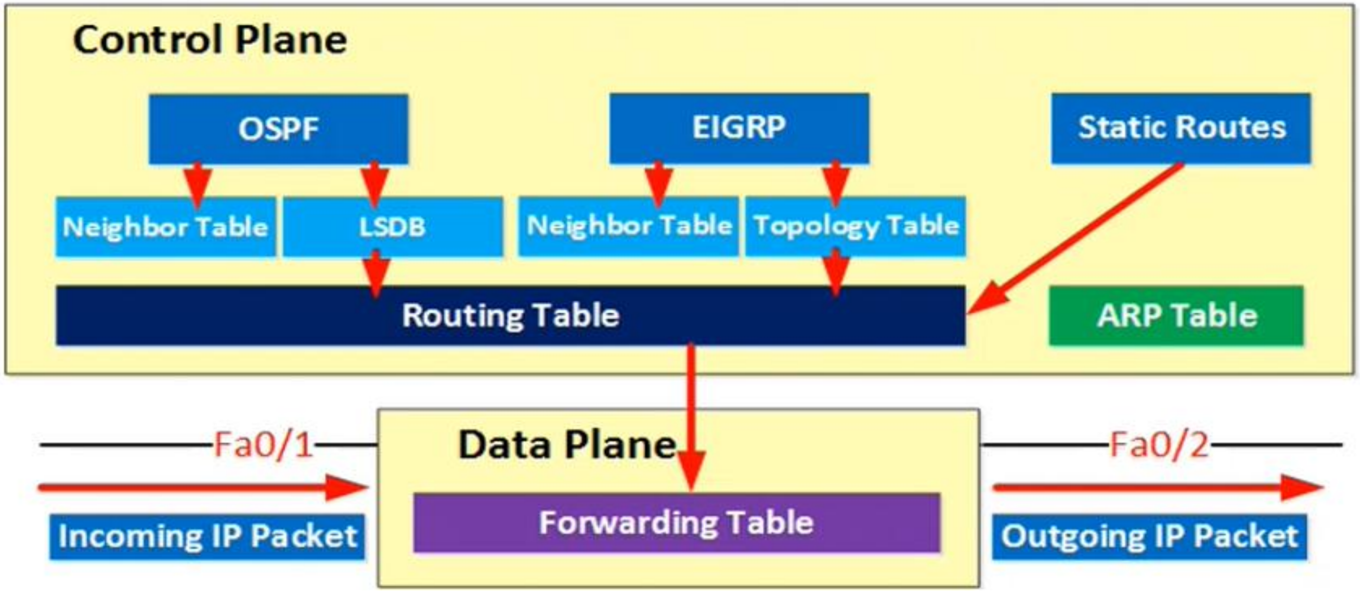
- Traditional Networking
  - Router functions
    - check the destination IP address in the routing table
    - Routing protocols like OSPF, EIGRP or BGP
    - use ARP to figure out the destination MAC address
    - Ethernet frame checksum recalculated

All these different tasks are separated by different **planes**. There are three planes:

•control plane
•data plane
•management plane

Different Planes

- Control Plane
- Data Plane
- Management Plane



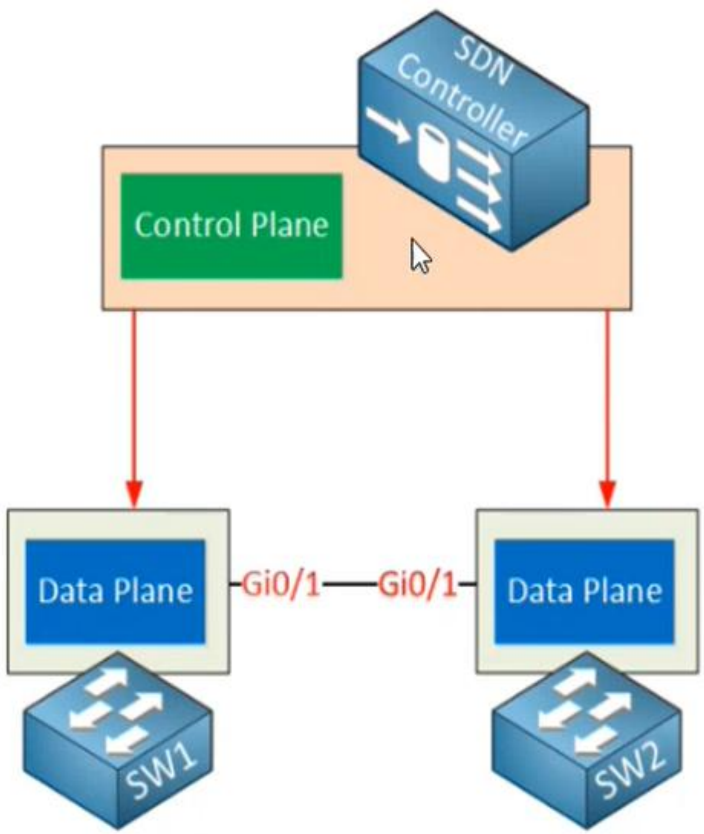
Limitations of traditional networks

- Configuration and re-configuration of network is SLOW, MANUAL process
  - VLANs have to be created on all switches
  - configure a root bridge for the new VLANs
  - assign four new subnets, one for each VLAN
  - create new sub-interfaces with IP addresses on the switches
  - configure VRRP or HSRP on the switches for the new VLANs
  - configure the firewalls to permit access to the new applications / subnets
  - advertise the new subnets in a routing protocol on our switches, routers, and firewalls

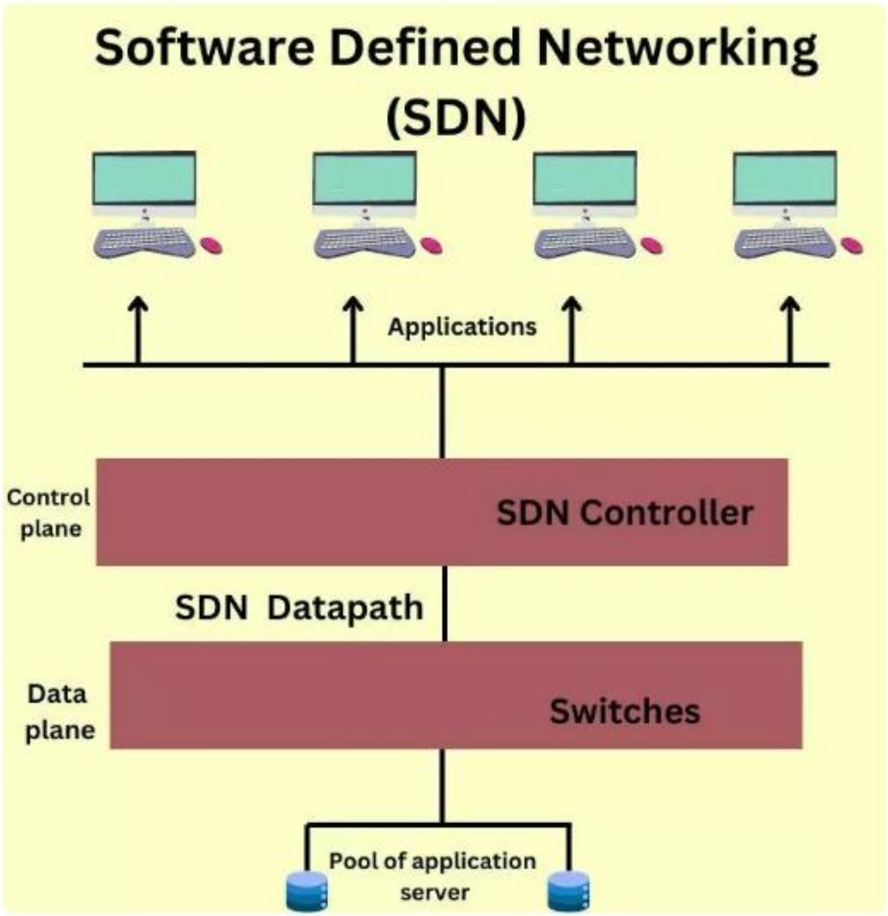
  - Might take hours to carry out these tasks in spite of having automation tools

Software Defined Networking (SDN)

- SDN controller - responsible for the activities done by control plane.
- The switches are now just “dumb” devices that only have a data plane, no control plane.
- The SDN controller is responsible for feeding the data plane of these switches with information from its control plane



SDN Architecture



SDN Architecture

1. **SDN Controller:** The brain of the SDN system, responsible for managing network traffic and making decisions.

2. **Data Plane:** The part of the network that forwards data based on instructions from the SDN controller.

3. **Control Plane:** Provides the instructions and policies to the data plane.

How does it help Edge Computing?

Edge computing pushes the computational infrastructure to the proximity of the data source, and the computing complexity will also increase correspondingly.

SDN provides a cost-effective solution for Edge network virtualization

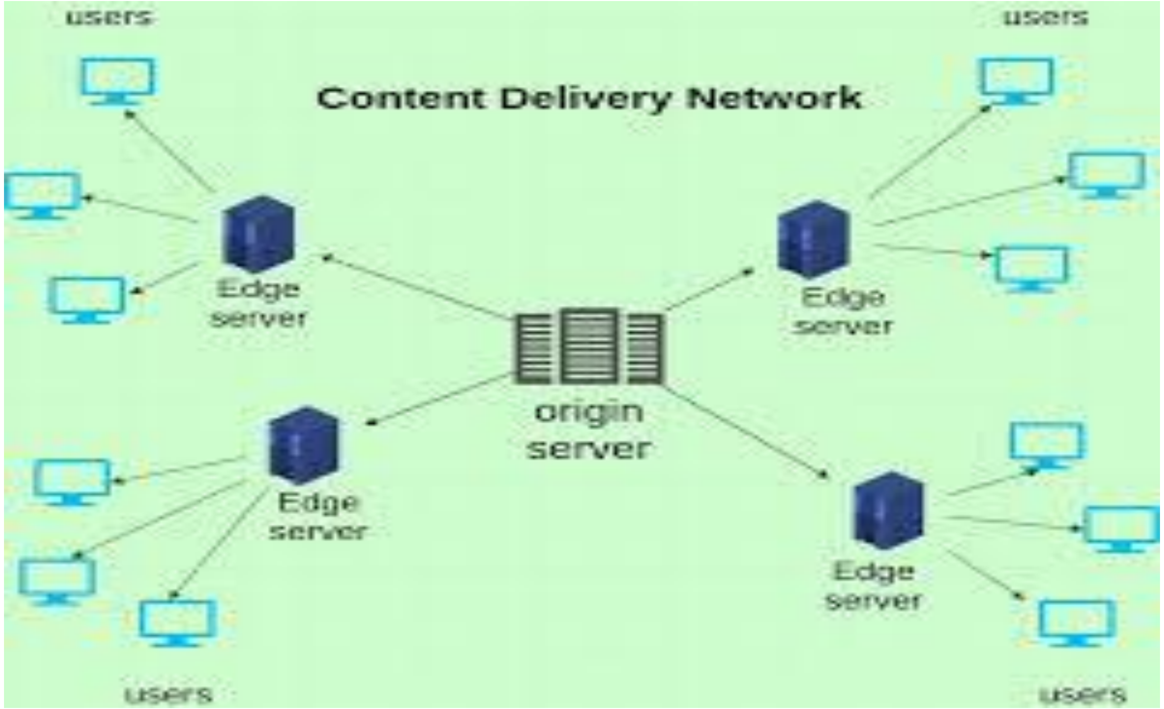
- Simplifies the network complexity by offering the automatic Edge device reconfiguration and bandwidth allocation.
- Edge devices could be set up and deployed in a plug-and-play manner enabled by SDN

Content Delivery/Distribution Network (CDN)

CDN is the concept of caching the content to the servers near the data consumers matches the system of Edge computing.

As the upstream server that delivers the content is becoming the bottleneck of the web due to the increasing web traffic, CDN can offer data caching at the Edge of the network with scalability and save both the bandwidth cost and page load time significantly.

Content Delivery/Distribution Network (CDN)



Cloudlets and Micro Data Centers (MDC)

Cloudlets and Microdata centers are the small-scale cloud data centers with mobility enhancement. They can be used as the gateway between Edge/mobile devices and the cloud. The computing power on the Cloudlets or MDCs could be accessed with lower latency by the Edge devices due to the geographical proximity.

Essential computing tasks for Edge computing such as speech recognition, language processing, machine learning, image processing, and augmented reality could be deployed on the Cloudlets or MDCs to reduce the resource cost.

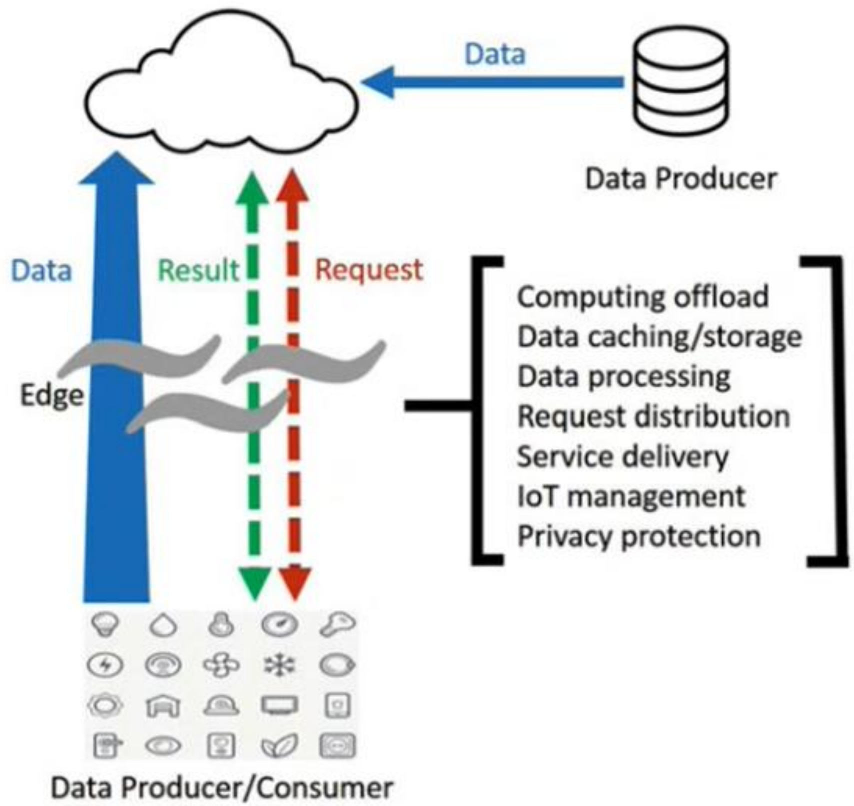
Edge Computing Definition

Edge computing refers to the enabling technologies allowing **computation** to be performed at the **edge of the network**, on *downstream data on behalf of cloud services* and *upstream data on behalf of IoT services*

- computing should happen at the proximity of data sources

Two-way computing streams

- Up stream
- Down stream



Basic Attributes of Edge

- Connectivity
- First Entry of Data
- Constraint
- Distribution
- Convergence

Basic Attributes of Edge

- Connectivity
  - Provide connection functions – Protocol support, deployment, management and maintenance
  - Research advancements - TSN, SDN, NFV, NaaS, WLAN, NB-IoT, and 5G
  - Interoperability with existing industrial buses

- First Entry of Data
  - Mass, real-time, diversity
  - Data management

Basic Attributes of Edge

- Constraint
  - Adaptability - harsh working conditions and operating environments
- Distribution
  - Support - distributed computing and storage, dynamic scheduling
- Convergence
  - Convergence of the Operational Technology (OT) and Information and Communications Technology (ICT)
  - Support collaboration in connection, data, management, control, application, and security.

“CROSS” Value of Edge Computing

Mass and Heterogeneous **C**onnection

- large number of connected devices
- heterogeneous Bus connections

Real-Time Services

- 10 ms

Data **O**ptimization

- large amount of heterogeneous data

“CROSS” Value of Edge Computing

Smart Applications

- intelligent applications

Security and Privacy Protection

- end-to-end protection
- data integrity and confidentiality

Edge Computing Enables Industry Intelligence – How?

- Connection – Physical and digital worlds
- Platform - Model driven, intelligent, distributed
- Collaborate - with cloud computing

Edge Computing Enables Industry Intelligence – How?

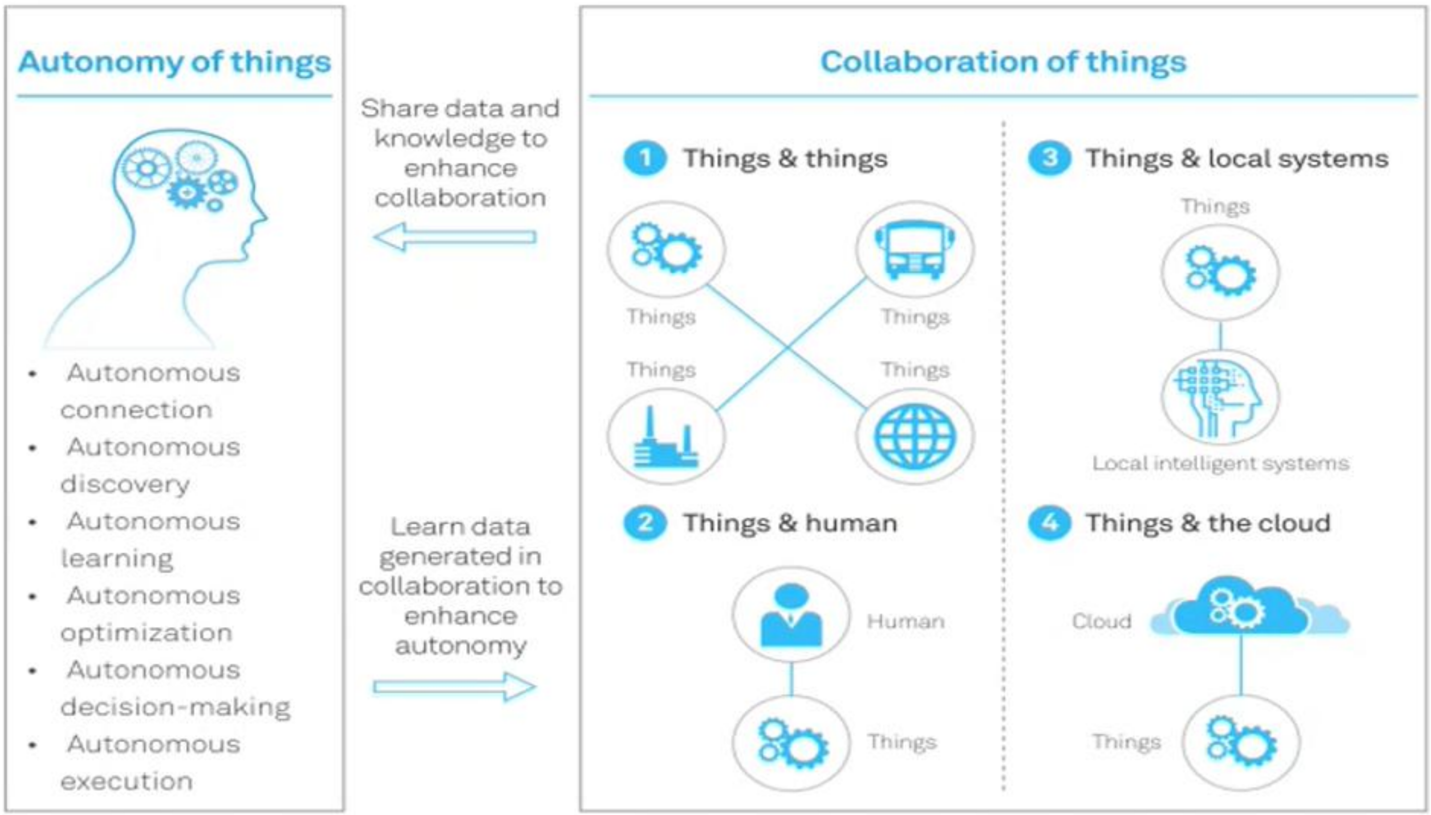
- Changes in the network field
  - Bandwidth increase – 1000 fold, Cost has decreased
- Changes in the computing field
  - Celeron – 6.40x10^3 MIPS, Xeon – 1.40x10^5 MIPS, A9 – 3.6x10^3 MIPS
- Changes in the storage field
  - Capacity increase – 10000 fold, cost has decreased
  - Speed also has increased

Connection – Physical and digital world

- Changes in the network field
  - Bandwidth increase – 1000 fold, Cost has decreased
- Changes in the computing field
  - Celeron – 6.40x10^3 MIPS, Xeon – 1.40x10^5 MIPS, A9 – 3.6x10^3 MIPS
- Changes in the storage field
  - Capacity increase – 10000 fold, cost has decreased
  - Speed also has increased

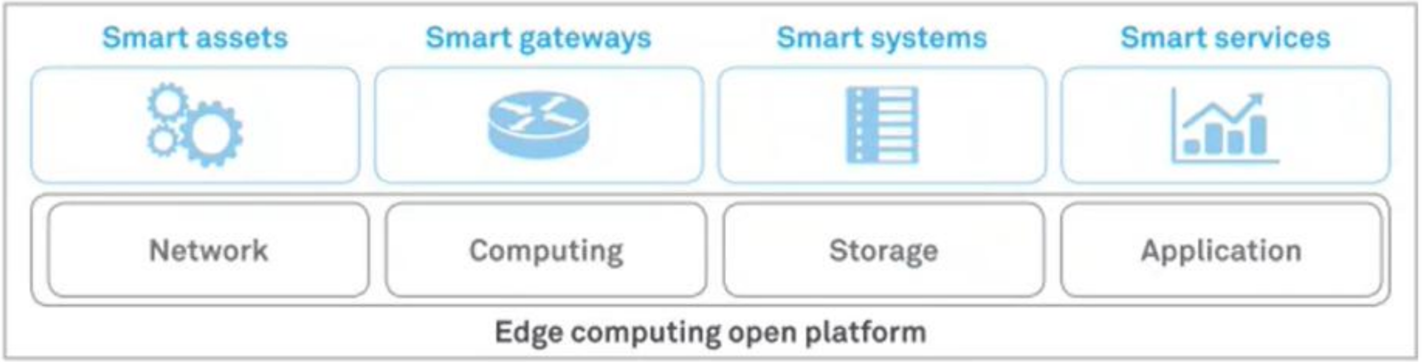
Platform – Model driven, intelligent, distributed

- Autonomy of things
- Collaboration of things



Intelligent distributed architecture

- Smart assets
- Smart gateways
- Smart systems
- Smart services

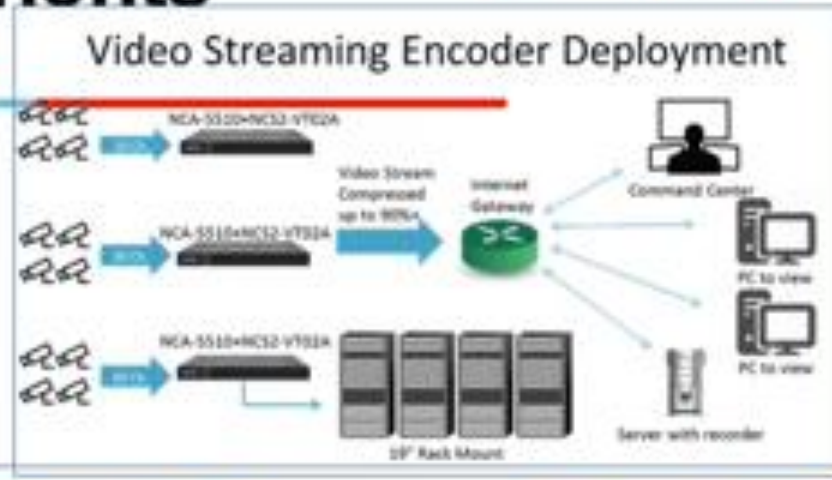


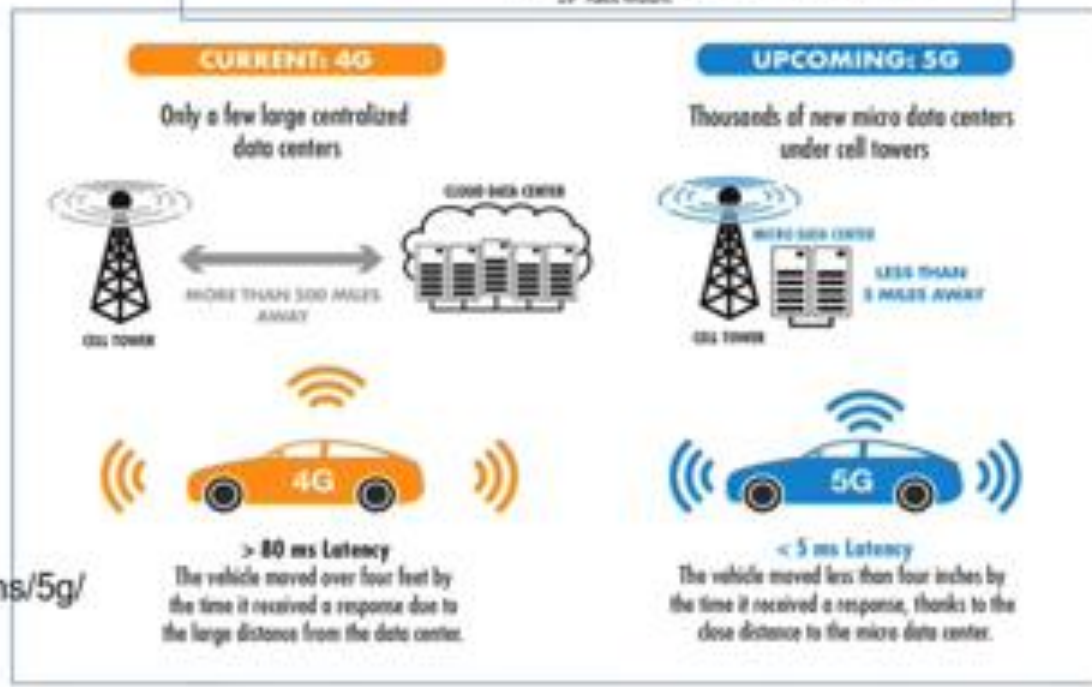
Collaboration of Edge Computing and Cloud Computing

Point of Collaboration	Edge Computing	Cloud Computing
Network	Data aggregation (TSN + OPCUA)	Data analysis
Service	Agent	Service orchestration
Application	Micro applications	Lifecycle management of applications
Intelligence	Distributed reasoning	Centralized training

Edge Computing Benefits

- Speed and Latency
- Security
- Cost Savings
- Greater Reliability
- Scalability





https://www.verizon.com/business/solutions/5g/edge-computing/industry-use-cases-examples/

Edge Computing Systems

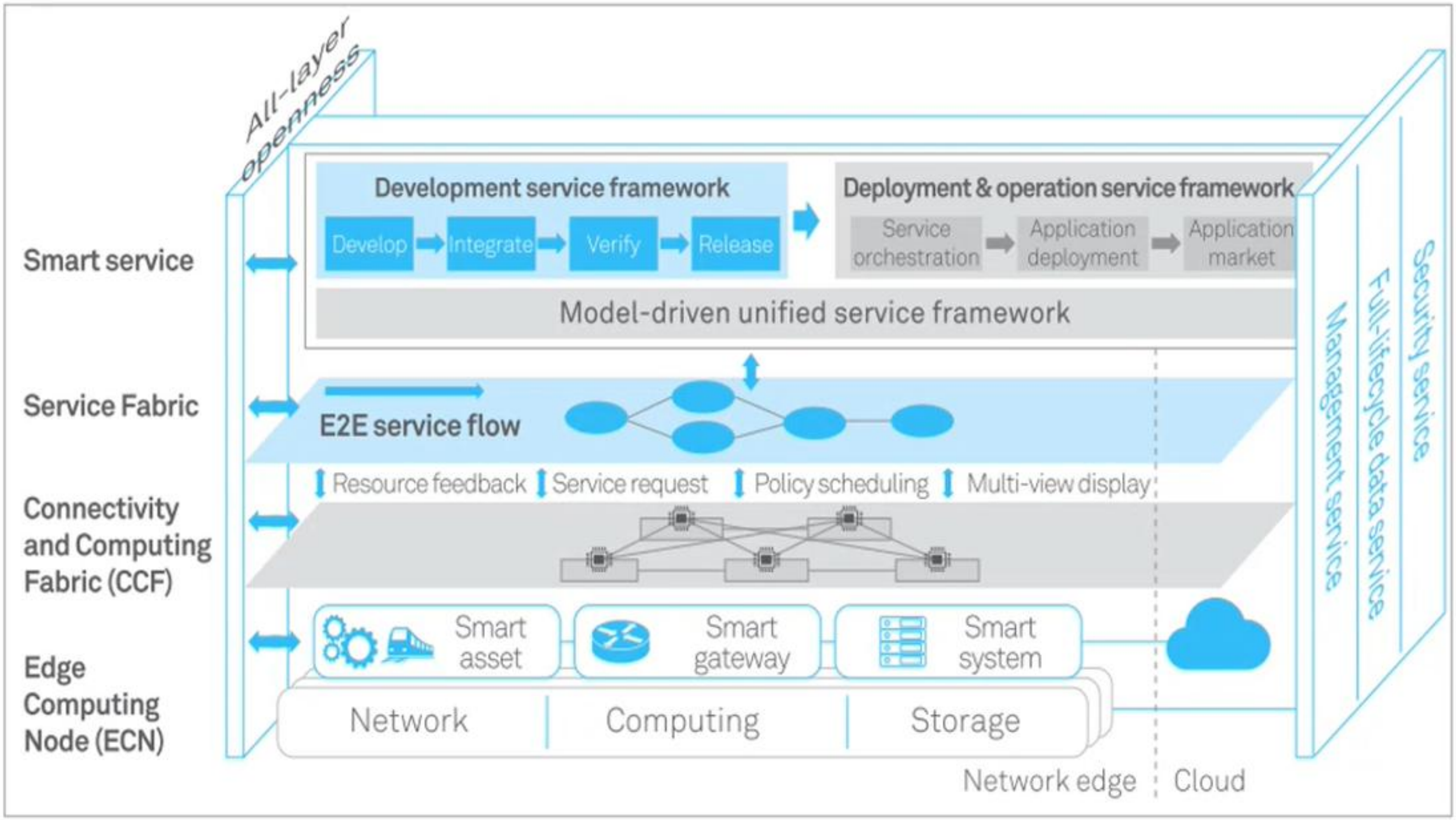
- Apache Edgent
- AWS Greengrass
- AWS Wavelength
- Azure IoT Edge
- Bosch IoT Edge
- EdgeX foundry

Edge Computing reference architecture 2.0

Model-Driven Engineering

- Coordination Between the Physical and Digital Worlds
- Cross-Industry Collaboration
- Reduced System Heterogeneity and Simplified Cross-Platform Migration
- Effective Support for System Lifecycle Activities

Edge Computing reference architecture 2.0



Edge Computing reference architecture 2.0

This Edge Computing Reference Architecture 2.0 provides a structured framework for managing **edge computing ecosystems**, ensuring:

- Efficient **service development, deployment, and operation**.
- Seamless **communication between edge and cloud**.
- Robust **security and management** capabilities.

Edge Computing reference architecture 2.0

1.Smart Service Layer

- This layer represents the high-level **services and applications** that run on the edge computing infrastructure.
- It includes:
  - **Development service framework**: A lifecycle process that includes:
    - **Develop**: Creating applications/services for edge computing.
    - **Integrate**: Merging various components.
    - **Verify**: Testing the functionality.
    - **Release**: Deploying the services.

Edge Computing reference architecture 2.0

- **Deployment & Operation service framework**: Manages:
  - **Service orchestration**: Coordinating multiple services.
  - **Application deployment**: Deploying applications efficiently.
  - **Application marketplace**: A repository for applications that can be deployed.

Edge Computing reference architecture 2.0

2. Service Fabric

- This layer enables seamless **End-to-End (E2E) service flow** and management.
- Key functionalities:
  - **Resource feedback**: Monitoring and optimizing resource usage.
  - **Service request**: Handling incoming service demands.
  - **Policy scheduling**: Implementing rules for resource allocation.
  - **Multi-view display**: Providing visibility into system performance and resource utilization.

Edge Computing reference architecture 2.0

3. Connectivity and Computing Fabric (CCF)

- This layer ensures **communication and computation** between different edge nodes and the cloud.
- It provides:
  - **Networking**: Connecting edge nodes, cloud services, and devices.
  - **Computing resources**: Processing data at the edge.
  - **Storage capabilities**: Storing relevant data closer to the user.

Edge Computing reference architecture 2.0

4. Edge Computing Node (ECN)

- The core infrastructure responsible for executing edge computing tasks.
- It consists of:
  - **Smart assets**: Devices generating data (e.g., IoT sensors, industrial machines).
  - **Smart gateways**: Devices handling data transfer between edge devices and the cloud.
  - **Smart systems**: Intelligent processing units that execute tasks at the edge.

Edge Computing reference architecture 2.0

5. Management and Security Services

- The architecture supports **all-layer openness**, meaning it allows integration with various external and cloud services.
- **Security services**: Protecting edge computing environments.
- **Full lifecycle data services**: Managing data from generation to analysis.
- **Management services**: Ensuring smooth operation and maintenance of edge resources.

Edge Computing reference architecture 2.0

6. Relationship with Cloud

- The architecture extends from the **network edge** to the **cloud**, ensuring hybrid processing.
- Some tasks are handled locally on the **Edge Computing Nodes (ECN)**, while others are processed in the cloud for **deeper insights**.

Multi-View Display

- Concept View
  - Domain models and key concepts of edge computing
- Function View
  - Functions and design concepts
- Deployment View
  - System deployment process

Multi-View Display

Multi-View Display in Edge Computing

Edge computing involves distributed processing across multiple edge nodes, requiring different perspectives to design, analyze, and deploy edge-based solutions efficiently. The **Multi-View Display** approach helps in understanding and managing edge computing systems from different perspectives.

Multi-View Display

Three Views of Multi-View Display

1. Concept View

1. Focuses on **domain models and key concepts** of edge computing.
2. Defines **fundamental principles**, including:
    1. Edge nodes and cloud integration.
    2. Latency-sensitive applications.
    3. Decentralized computing models.
3. Helps in designing the **overall architecture** before implementation.

Multi-View Display

2.Function View

- Represents **functions and design concepts** related to edge computing.
- Includes:
    - **Data processing functions** at the edge.
    - **AI and analytics** at the edge for real-time insights.
    - **Communication models** (e.g., edge-to-cloud, edge-to-edge).
- Essential for optimizing edge computing operations.

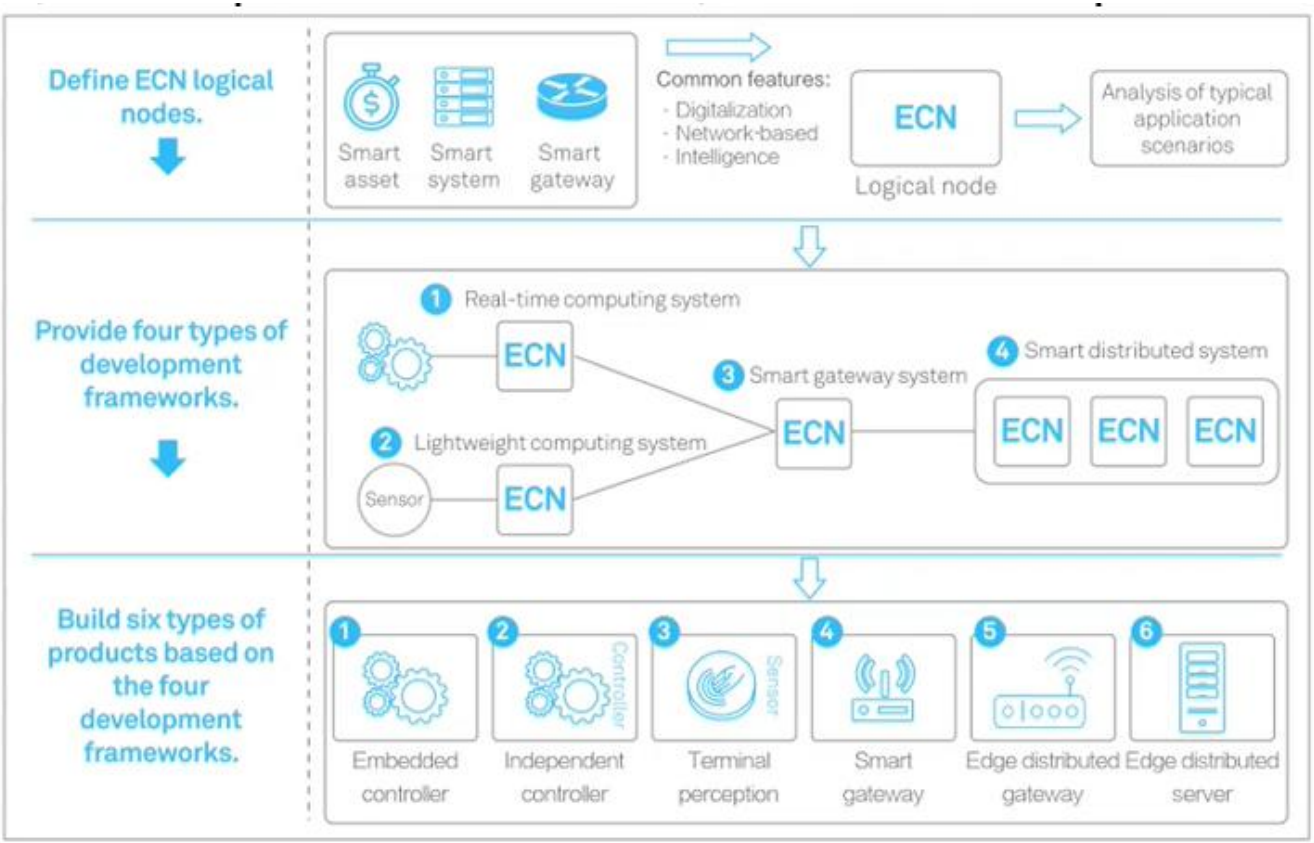
Multi-View Display

3.Deployment View

- Describes the **system deployment process** for edge computing.
- Covers:
  - Edge computing **infrastructure setup**.
  - **Orchestration of workloads** across edge devices.
  - **Integration with cloud and on-premise systems**.
- Ensures a **seamless transition** from design to practical deployment.

Multi-View Display

ECNs, Development Frameworks, and Product Implementation



Multi-View Display

Multi-View Display focusing on **Edge Computing Nodes (ECNs), Development Frameworks, and Product Implementation**. Below is an explanation of each section:

Multi-View Display

1.Defining ECN Logical Nodes

- ECNs (Edge Computing Nodes) represent **logical computing units** within an edge computing architecture.
- ECNs can be categorized into:
  - **Smart assets**
  - **Smart systems**
  - **Smart gateways**
- Key features of ECNs include:
  - **Digitalization** – Processing data at the edge.
  - **Network-based intelligence** – ECNs enable smart decision-making.
- The framework allows for **typical application scenario analysis**, which helps determine where and how ECNs should be deployed.

Multi-View Display

2.Four Types of Development Frameworks

The Four **key development frameworks** for ECNs:

1. **Real-time computing system** – Handles **time-sensitive tasks** with **low-latency processing**.
2. **Lightweight computing system** – Works with sensors and **low-power devices** for **basic computing needs**.
3. **Smart gateway system** – Bridges **local edge nodes** and the **cloud**, enabling **seamless data flow**.
4. **Smart distributed system** – Connects multiple ECNs to create **a decentralized and distributed edge network**.

Each of these frameworks is designed to accommodate different **application needs** in edge computing.

Multi-View Display

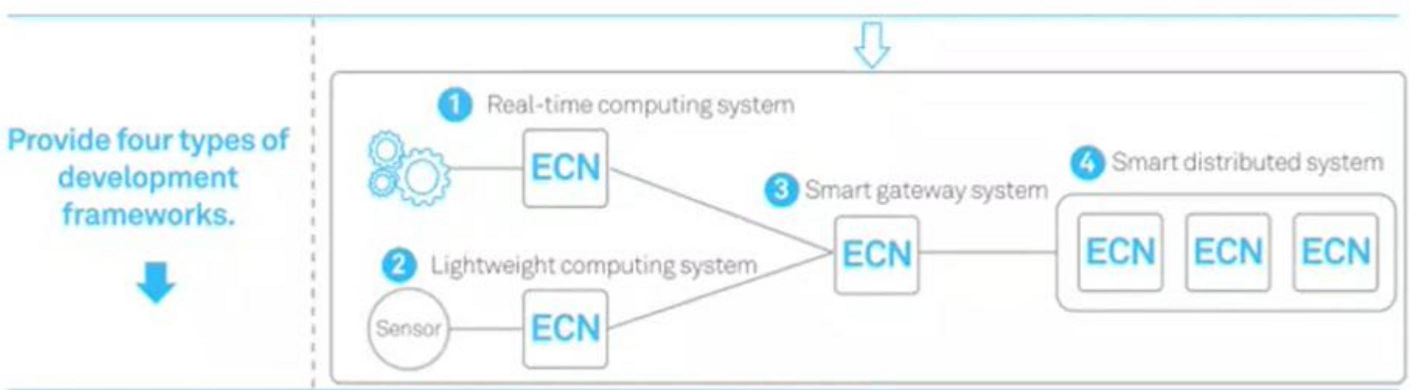
3.Six Types of Products Built on ECNs

Using the four development frameworks, six types of edge computing products can be built:

1. **Embedded controller** – Provides **on-device intelligence** for IoT systems.
2. **Independent controller** – Manages **local decision-making** without a cloud dependency.
3. **Terminal perception** – Supports **real-time data collection** from sensors and IoT devices.
4. **Smart gateway** – Facilitates **data transmission** between edge nodes and the cloud.
5. **Edge distributed gateway** – Optimized for **distributed computing** across multiple edge devices.
6. **Edge distributed server** – Supports **high-performance edge computing** with enhanced **storage and processing power**.

Development frameworks of ECNs

- Real-Time Computing System
- Lightweight Computing System
- Smart Gateway System
- Smart Distributed System



Development frameworks of ECNs

Development Frameworks of Edge Computing Nodes (ECNs) by categorizing them into **four primary types** based on functionality and application needs.

1. Real-Time Computing System

- Designed for **low-latency, time-sensitive** tasks.
- Suitable for applications that require **instantaneous processing** and **real-time decision-making**, such as **autonomous vehicles, industrial automation, and smart grids**.
- ECNs in this system ensure **fast computation** at the edge to minimize **network delays**.

Development frameworks of ECNs

2. Lightweight Computing System

- Optimized for **resource-constrained** environments.
- Works well with **low-power IoT sensors and edge devices** that require **basic data processing**.
- Common use cases include **smart home devices, environmental monitoring, and wearable technology**.

Development frameworks of ECNs

3. Smart Gateway System

- Serves as a **bridge between edge devices and the cloud**.
- Helps in **data aggregation, filtering, and preprocessing** before transmitting data to the cloud.
- Enhances **network efficiency** by **reducing bandwidth usage** and **offloading computation from cloud servers**.
- Suitable for **smart cities, healthcare monitoring systems, and industrial IoT networks**.

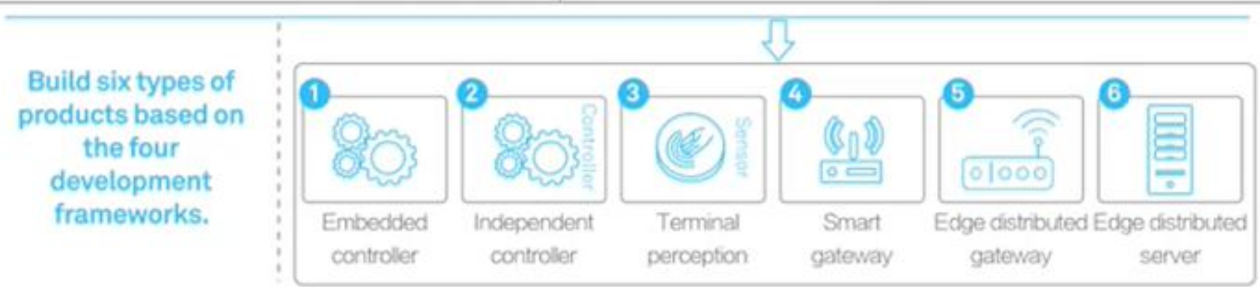
Development frameworks of ECNs

4. Smart Distributed System

- Involves multiple ECNs working together in a **distributed manner**.
- Ensures **scalability and redundancy**, making it ideal for **large-scale applications** like **smart factories, connected vehicles, and edge AI deployments**.
- Supports **fault tolerance and load balancing** by distributing computing tasks among multiple nodes.

ECN product implementation

Product	Typical Scenario
ICT-converged gateway	Connection of elevators, smart street lamp
Independent controller	Industrial Programmable Logic Controller (PLC)
Embedded controller	Virtual Programmable Logic Controller (vPLC), robot
Sensing terminal	Computer Numerical Control (CNC), instrument
Distributed service gateway	Smart power distribution
Edge cluster (edge cloud)	Digital workshop



ECN product implementation

The slide outlines the **Edge Computing Node (ECN) Product Implementation**, detailing six types of ECN-based products along with their **typical application scenarios**.

ECN product implementation

1. ICT-Converged Gateway

- **Function:** Acts as a **central hub** to integrate various smart systems.
- **Use Case:** Used in **smart infrastructure** like **elevators and smart street lamps** to facilitate connectivity and automation.
- **Relevance to Edge Computing:** Enables **real-time data processing** at the edge, reducing dependency on cloud communication.

ECN product implementation

2. Independent Controller

- **Function:** A standalone industrial controller designed for **programmable automation**.
- **Use Case:** Common in **Industrial PLC (Programmable Logic Controllers)**, where automated decision-making is needed without constant cloud connectivity.
- **Relevance:** Ensures **low-latency and high-reliability** automation in **factories and industrial setups**.

ECN product implementation

3. Embedded Controller

- **Function:** Provides embedded processing for intelligent control.
- **Use Case:** Used in **Virtual Programmable Logic Controllers (vPLC) and robotics**, enabling real-time embedded computations.
- **Relevance:** Reduces **computational load** on centralized cloud resources by executing functions **locally on edge devices**.

ECN product implementation

4. Sensing Terminal

- **Function:** A terminal that collects **real-time data** from physical systems.
- **Use Case:** Used in **Computer Numerical Control (CNC) machines and precision instruments** for real-time monitoring.
- **Relevance:** Essential for applications requiring **high precision and low latency**, such as **manufacturing and automation**.

ECN product implementation

5. Distributed Service Gateway

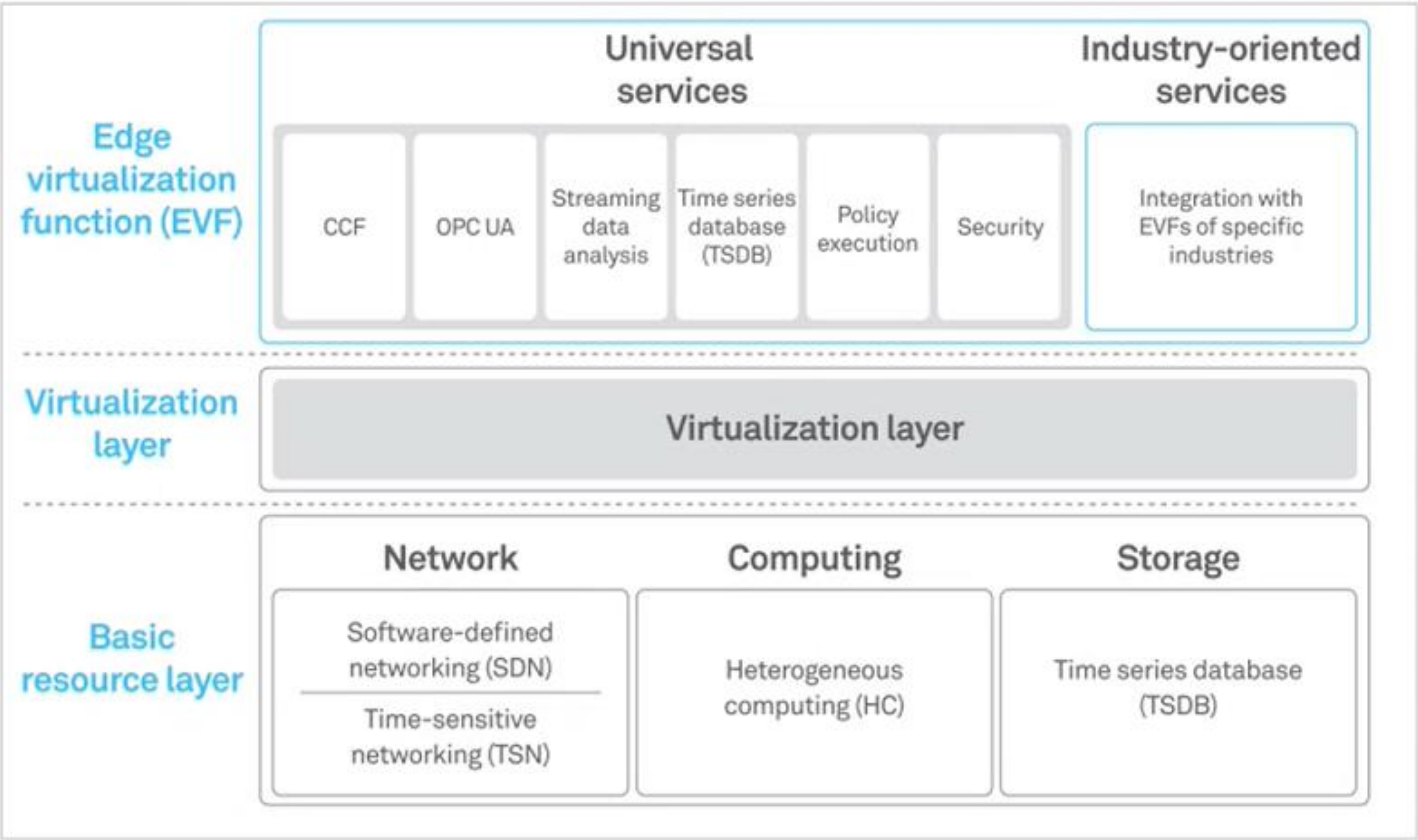
- **Function:** Facilitates distributed computing across multiple networked devices.
- **Use Case:** Used in **smart power distribution** systems to manage and optimize **energy flow dynamically**.
- **Relevance:** Helps **balance computing loads** in a networked environment while enhancing **system efficiency**.

ECN product implementation

6. Edge Cluster (Edge Cloud)

- **Function:** Provides a **distributed edge computing infrastructure** that acts as a mini-cloud.
- **Use Case:** Used in **digital workshops**, where multiple machines and devices collaborate in real-time.
- **Relevance:** Supports **edge-to-cloud communication**, ensuring **scalability and fault tolerance**.

Function view : ECN functional layer



Function view : ECN functional layer

ECN Functional Layer and its **three-tier architecture** for edge computing.

1. **Edge Virtualization Function (EVF) Layer**

- This layer provides **virtualized services** to enable efficient computing at the edge.
- It consists of **two service categories**:
  - **Universal Services** (Common across industries)
    - **CCF (Cloud Control Function)**: Manages cloud-edge interaction.
    - **OPC UA (Open Platform Communications Unified Architecture)**: Facilitates secure, platform-independent industrial communication.

Function view : ECN functional layer

- **Streaming Data Analysis**: Processes real-time data streams.
- **Time-Series Database (TSDB)**: Stores and manages time-dependent data.
- **Policy Execution**: Enforces system-level policies.
- **Security**: Ensures data protection and compliance.

- **Industry-Oriented Services** (Tailored for specific industries)
  - **Integration with Industry-Specific EVFs**: Custom virtualized services for **manufacturing, automation, healthcare, etc.**

Function view : ECN functional layer

2. Virtualization Layer

- This layer **abstracts hardware resources** to allow multiple applications to share computing, networking, and storage efficiently.
- It enables **flexible deployment** of virtual machines (VMs) or containers for running edge applications.

Function view : ECN functional layer

3. Basic Resource Layer

- The foundation layer that provides the **physical and software infrastructure** for ECN.
- It includes:
  - **Network**
    - **Software-Defined Networking (SDN)**: Dynamically manages network traffic for optimal performance.
    - **Time-Sensitive Networking (TSN)**: Ensures low-latency and high-reliability communication, crucial for industrial IoT.

Function view : ECN functional layer

- **Computing**
  - **Heterogeneous Computing (HC)**: Uses a mix of CPUs, GPUs, and FPGAs to optimize processing performance.
- **Storage**
  - **Time-Series Database (TSDB)**: Stores sequential data efficiently, essential for IoT and real-time analytics.

Basic Resource Layer

This layer includes the following modules:

- Network
- Computing
- Storage

Basic Resource Layer

1.Network

- Enables communication between edge devices, cloud, and IoT sensors.
- Technologies involved:
  - **Software-Defined Networking (SDN)**: Dynamically manages network traffic for efficiency.
  - **Time-Sensitive Networking (TSN)**: Ensures low-latency and real-time data transmission, crucial for industrial IoT and automation.

Basic Resource Layer

2.Computing

- Provides the **processing power** for edge computing applications.
- Involves **Heterogeneous Computing (HC)** using:
  - **CPUs** for general computing tasks.
  - **GPUs** for parallel processing and AI/ML tasks.
  - **FPGAs & ASICs** for custom hardware acceleration.

Basic Resource Layer

3.Storage

- Manages data efficiently at the edge to reduce reliance on cloud computing.
- Uses **Time-Series Database (TSDB)** for handling **real-time sensor data, logs, and analytics**.
- Supports **distributed storage** for data resilience and quick retrieval.

Virtualization Layer

Virtualization technology

- reduces system development
- deployment costs

- Virtualization technologies
  - Bare metal architecture
  - Host architecture

- The bare metal architecture has better real-time performance and is generally used by smart assets and smart gateways.

Edge Virtualization Functions Layer

The EVF layer delivers the following basic services:

- Distributed CCF service
- OPC UA service
- Streaming real-time data analysis service
- TSDB service
- Policy execution service
- Security service

Edge Virtualization Functions Layer

innovate achieve lead

1. Distributed CCF (Confidential Computing Framework) Service

- **Purpose**: Ensures **secure and trusted** execution of workloads across a distributed infrastructure.
- **Functionality**:
  - Protects sensitive data during processing.
  - Uses **trusted execution environments (TEEs)** such as Intel SGX or ARM TrustZone.
  - Provides **confidential computing** capabilities for secure collaboration between multiple entities.

Edge Virtualization Functions Layer

2. OPC UA (Open Platform Communications Unified Architecture) Service

- **Purpose**: Enables **standardized, secure, and scalable** communication between industrial devices and software systems.
- **Functionality**:
  - Facilitates **interoperability** between different vendors in Industrial IoT (IIoT) environments.
  - Supports **real-time and historical data exchange**.
  - Enhances **security** with encryption and authentication mechanisms.

Edge Virtualization Functions Layer

3. Streaming Real-Time Data Analysis Service

- **Purpose**: Processes and analyzes **high-speed data streams** in real-time for actionable insights.
- **Functionality**:
  - Supports **event-driven processing** for IoT and industrial automation.
  - Enables **anomaly detection**, predictive maintenance, and real-time decision-making.
  - Works with frameworks like Apache Kafka, Apache Flink, or Spark Streaming.

Edge Virtualization Functions Layer

4. TSDB (Time-Series Database) Service

- **Purpose**: Stores and manages **time-series data** efficiently for analytics and monitoring.
- **Functionality**:
  - Optimized for **timestamped data** such as sensor readings, logs, and metrics.
  - Supports **fast retrieval** of historical and real-time data.
  - Commonly used in **IoT, financial applications, and monitoring systems** (e.g., InfluxDB, Prometheus).

Edge Virtualization Functions Layer

5. Policy Execution Service

- **Purpose**: Implements **governance, security, and compliance** rules for enterprise applications.
- **Functionality**:
  - Enforces **access control policies** and data protection rules.
  - Automates **workflow and decision-making** based on predefined policies.
  - Integrates with **identity and access management (IAM) systems**.

Edge Virtualization Functions Layer

6. Security Service

- **Purpose**: Provides **end-to-end security** for data, applications, and communication channels.
- **Functionality**:
  - Implements **encryption, authentication, and authorization** mechanisms.
  - Protects against **cyber threats, unauthorized access, and data breaches**.
  - Ensures compliance with **security standards like ISO 27001, NIST, and GDPR**.

SDN

SDN's unique benefits:

- Mass Connections
- Model-Driven Policy Automation
- E2E Service Protection
- Lifecycle Management of Applications
- Architecture Openness

TSN

- Standard Ethernet technologies
  - High transmission
  - Speed
  - Flexible topology
  - Long transmission distance
  - Cost-effectiveness
- Constraints
  - Quality of Service (QoS) mechanism and
  - Carrier Sense Multiple Access with Collision Detection (CSMA/CD) mechanism
- Key industry requirements - timeliness and determinism

TSN

- Advantages
  - Ensures μs-level latency and jitter of less than 500 ns
  - Large bandwidth requirements
  - Reliable data transmission

Heterogeneous Computing

The heterogeneous computing architecture uses the following key technologies:

- Memory processing optimization
- Task scheduling optimization
- Tool chain for development

Time Series Database (TSDB)

- Distributed storage
  - Data fragmentation
- Priority-based storage
  - Data processing
  - Data storage
- Fragment-based query optimization
  - Data segments are queried based on query conditions

- Open-source TSDBs, such as OpenTSDB, KairosDB, and InfluxDB

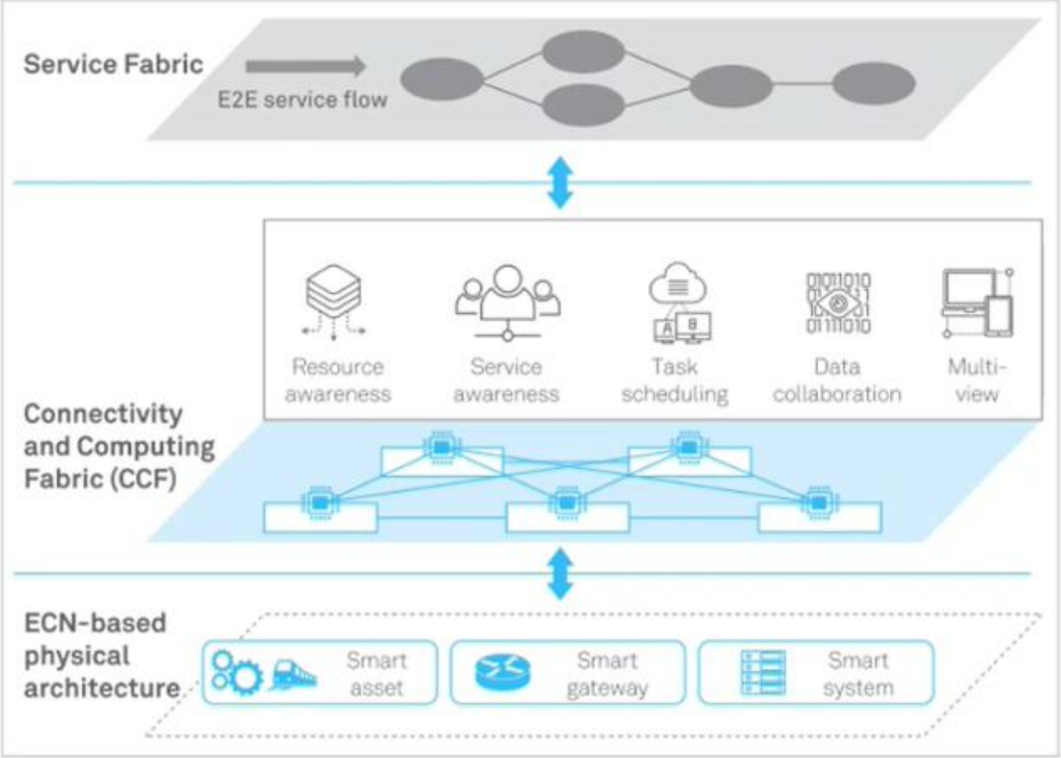
Functional View : Service Fabric

- The service model includes the following information:
  - Service name
  - Function to be executed or provided
  - Nesting, dependency, and inheritance relationships between services
  - Input and output of each service

- A service fabric provides the following functions:
  - Workflow and workload definition
  - Visualized display

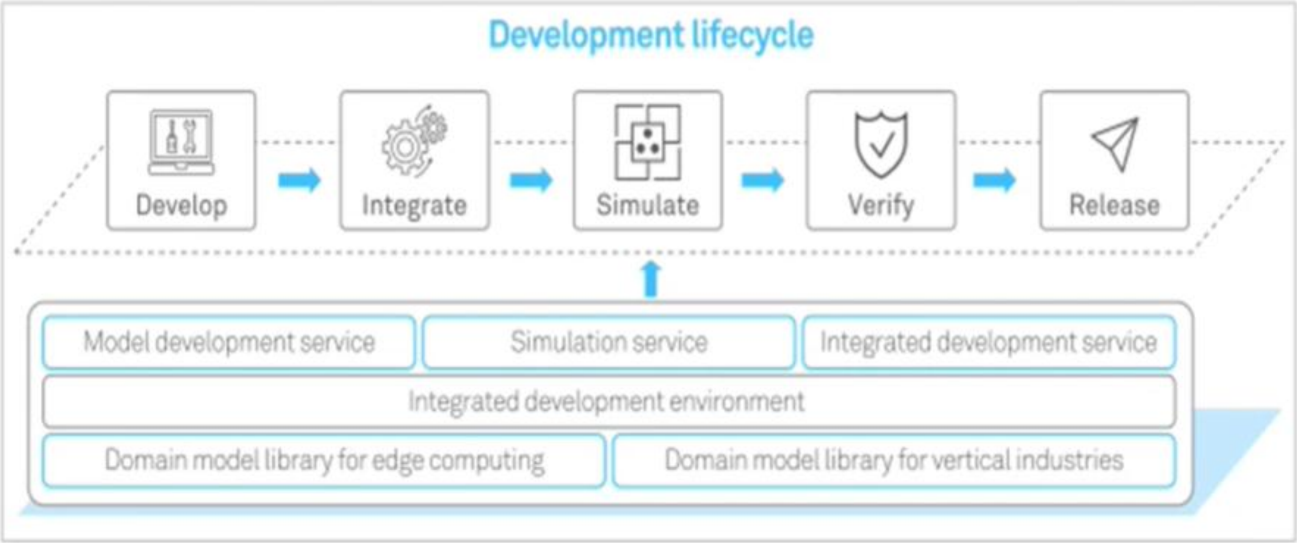
Functional View : CCF

CCF is a virtualized connectivity and computing service layer

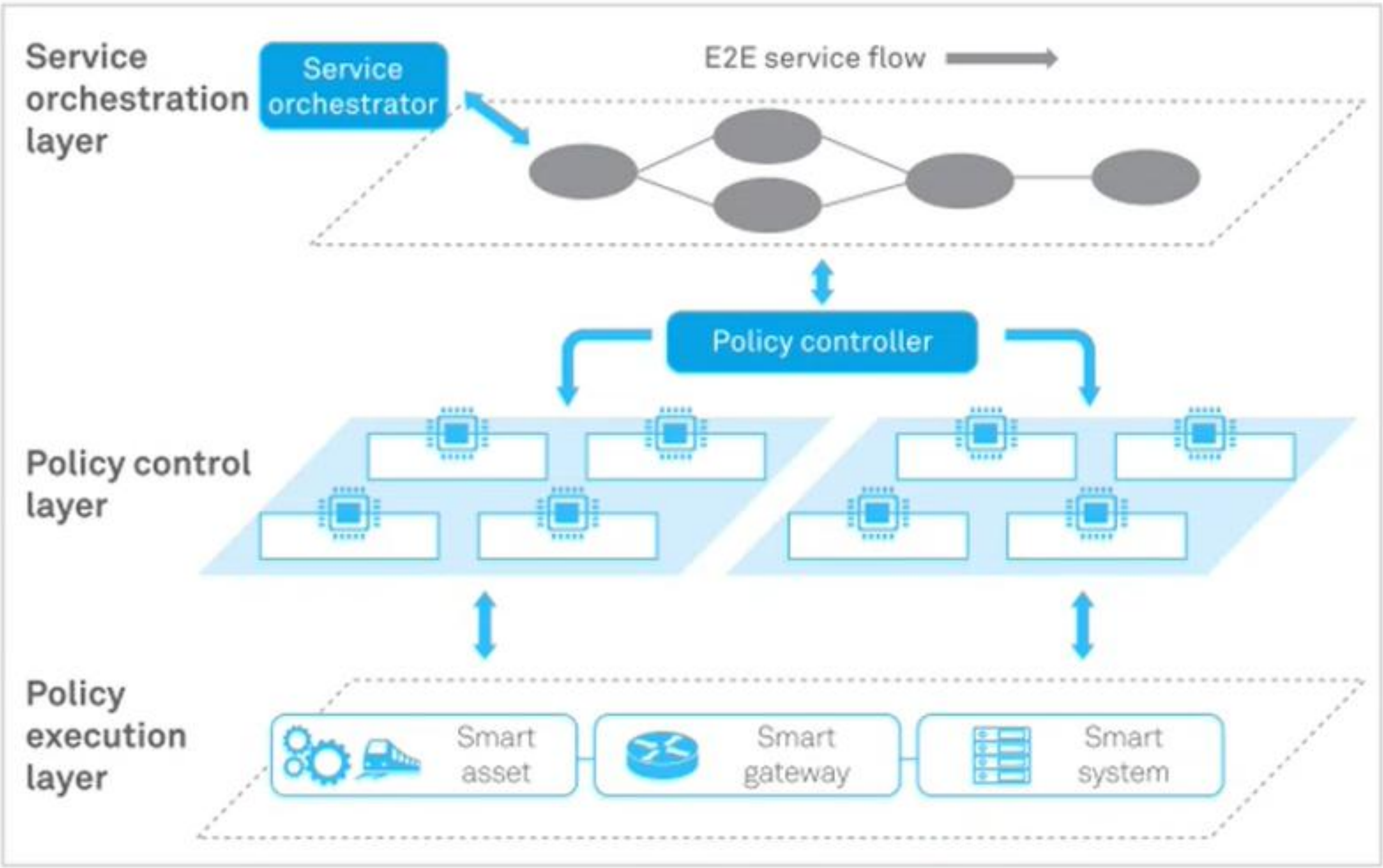


Function View : Development Service Framework

- The development service framework supports the following key services:
  - Model-based development service
  - Emulation service
  - Integrated release service



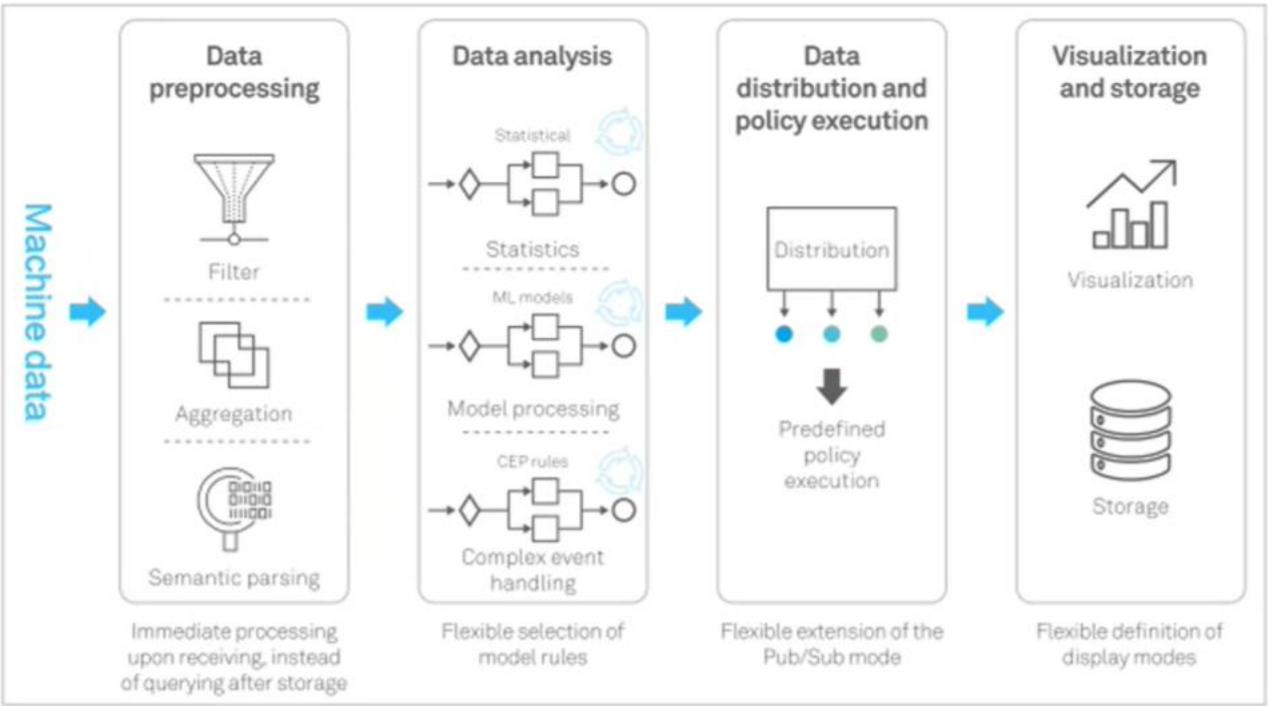
Function View : Development Operation Service Framework



Function View : Development Service Framework

- Edge data characteristics
  - Causal relationship vs. association relationship
  - High reliability vs. low reliability
  - Small data vs. big data

Function View : Full-Life Service Framework



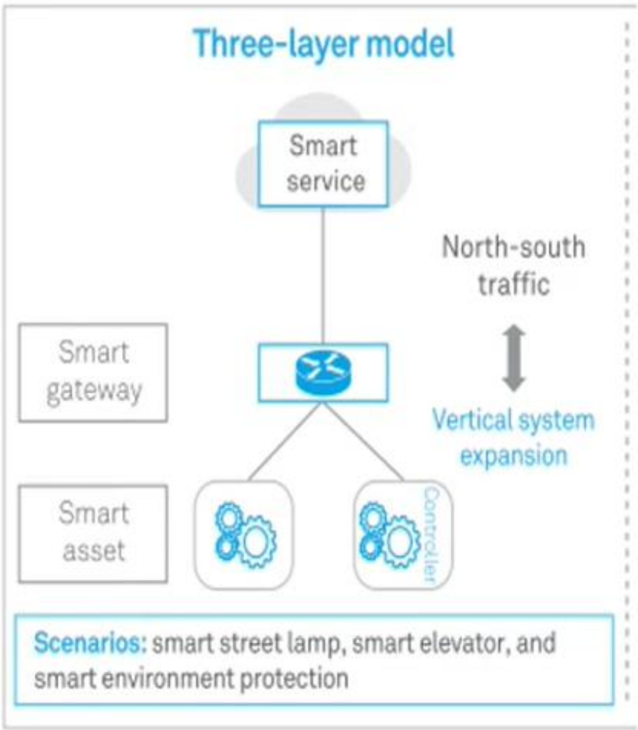
Function View : Security Service

- ECN security
- Network (fabric) security
- Data security
- Application security
- Identity and authentication management

Deployment View : Three-Layer model

- This model is applicable to scenarios where services are deployed in one or more scattered areas, each with a low traffic volume.

- Scenarios include smart street lamps, smart elevators, and smart environmental protection.



Deployment View : Four-Layer model

- This model is applicable to scenarios where services are deployed centrally and the traffic volume is high
- Scenarios include smart video analysis, distributed grid, and smart manufacturing