Table of content

BMIS2113 Information Technology Infrastructure

C1: Introduction to IT Infrastructure

IT infrastructures become more complicated due to new types of applications:

E-Commerce with Personalized Shopping

- Mobile Computing: Users shop via mobile apps with personalized recommendations.
- Cloud Computing: E-commerce platform scales resources dynamically to handle millions of transactions.
- Big Data: Purchase histories, browsing patterns, and social media activities are analyzed.
- Artificial Intelligence: AI engines provide product recommendations and chatbot support.
- IoT: Smart inventory management in warehouses with RFID sensors and automated stock updates.

University Smart Campus System

- Mobile Computing: Campus apps provide schedules, exam results, navigation, and e-wallet services.
- Cloud Computing: Centralized academic management system on the cloud integrates admissions, finance, and library services.
- Big Data: Analyzes student performance trends, library usage, and resource allocation.
- Artificial Intelligence: AI-powered chatbots answer student queries about courses, timetables, or services.
- IoT: Smart ID cards, sensors in libraries/labs, and connected energy-efficient classrooms.

Smart City Traffic Management System

- Mobile Computing: Citizens access traffic updates, parking availability, and alternate routes via a mobile app.
- Cloud Computing: Centralized cloud platform integrates data from different city sensors.
- Big Data: Processes terabytes of data from CCTV, GPS, and sensors to analyze traffic patterns.
- Artificial Intelligence: AI optimizes traffic lights in real-time and predicts congestion hotspots.
- IoT: Roadside sensors, smart cameras, and connected vehicles feed live data.

Agile adaptations require infrastructure:

- **Solid**: Loosely coupled (min dependency) , highly cohesive (max relationship)
- **Scalable**: Ability to handle increased load without performance degradation
- **Modular**: Smaller, independent & reusable components / modules

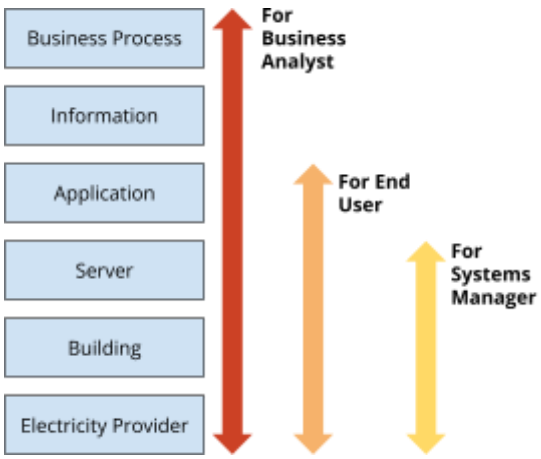
Infrastructure Architecture

Definition

IBM:

- IT infrastructure refers to the combined components needed for the operation and management of enterprise IT services and IT environments.
- Objective: For the operation & management of enterprise IT services and IT environments.

...



Infrastructure comprises组成 depends on: who you ask and what their point of view

BMIS2113 Information Technology Infrastructure

For most people, infrastructure is invisible and taken for granted 是看不见的，是理所当然的

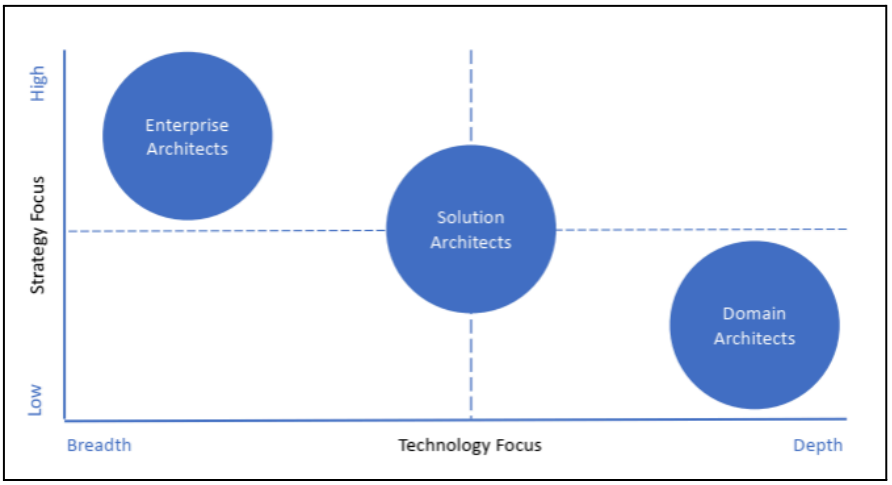
IT Architecture

Architecture is the philosophy that underlies a system and defines its purpose, intent, and structure.

Architecture is crucial to control the infrastructure when it is:

- Designed
- In use
- Changed

Strategy vs technology focus



	Enterprise architects	Solution architects	Domain architects
Strategy focus	High	Moderate	Low
Technology focus	Breadth (wide)	Moderate	Depth
Expert on	Align IT landscape with the business activities	IT solutions, technical & authority of a project	Business / technology topic
Work for	Align the business needs with current & future IT	Project architectural decisions	Infrastructure /software vendors
Assists	CIO & business units	Project manager	Solution architects

IT building blocks & NFA

Functional Management	Process/ information	**(Non-Functional Attributes)** • Availability • Performance • Security
Application Management	Application	
Platform Management	Application platform	
Infrastructure Management	Infrastructure • End User Devices • Operating Systems • Compute • Storage • Networking • Datacentres	

Functional Management	**Process/ information** ● Business processes implemented to fulfil company's mission & vision ● Business processes create & use information
Application Management	**Application** 1) Usage ● Single-user application ● Multi-user application 2) Source ● Commercial off-the-shelf (COTS) ● Custom software. 3) Architecture ● Standalone applications ● Multi-tier applications (front-end, API & backend (database) 4) Timeliness ● Real-time systems: timeliness is critical. ● Interactive applications: respond to user actions ● Batch-based systems: Regular processing
Platform Management	**Application platform** 1) Application servers ● A server that hosts applications or software 2) Container platforms ● A software solution that enables the management of containerized applications 3) Connectivity ● Application server to database ● Application server to container platform ● Databases in container 4) Databases ● Provides ways to store & retrieve data
Infrastructure Management	**Infrastructure** ● **End User Devices** ○ devices used by end users to work with applications ○ PCs, Laptops, Thin clients, Mobile devices, Printers ● **Operating Systems** ○ collections of programs that manage a computer's internal workings: ○ Memory, Processors, Devices, File system ● **Compute** ○ physical and virtual computers in the datacenter. Also known as servers ● **Storage** ○ Storage are systems that store data ○ Hard disks, Tapes磁带, Direct Attached Storage (DAS), Network Attached Storage (NAS), Storage Area Networks (SANs) ● **Networking** ○ Networking connects all components ■ Routers, Switches, Firewalls, WAN, LAN, Internet access, VPNs ○ Includes infrastructure services ■ DNS, DHCP, Time services ● **Datacentres**

	○ locations that host most IT infrastructure hardware ■ Uninterruptible power supplies (UPSs) 不间断电源 ■ Heating, Ventilation, and Air Conditioning (HVAC) 暖通空调 ■ Computer racks ■ Physical security measures

Infrastructure management are processes

- Information Technology Infrastructure Library (ITIL)
- Control Objectives for Information and Related Technology (COBIT)
- DevOps

Tools are used for:

- Monitoring
- Backup
- Logging

Infrastructure building blocks are not per definition hierarchically related!

- For instance, servers need both networking and storage
- Both are equally important

Non-functional attributes (NFRs)

- The name "Non-functional attributes" suggests they have no function 表明它们没有功能
- They are very important for the successful implementation and use of an IT infrastructure
- The term non-functional requirements or NFRs is frequently used and widely known
- The acceptance of a system is largely dependent on the implemented non-functional requirements

Conflicting NFRs

- Many of the non-functional attributes are delivered by the infrastructure
- Non-functional requirements are often conflicting:
  - Security versus user friendliness
  - Performance versus cost
- The infrastructure architect should present stakeholders with these conflicting requirements and their consequences 应该展示这些相互冲突的需求及其后果

Cloud Computing	model that **provide on-demand network access** to a **shared pool of configurable computing resources,** users can easily access and use shared IT resources from anywhere, anytime, **without managing the infrastructure themselves**
Objective	Enables ubiquitous, convenient, on-demand network access to a shared pool of computing resources for rapid provision & release with minimal management effort / service provider interaction. Outsourcing外包模式. To cut cost while focusing on core business
Datacenters	• On premises • On cloud • On hybrid mode: premises + cloud
Popular public cloud providers	• Amazon Web Services (AWS) • Microsoft Azure • Google Cloud Platform (GCP)

Cloud is new infrastructure

Most organizations will be using on-premises infrastructure本地基础设施 for many years to come

In many cases, there will be a hybrid situation: on-premises + clouds

The cloud is just a number of datacenters that are still filled with hardware: compute, networking and storage.

Cloud computing is a model for enabling ubiquitous无处不在, convenient, **on-demand按需 network access to a shared pool** of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be **rapidly provisioned** 快速配置 and **released with minimal management effort** or service provider interaction.

Cloud computing is an outsourcing business model外包业务模式. It enables organizations to cut costs while at the same time focusing on their primary business.

Cloud Characteristics

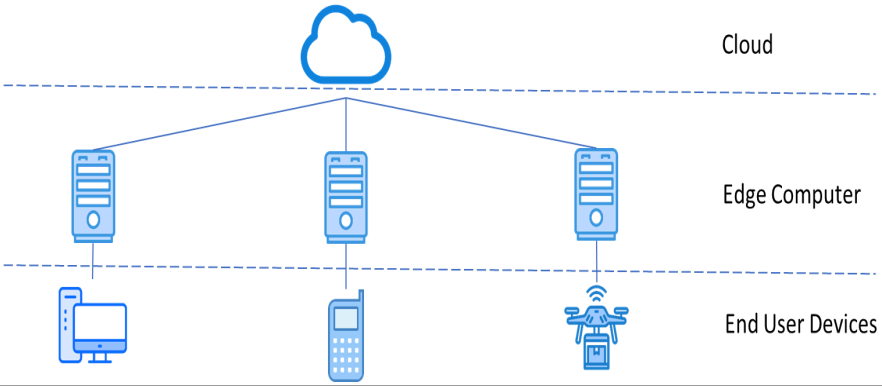
1. On demand self-service	• Min systems management effort is needed for deployment • End users can configure, deploy, start & stop systems on demand
2. Rapid Elasticity	• Able to quickly scale-up & scale-down resources
3. Resource Pooling	• Provides resources from a shared pool (using virtualization technologies)
4. Measured service	• The actual resource usage is measured and billed • There are no capital expenses, only operational expenses
5. Broad network access	• Capabilities are available over the network

Cloud Deployment Models (4)

Public cloud: delivered by a cloud service provider, accessible through internet
Private cloud: operated solely for a single organization, managed internally
Community cloud: operated by one or more parties in communities with shared concerns,
Hybrid cloud: combine Public + (community / private) cloud; Public: Run generic services (email servers) Community/ Private: Host specialized services (specific apps)

Cloud Service Models (3)

Software-as-a-Service (SaaS)	Consume	**Application** Building block	• Delivers full applications • Little / no configuration needed • E.g.: Microsoft Office365, LinkedIn
Platform-as-a-Service (PaaS)	Build	**Application Platform** Building block	• Delivers a scalable, high available, open programming platform • Used by developers to build applications • E.g.: Microsoft Azure Cloud Service, Google App Engine
Infrastructure-as-a-Service (IaaS)	Host	**Infrastructure** Building block(Compute[Operating, Virtualization, Servers], Storage, Networking, Datacentres)	• Delivers virtual machines, networking, storage • Needs to install and maintain the OSs & the layers above that. • E.g.:Amazon Elastic Cloud (EC2 and S3) and Microsoft Azure IaaS



Edge computing	Brings computing power and data storage closer to where it is needed

	such as at the edge of the network
Objective	Min cloud / on-premises datacenter access
Components	Routers, gateways, switches & sensors

Pros	• Low latency
	  ○ Data is processed **close to the source** (e.g., sensors, devices), reducing the delay to send data/ get response with distant cloud servers
	• Low bandwidth needs
	  ○ Only necessary/ summarised data sent to the cloud, minimize network usage
	• Real-time processing
	  ○ data is analyzed locally at the edge, responses and decisions can be made instantly
	
	# Most processing happens locally at the edge node (closer to the device), so only small or non-urgent data goes to the cloud. Cloud used only for heavy or historical tasks

Application	IoT applications (a large number of devices generate data to be processed in real time)

C2: Non-Functional Attributes

Non-functional attributes	• To describe the **qualitative behaviour of a system** • For the successful implementation, use & acceptance of an IT infrastructure
A.k.a.	Non-functional requirements or NFRs
Components	• Availability • Performance • Security
Conflicts	• Security vs. user friendliness • Performance vs. cost

Availability

Definition	Uptime / Ability to provide products or services without interruption / downtime.
Measured by	Expressed as a percentage of uptime (in one year / one month basis)
Characteristics	Cannot be guaranteed upfront
Carrier grade availability	• 99.999% uptime (for one component) • For A full IT system: 99.8% or 99.9% (per month) • For the IT infrastructure: 99.99% or higher
Calculating availability	• a) Mean Time Between Failures (MTBF): Uptime b) Mean time to Repair (MTTR): Downtime • **Availability** 1) **Single component**: One defect leads to downtime **Availability = (MTBF / (MTBF+MTTR)) x 100%** 2) **Serial components**: One defect leads to downtime **Total Availability = Availability x Availability** 3) **Parallel components**: One defect leads to no downtime But beware of SPOFs **Total Availability = 1 - (1 - availability) x (1 - availability)**

To **min MTTR** / downtime	1) Having a service contract with the supplier 2) Having spare parts on-site 3) Automated redundancy & failover
Steps to complete **repairs**	1) Notification of the fault (time before seeing an alarm message) 2) Processing the alarm 3) Finding the root cause of the error 4) Looking up repair information 5) Getting spare components from storage 6) Having technician come to the datacenter with the spare component 7) Physically repairing the fault 8) Restarting & testing the component

Sources of unavailability

1) Human errors (accident)
2) Software bugs (software complexity)
3) Planned maintenance (upgrade, migration, changes)
4) Physical defects (mechanical part likely to break first)
5) Bathtub curve 类似一开始不会坏，用着也不会坏，但是长时间后坏(new component likely to fail)
6) Environmental issues (failing facilities, disaster)
7) Complexity of the infrastructure (complex system)

To achieve high availability

1. **Redundancy**
   - The duplication of critical components in a single system
   - E.g. A single component having 2 power supplies

2. **Failover**
   - A (semi) **automatic** switch-over to a standby system from the **same location**
   - E.g. Windows Server failover clustering

3. **Fallback (3)**
   - A **manual** switchover to an identical standby computer system in a **different location**
   - Used for disaster recovery

Solutions (3)

	Hot site	**Cold site**	**Warm site**
Hardware , Power & cooling	**Fully equipped & configured**	**Hardware only**	Ready
Applications	Installed on servers	Needs to be installed	Not be installed
Data kept	**Up-to-date**	Current data	Need to restore
	For full mirror	Restored from backup	
Pros	Accurately mirrors	Low cost	Hot + cold site
Cons	Requires constant maintenance	Least preference method	Testing needed. Slow.

4. **Availability in the cloud**
   - **Regions & availability zones**

	Regions	Availability zone
Definition	A geographically defined area that contains multiple Availability Zones.	A physically isolated location within a Region
Scope	Broader	Localized within region
To maintain availability	Fallback	Local failover (using update & fault domains)

   - **Fault & update domains**

	Fault domains	**Update domains**
Definition	A group of virtual machines that share a common power	A group of virtual machines that can be rebooted during

	source & network switch	planned maintenance
Protection against	Hardware and infrastructure failures	Downtime during planned maintenance and updates
E.g.	When a fault domain fails (e.g., a rack outage), VMs within that domain are affected, but **VMs in other fault domains remain online**.	When an update domain is rebooted, it's given a **recovery time** (e.g., 30 minutes) **before the next update domain** is affected

5. **Business continuity - IT disaster**
   - An irreparable problem in a datacenter due to unusable
   - Types
     i. Natural disaster - Floods, hurricanes, tornadoes, earthquakes
     ii. Man-made disaster - Hazardous material spills, infrastructure failure, bio-terrorism
   - Impact: Infrastructure become unavailable
   - Solutions

1) Business continuity management (BCM)	• IT • Managing business processes • Availability of people and workplaces in disaster situations
2) Disaster recovery plan (DRP)	• A set of measures to take if disaster • To accommodate the IT infrastructure in an alternative location

RTO & RPO are critical components of BCM & DRP

RTO	• Recovery Time Objective (RTO) • The maximum duration to be restored after a disaster • To avoid unacceptable consequences • A shorter RTO implies a faster recovery and often requires more resources.
RPO	• Recovery Point Objective (RPO) • The point (in time) to which data must be recovered considering some "acceptable loss" in a disaster situation • A lower RPO means less data loss and usually requires more frequent backup

Performance

Definition	Perceived performance refers to how quickly a system appears to perform its task
Indicator	Inform the user about how long a task will take, using progress bars, splash screens

Calculating performance (2)

A. Performance during infrastructure design phase

Nature	Complexity, extremely difficult, unreliable
Considerations	● When the system works as expected - normal ● When the system is in a special state (e.g.: failing parts, maintenance state, performing backup, and running batch job)

To evaluation performance:

Benchmark 跑分对比

- Uses a test program to assess the relative performance of an infrastructure component
- Scope
  - Performance of various subsystems
  - Across different system architectures
- E.g.
  - Compare the raw speed of parts of an infrastructure (processor)

👍**Vendor experience**

- Vendors have experience running their products in various configurations
- Vendors can provide: Tools, Figures & Best practices

Prototype (aka PoC)

- To measure the performance of a system at an early stage, for part with highest risk
- How?
  - Hiring equipment from suppliers
  - Using datacenter capacity at a vendor's premise
  - Using cloud computing resources

User profiling 预计负载 在还没正在开发

- Predict the load a new software system to the infrastructure before the software is actually built to get expected usage of the system
- How?
  - Define a number of typical user groups
  - Create a list of tasks personas will perform on the new system
  - Decompose tasks to infrastructure actions
  - Estimate the load per infrastructure action
  - Calculate the total load

How?

Scalable cloud environment

- In cloud environments, it offers rapidly elasticity
- Cloud environments have extensive logging and monitoring capabilities

B. Performance of a running system(2)

1. **Manage bottleneck**
   a. A component causing the system to reach limit, that negatively influence performance
   b. Every system has at least 1 bottleneck that limits its performance
   c. If the bottleneck has no negative impact to performance of the system under the highest expected load, it is OK
   d. BASED ON
      The performance of all its components
      The interoperability互通性 of various components
2. **Performance test (3)**
   a. **Load test**
      i. Shows how a system performs under the expected load
   b. **Stress test**
      i. Shows how a system reacts when it is under extreme load
   c. **Endurance test** 耐力测试

<table>
<tr><td></td><td>
i. Shows how a system behaves when it is used at the expected load for a long period of time
d. Breakpoint - How
e. Ramp up / increase the load
f. Ideal using Cloud environment, due to: Rapidly elasticity, Reduce the cost, Simulating a very large number of users
g. Software
i. One or more servers to act as injectors
1. Each emulating a number of users
2. Each running a sequence of interactions
ii. A test conductor
1. Coordinating tasks
2. Gathering metrics from each of the injectors
3. Collecting performance data for reporting purposes
h. Where
i. Production-like environment (for reliability)
ii. Temporary (hired) test environment (for min cost)
</td></tr>
<tr><td>Performance patterns (to improve performance)</td><td>
1. Increase Upper Layer Performance
● Definition/Working: Improve performance at application & database level using tuning, task prioritization, working on memory rather than disk , make good use of queues and schedulers.
● Pros: Low-cost, flexible, faster response.
● Cons: Limited impact if hardware is weak.
● Suitable for: General systems where quick software optimization is needed.
2. Caching
<u>a. Disk Caching</u>
● Definition: Stores recently read disk data in cache.
● Pros: Faster access to common data.
● Cons: Cache miss still requires slow disk access.
● Suitable for: File systems, operating systems.
<u>b. Web Proxies</u>
● Definition: Proxy server storing frequently accessed web content.

● Pros: <u>Faster browsing, more bandwidth to use (due to no download needed)</u> .
● Cons: May serve outdated content if not refreshed.
● Suitable for: Organizations with many users accessing similar web content.
<u>c. Front-End Server (Web Server)</u>
● Definition: Serves static data and acts as reverse proxy(auto-cache mort requested data).
● Pros: Reduces backend load, caches static content.
● Cons: Limited for dynamic data.
● Suitable for: Websites with many static resources (images, CSS, JS).
<u>d. In-Memory Databases</u>
● Definition: Entire DB stored in RAM.
● Pros: Extremely fast performance.
● Cons: Risk of data loss if power fails; costly RAM.
● Suitable for: Real-time financial trading, gaming, analytics.
<u>e. Edge Servers</u>
● Definition: Cache data close proximity to end-users (edge locations).
● Pros: Low latency, reduced backbone traffic.
● Cons: Expensive to deploy globally.
● Suitable for: Cloud providers, CDNs (Netflix, Cloudflare).

3. Operational Data Stores (ODS)
● Definition: Database <u>integrating real-time operational data from multiple sources</u>.
</td></tr>
</table>

- Pros: Real-time insights, reduces load on main DB.
- Cons: Extra system to manage, requires integration.
- Suitable for: Businesses needing consolidated, real-time operational reporting.

4. Scalability

a. Vertical Scaling (Scale Up)

- Definition: Add resources (CPU, RAM) to a single server/components.
- Pros: Simple, no software change.
- Cons: Hardware upgrade limit, costly.
- Suitable for: Small/medium systems with moderate growth.

b. Horizontal Scaling (Scale Out)

- Definition: Add more servers/components to infrastructure.
- Pros: High scalability, fault-tolerant.
- Cons: Complexity, requires distributed system design.
- Suitable for: Cloud computing, big data, SaaS platforms.

5. Load Balancing

- Definition: Distributes incoming requests across multiple servers.
- Pros: Improves availability, avoids single-server overload.
- Cons: Extra cost, requires application to support multi-server.
- Suitable for: Web apps, e-commerce, cloud services.
- **Load balance**
  - Spreads the load to available machines:
  - Checks the current load on each server in the farm
  - Sends incoming requests to the least busy server
- **Advanced load balancers**
  - Spread the load based on Server side's:
  - Number of connections
  - Response time

6. High-Performance Computing

a. High Performance Cluster

- Definition: Combines many servers into one powerful computing unit.
- Pros: Massive computing power at relatively low cost (off-the-shelf servers).
- Cons: Needs high-speed interconnects, complex management.
- Suitable for: Scientific research, simulations, weather forecasting.

b. Grid Computing

- Definition: Geographically distributed computing cluster.
- Pros: Utilizes underused resources across locations.
- Cons: Bandwidth limitation, security risks.
- Suitable for: Research collaborations across universities/companies.

7. Design for Use

- Definition: Optimize system based on workload purpose.
- Tips:
  - Match design to purpose (batch, online, real-time).
  - Spread load across systems.
  - Move rarely used data off the main system.
- Pros: Tailored, efficient performance.
- Cons: Requires deep understanding of workload.
- Suitable for: Enterprises with mixed workloads (e.g., ERP + analytics).

8. Capacity Management

- Definition: Monitor and plan resources for long-term high performance.
- Pros: Prevents future bottlenecks, aligns with business growth.

	• Cons: Requires continuous monitoring & forecasting. • Suitable for: Any business-critical IT system.

Security

Definition	• Security is the combination of: - Availability : Authorised user has reliable access when needed - Confidentiality : Sensitive info is accessible by authorised user only - Integrity : Maintain accuracy & trustworthiness of data	
Characteristics	Focused on the recognition and resistance of attacks	
Core infrastructure security	• Irreversibility of hash keys • Practical unbreakable encryption • Unbreachable virtualization	
Motive /reason for crime against IT infrastructure	• Personal exposure and prestige (visibility / perceiveability by others) • Creating damage • Financial gain • Terrorism • Warfare	
Cloud security	• The public cloud applies a shared responsibility model: - The cloud provider takes care of security of the cloud, with many specialists - The customer takes care of security in the cloud	
Prevention	• Design for minimum risk (using source code analysis, standalone system) • Incorporate safety devices (using firewall, hardened screened routers) • Implement training & procedures (to mitigate risk, ensure proper use)	
Implementation	Zero trust	Assumes no implicit trust regardless of location & requires continuous verification of users and devices before granting access
	Segregation of duties 职责分离	Divides critical tasks among multiple individuals to prevent fraud, errors & unauthorized access
	Least privilege	Granting users only the minimum necessary access rights to perform their job functions
	Privileged Access Management (PAM)	Manage & monitor access to sensitive resources and privileged accounts.
	Layered security	Multiple security controls to protect against different types of threats
	Identity and Access Management (IAM)	Manage user identities &access rights across an organization
	Authentication	Verify the identity of a user or device before granting access
	Password	A secret string of characters used to authenticate a user's identity
	Role Based Access Control (RBAC)	Control access to resources based on user roles
	Cryptography	Secure communication & data by transforming it into an unreadable format, making it incomprehensible to unauthorized

	Encryption	Encode data using algorithms & keys, converting readable information (plaintext) into an unreadable format (ciphertext)
	Computer Emergency Response Team (CERT)	A team that responds to computer security incidents

C3: Datacenter

3.1 Functions of a Datacenter

Definition: Most IT infrastructure hardware, except for end-user devices, are hosted in datacenters.

Functions A datacenter provides:

- Power supply
- Cooling
- Fire prevention and detection
- Equipment racks

History Early datacenters (computer rooms): Designed and built for large mainframe systems.

Today's Datacenters

- Equipped with standardized 19" racks: House CNS equipment
- Contains shipping containers packed with thousands of servers each (for large scale)

3.2 Datacenter Building Blocks

a) Sub Equipment Room (SER)

- smaller (office building)
- Size: Patch closet, small room, or closet that houses networking/electrical equipment
- Key Function: Acts as a distribution point for network connections

b) Main Equipment Room (MER)

- larger (centralized room)
- Size: A small datacenter located in the organization's subsidiaries or buildings
- Key Function: Houses core network infrastructure

c) Organization Owned Datacenter (OOD)

- larger (purpose-built facility)
- Size: A datacenter that contains all central IT equipment for the organization
- Key Function: Houses the organization's entire IT infrastructure

d) Multi-Tenant Datacenter (MTD)

- largest (facility shared by multiple org)
- Size: Used by service providers to offer services to multiple organizations
- Key Function: Hosts IT equipment for multiple tenants

Factors Determining the Locations of Datacenters

- Environment of the datacenter
- Visibility of the datacenter
- Utilities available to the datacenter
- Datacenters located in foreign countries

Physical structure: FWWDP

Floors

1. Datacenter Floor Load
    - Can carry 1500 to 2000 kg/m²
    - A fully filled 19" computer rack weighs approximately 700 kg
    - Rack footprint (60x100 cm) results in a load of 1166 kg/m²
2. Office Floor Load
    - Can carry approximately 500 kg/m²
3. Raised Floor
    - Constructed with metal framework and removable tiles (60x60 cm)
    - Disadvantages:
        - Expensive

        - Reduces total available height in the datacenter
        - Limits maximum floor load
        - Difficult to install doors and equipment loading slopes
        - Fire can easily spread through
        - **Overhead cable trays** are a safer alternative

Vents

- provide cool airflow to racks on the floor

Walls, Windows, Doors

- Walls
    - Must extend from floor to ceiling
    - Require adequate fire rating to act as physical firewalls
- Windows
    - Not desirable in datacenters
    - If present, must be:
        - Translucent
        - Shatterproof
        - Impossible to open
- Doors
    - Must be large enough for equipment
    - Should resist forced entry

Water and Gas Pipes

- Leakage from ceiling-mounted water pipes can damage equipment
- Datacenter operators must know the location of shutoff valves

Layout of Datacenter – Rooms and Components

1. Computer Room
    - Location where actual IT infrastructure components are installed
2. UPS or Generator
    - Diesel generator provides backup electrical power
    - Fuel should be stored outside or in an isolated, secure room nearby
3. Input Power Transformers
    - Transformers from the power utility company
4. UPS (Uninterruptible Power Supply)
    - Provides continuous power during short outages
5. UPS Batteries
    - Battery set for short-term power supply
6. Cooling
    - Systems for maintaining optimal temperature
7. Fire Extinction
    - Fire suppression systems
8. Operator Room
    - Includes a large window facing the computer room for monitoring activity
9. Storage Room
    - Stores spare hardware and equipment
10. Entrance
    - Access point to other rooms; no windows
11. Meeting Room

- Used for staff and visitor meetings
- Includes a secured (shatterproof) window for sunlight

🔌 Power Density and Server Rack

Power Density (kilowatts per m^2):

- Power drawn:
  - 1 rack of servers: measured in kilowatts (kW)
  - Large facilities: measured in megawatts (MW)
- Normal-density datacenter: 2–6 kW/m^2
- High-density datacenter: 10–20 kW/m^2 (racks filled with 40–80 servers)

Server Rack Considerations:

- Cannot be fully equipped due to:
  - Stability and transport
  - Power and cooling limitations
  - Accessibility and maintenance needs
  - Future expansion
  - Cable management

⚡ Uninterruptible Power Supply (UPS) Systems

Objective

- Prevent power issues that lead to downtime and equipment damage

Characteristics

- Operates independently of utility power
- Delivers high-quality electrical power

Installation Components

- Filters
- Diesel power generator
- Battery set or flywheel system

Types of UPS Systems

- Battery Powered UPSs
  - Also called Standby UPS or Off-line systems
  - Suitable for small setups (few workstations or servers)
- Line Interactive UPS Systems
  - Use a transformer between utility power and IT equipment
  - Filter many power issues
- Double Conversion UPS Systems
  - Convert AC to DC and back to high-quality AC
  - Ensures stable and clean power supply

🔋 Power Distribution Units (PDUs)

Definition

- A device with multiple outlets that distributes power to datacenter equipment

Types

- Floor PDUs:
  - Used in large datacenters
  - Distribute power to multiple racks or equipment
- Rack PDUs:
  - Mounted in standard 19-inch racks
  - Offer various power distribution options

High Availability Strategy

- Equip infrastructure with two power supplies for redundancy
- Use at least two power strips per rack for reliable power delivery

❄️ Cooling Systems in Datacenters

Objective

- To dissipate heat generated by IT equipment

Types of Cooling Systems

- CRAC (Computer Room Air Conditioners)
    - Use refrigerants
    - Connected to external condensing units
- CRAH (Computer Room Air Handlers)
    - Use chilled water
    - Connected to outside chillers
    - Chillers produce chilled water via refrigeration

Efficiency Metrics

- EER (Energy Efficiency Ratio)
    - Measures efficiency at maximum load
    - Ratio: BTU/hour output cooling ÷ Watts input
- SEER (Seasonal Energy Efficiency Ratio)
    - Similar to EER but based on seasonal usage (typically summer)
- COP (Coefficient of Performance)
    - Ratio: Cooling load in kW ÷ Electric energy input in kW
    - Typical values: 3 to 10

Operating Temperature

- Normal range: 18–27 °C
- Server shutdown threshold: 40 °C at air inlet
- Raising temperature by 1 °C reduces cooling cost by 5%

🌬️ Environmental Guidelines for Datacenter Operation

Operating Temperature

- Maintain air temperature between 18–27 °C
- Servers shut down at 40 °C inlet temperature
- Increasing temperature by 1 °C saves 5% in cooling costs

Airflow

- Optimize airflow to eliminate hot spots
- Avoid excessive cooling of the entire datacenter

Liquid Cooling

- Suitable for large datacenters
- Components immersed in non-conductive, non-corrosive fluid
- Allows closer placement of system boards, increasing CPU density

Humidity and Dust Control

- Maintain humidity between 40% and 60%
- Minimize dust by:
    - Restricting visitor access
    - Wearing dust-free clothing and protective shoe sleeves

🔥 Fire Prevention, Detection, and Suppression

- Fires may be caused by short circuits or defective equipment
- Airflow and raised floors can accelerate fire spread
- Smoke can damage equipment

Four Levels of Fire Safety:

1. Fire Protection – Prevent fire occurrence
2. Passive Fire Protection – Limit fire exposure once started
3. Fire Detection – Identify smoke and fire
4. Fire Suppression – Extinguish fire after detection

🗄️ Equipment Racks

- A 19-inch rack is a standardized metal enclosure for IT components
- Rack height is measured in rack units (U)
    - 1U = 44.5 mm
    - Typical rack height = 42U

⚡ Datacenter Energy Efficiency

- Energy cost often exceeds server cost
- Power Usage Effectiveness (PUE) measures total datacenter power usage:
    - PUE = Total datacenter power ÷ IT equipment power
    - Typical PUE values range from 1.1 to 2.0

Example:

- A datacenter with PUE = 1.5 means:
    - 1 watt used by IT equipment
    - 0.5 watt used by the rest of the datacenter (cooling, lighting, etc.)

3.3 Datacenter APS

🏢 Datacenter Availability- 4 Tiers

Availability tiers describe the reliability of datacenter facilities, not the IT infrastructure components. They are based on uptime, redundancy, and fault tolerance.

Tier	Infrastructure Description	Expected Uptime
Tier 1	Basic infrastructure	99.671%
Tier 2	Adds some redundancy and backup paths for power and cooling	99.741%
Tier 3	Multiple redundant paths for power and cooling; allows maintenance without downtime	99.982%
Tier 4	Fully fault-tolerant; redundant systems for power, cooling, and all IT equipment	99.995%

Redundant Datacenters

- Used to increase availability beyond Tier 4 (99.995%)
- Multiple datacenters are required for ultra-high availability
- Redundant datacenters should be located at least 5 km apart

🧹 Floor Management (by Floor Manager)

- Minimize personnel walking around server racks
- Keep the datacenter floor tidy
- Change backup tapes
- Provide power connections to racks
- Maintain and test fire extinguishing and UPS systems

🚀 Datacenter Performance

- Datacenter does not directly affect IT infrastructure performance
- Performance contributions are limited to:
    - Bandwidth of internet connectivity
    - Scalability of the location

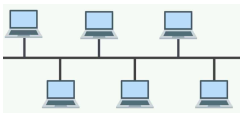
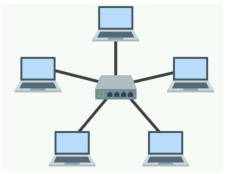
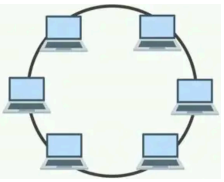
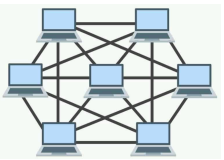
🔐 Datacenter Security

Physical Security Measures:

- Restrict physical access to selected, qualified staff
- Use entry registration systems and maintain access logs
- Secure doors with conventional or electronic locks (with authentication)
- Entry points may include:
    - Regular doors
    - Mantraps
    - Revolving doors
    - Weighing scales to ensure only one person enters at a time

C4: Networking

Network Topologies

	Bus	Star	Ring	Mesh
				
Connection by	Single cable / bus	Central hub / switch	A circular fashion	By connected to multiple devices
Data transmission	Along the bus, to intended recipient	Through the central point	In 1 direction around the ring	Routed through different paths
Pros	Simple & inexpensive to implement	Easy to manage and troubleshoot	Simple & efficient	Fault-tolerant
Cons	Prone to performance issues	Central hub as a SPOF	Disruptive to add / remove devices	Expensive to implement

Networking Building Blocks

- OSI Reference Model (OSI-RM) defines the different stages that data must go through to travel from one host to another over a network

1 Physical Layer

- Deal with the physical components that carry the data
- Data transmission media and connectors
- Signal characteristics
- Network topology

Cables	Twisted pair cables	<u>Unshielded Twisted Pair (UTP)</u>
		• Consists of twisted wire pairs without any additional shielding
		• Used in **Ethernet** networks and for telephone systems
		• Pros:
		  ○ Cost-effective
		  ○ Easy to install
		  ○ More flexible
		• Cons:
		  ○ Susceptible to interference
		  ○ Lower data rate
		<u>Shielded Twisted Pair (STP)</u>
		• Suitable for most home and office networks where interference isn't a major concern
		• Used in **factories, datacenters, or areas with high electrical noise**
		• Pros:
		  ○ Reduce interference
		  ○ Higher data rates
		• Cons:
		  ○ More expensive
		  ○ More complex installation
		  ○ Less flexible

<table>
<tr><td rowspan="2"></td><td>Fiber Optic Cables</td><td>Multimode
Core diameter: 50 or 62.5 microns (larger)
Light source: LEDs (multiple paths)
Used in LAN, datacenter, short distance
Cost: Less expensive
Single-mode
Core diameter: 9 microns (smaller)
Light source: Laser (single path)
Used in telecommunication, long distance
Cost: More expensive</td></tr>
<tr><td>Coaxial cables</td><td>For TV & Internet connection, outside Malaysia</td></tr>
<tr><td rowspan="2">Patch Panel</td><td>Function</td><td>Provides a centralized location to organize and manage network cables</td></tr>
<tr><td>Pros</td><td>Organise cables
Ease changes
Protect connections
Provide clear reference for labelling & documentation</td></tr>
<tr><td rowspan="2">Cabling</td><td>Vertical Cabling</td><td><table><tr><th>Vertical Cabling</th><th>Horizontal Cabling</th></tr><tr><td>connects different floors or buildings within a network infrastructure
Connection: Patch panels to data center
Long distance</td><td>Connects devices within a floor
Endpoints in the walls to patch panels
Up to 90 meters</td></tr></table>Pros
Enhance network reliability: fault isolation, increase uptime, disaster recovery
Improve network performance: Load balancing, reduce latency
Simplify network management: Centralized management, ease troubleshooting
Connection
The main distribution cabling connects the patch panels on the floors to the datacenter
Connects floors and buildings
Cable Types
Fiber Optic (primary)
Twisted pair
Coaxial</td></tr>
<tr><td>Horizontal Cabling</td><td>Pros
Same with vertical cabling
Connection
Endpoints in the walls are connected to the patch panels
Within a specific floor / building
Cable Types
Twisted pair (primary)
Fiber optic</td></tr>
</table>

<table>
<tr><td rowspan="2">Leased lines</td><td>Definition</td><td>• A dedicated data connections between two locations
• Provided by a telecom provider</td></tr>
<tr><td>Types</td><td>
<u>T or E lease line</u>
• Definition: Not standard industry terms for leased lines
• Pros:
  ○ Low data rate: T (1.544 Mbps), E (2.048 Mbps)
• Used in
  ○ T (USA, Canada and Japan)
  ○ E (most other countries)
<u>Synchronous Optical Network (SONET)</u>
• Definition: Transmit large amounts of data using fiber optic
• Pros:
  ○ High data rate
  ○ Low latency
• Used in:
  ○ Connecting corporate offices & datacenters
  ○ Enabling high-speed data replication & backup
  ○ Supporting real-time applications with low latency requirements
<u>Synchronous Digital Hierarchy (SDH)</u>
• Definition: The international standard equivalent to SONET
• Pros:
  ○ High reliability
  ○ Scalability
• Used in:
  ○ Telecommunications to transmit multiple digital data stream over a fiber optic
Dark Fiber
<table>
<tr><th>Dark Fiber</th><th>Regular Fiber (Lit Fiber)</th></tr>
<tr><td>Not actively used by the provider
Installed but unused fiber optic cables</td><td>Actively used by the provider to deliver services</td></tr>
<tr><td>Fully controlled by the customer(lessee)</td><td>Controlled and managed by the provider</td></tr>
<tr><td>Lessee installs and manages own equipment</td><td>Provider supplies and manages equipment</td></tr>
<tr><td>Private, point-to-point network</td><td>Shared infrastructure managed by ISP</td></tr>
</table>
• Definition: Unused fiber optic cables that have been installed but are not actively transmitting data
• Pros:
  ○ High bandwidth, Low latency
  ○ Security, Scalability, Reliability
• Used in:
  ○ To accommodate future growth
  ○ Leassee then installs their own equipment to "light up" the fiber and create their own private network
</td></tr>
<tr><td>Internet Access</td><td>DIA</td><td>
Characteristics
• High speed
• Reliable connectivity
• Guaranteed bandwidth
Connection
• Dedicated cable connection to single business
Ideal for Business with critical applications
</td></tr>
</table>

	Broadband	Characteristics • High-speed over cable, Digital Subscriber Line (DSL) or fiber optic networks Connection • Shared among multiple users Ideal for Internet browsing, Email
	WiFi	Characteristics • Includes Wi-Fi, cellular data and satellite Internet Connection • Shared connection without physical cable Ideal for Wirelessly

2 Data Link Layer

- Ensure reliable & error-free data transfer between devices
- Takes packets received & breaks them into frames
- Detect errors in the frames
- Manage how devices access the physical medium to prevent collisions

Network

	PAN	LAN	MAN	WAN
Distance	Within a few meters	Home, office building or school	A city or a large campus	Large geographical area
Used in	Connecting personal devices within a few meters	Setting up a network in a home office, or campus for fast local access	Linking multiple LANs across a city or large institution	Connecting networks across cities, countries or globally
Protocols	Bluetooth (L2CAP), Zigbee, Infrared	Ethernet (CSMA/CD), Wi-Fi (CSMA/CA)	Metro Ethernet, DQDB	PPP, HDLC, Frame Relay

Common Protocols Used

	Ethernet	WiFi	PPP
Technology	Wired	Wireless	Direct connection between two devices
Used for	Connecting devices in a LAN	Connecting to the Internet wirelessly	VPNs, dial-up & Internet connection

Implementation

- Ethernet
- Switching: Split a single network segment into multiple segments, for each device
- Public wireless networks:

	Features	Used in
1G	Limited data speeds and security features	(Analog) Support voice calls
2G	Improved call quality and security	(Digital) Improved call quality and security
3G	Increased data speeds, allowing for mobile internet access	Support email, web browsing

4G	Further speed improvements	Streaming videos and online gaming
5G	Faster speeds, lower latency (delay) and greater capacity	Internet of Things (IoT), virtual reality and self-driving cars

3 Network Layer

- Define the route the data is sent to the recipient
- Implementations:
    - IPv4, IPv6, Routing and addressing

IP Protocol

A unique numerical label assigned to each device connected to a network

Server can have multiple IP addresses, both public and private

- **Load balancing**
- **Hosting multiple websites**
- **Providing different services on the same server**

	Public IP Address	**Private IP Address**
Visible to	**Entire network**	**Within a local server within a private network**
Used for	• Web server • Email server • Public-facing applications	• Internal communication

Addressing

	Static IP Address	**Dynamic IP Address**
Nature	Constant	Change periodically
Ideal for	Public servers	Internal servers

Routing

- Routers compile routing tables to make IP packet forwarding decisions
- Routing in the context of a server, covers:
    - Network Routing
        - Direct data packets are from the server to the other network devices
    - Application-Level Routing
        - Handle incoming requests within the server itself: load balancing, Content Delivery Networks (CDNs), application-specific routing

4 Transport Layer

- Maintain flow control
- Provide error checking
- Recovery of data between network devices

Protocols Used

	Transmission Control Protocol (TCP)	User Datagram Protocol (UDP)
Function	Provides reliable delivery of a stream of data between applications	Reduced latency over reliability by sending data without checking if the data arrived

Used in	FTP Web browsing Email	Live stream Online games VoIP
Used for	Large DNS queries	Small DNS queries

DNS - Domain Name System

- Port 53 is specifically for DNS. Other services should not use this port.
- Firewalls should allow traffic on port 53 for DNS to function correctly

5 Session Layer

- Provides mechanisms for opening, closing and managing a session between end-user application processes

Virtual Private Network (VPN)

- Uses a public network to interconnect private sites in a secure way, a.k.a. VPN tunnel
- Uses "virtual" connections based on IPsec / SSL
- Most network providers also offer private VPNs based on MPLS
- VPNs use strong encryption and strong user authentication. Using the Internet for transmitting sensitive data is considered safe
- VPN tunnels are often used for remote access to the LAN by users outside of the organization's premises
- Most common VPN communication protocol standards:

	Point-to-Point Tunneling Protocol (PPTP)	Layer 2 Tunneling Protocol (L2TP)	IPsec
Uses	For individual client to server connections	For individual client to server connections	For network-to-network connectivity. IPsec is built into IPv6 standard and is implemented as an add-on to IPv4
Security	Least	No encryption and relies on IPSec for encryption and authentication	Provides encryption and authentication for network traffic
Used in	Legacy system or when speed is prioritized over security	Paired with IPSec for a more secure VPN connection	VPNs and other secure network applications

6 Presentation Layer

- Takes the data provided by the application layer and converts it into a standard format that the other layers can understand

	Secure Socket Layer (SSL)	Transport Layer Security (TLS)
Nature	SSL is considered insecure and should not be used	TLS is securing WWW traffic carried by HTTP to form HTTPS

7 Application Layer

- Interacts with the OS or application

Roles of application layer in servers

- Protocol implementation
- Data formatting
- Error handling

- Security
- Session management

Protocols used

Protocol	Definition	Function
Domain Name System (DNS)	DNS is a distributed database that links IP addresses with domain names DNS was not designed with security in mind Updates to DNS records are done in non-encrypted clear text Authorization is based on IP addresses only	Translates human-readable domain names (like google.com) into machine-readable addresses
DNS Security Extensions (DNSSEC)	Provides origin authentication of DNS data for data integrity	Verifies the authenticity & integrity of DNS data
IP Address Management (IPAM)	IPAM systems are appliances that can be used to plan, track and manage IP addresses in a network IPAM systems integrate DNS, DHCP and IP address administration in one high available redundant set of appliances	Plans, tracks and manages the IP address space within a network
Network Time Protocol (NTP)	NTP ensures all infrastructure components use the same time in their real-time clocks Particularly important for: Log file analysis, clustering software, Kerberos authentication	Synchronization clocks of computers in a network to a common time source
Post Office Protocol (POP)	Used by email clients to retrieve email messages from a mail server	Retrieves emails from a mail server
Simple Mail Transfer Protocol (SMTP)	Used to send email messages from a mail client to a mail server or between mail servers	Sends emails from a mail client to a mail server or between mail server
Multipurpose Internet Mail Extensions (MIME)	Enables SMTP to support file attachments in email messages	Transmits non-text data (like images, audio, video) in email
File Transfer Protocol (FTP)	A protocol for transferring files between computers	Transfers files between computers over a network
Hypertext Transfer Protocol (HTTP)	Defines how messages are formatted and transmitted, and how web servers and browsers should respond to various commands	Transfers hypertext (HTML, CSS, JavaScript, images, etc)
Hypertext Transfer Protocol Secure (HTTPS)	Used when browsing the web with a web browser	Protected data during transfer

* Example refers to images

4.2 Networking Virtualization

Approaches	Used for / as	How
Virtual LAN (VLAN)	Logical grouping (for network segmentation)	Logically divides a single physical network into multiple broadcast domains
Virtual Extensible LAN (VXLAN)	Network virtualization technology	Uses encapsulation to create virtual networks that can span across physical networks, allowing for greater scalability and flexibility compared to VLANs
Virtual Routing and Forwarding (VFR)	Network routing technology (for segmentation)	Hosts multiple independent routing tables in a single physical router
Virtual Network Interface Controllers (VNIC)	Software-based representation of a network interface	Enables virtual machines to connect to the network and communicate with other virtual machines or physical devices
Virtual Switch (VS)	Software-based equivalents of physical network switches	Manages network traffic within a virtualized environment, providing functionalities like VLAN tagging, traffic shaping and connection to physical networks
Software Defined Networking (SDN)	Software-based controllers (Abstract control plan from data plane)	Manages network traffic and resources, enabling dynamic and programmable network configurations
Network Function Virtualization (NFV)	Network architecture (virtualize network function)	Replaces traditional dedicated network hardware appliances with virtualized software instances running on commodity servers

* Example refers to images

4.3 Networking: Availability, Performance & Security

Networking Availability

Layered network topology

Definition	Divides a network into multiple layers, each with specific functions and responsibilities
A.k.a.	Tiered or hierarchical topology
Components	Core Layer ● Definition ○ The center of the network ○ The backbone of the network ○ It typically uses high-capacity routers & switches ● Nature ○ High-speed network backbone between network segments ● Function ○ Offer rapid & reliable transfer for large volume of data Distribution / Aggregation Layer ● Definition ○ It combines the access layer data and sends its combined data to one or two ports on the core switches ● Nature ○ Sits between the core (in datacenter) & access layers (in patch closet) ● Function ○ Performs routing functions, implements security policies, access and core layers Access Layer ● Definition ○ For servers, located at server racks / in blade enclosures ○ For workstations, placed in patch closets ● Nature ○ Connect workstations & servers to the distribution layer ● Function ○ Connect end-users and devices to the network
Benefits	● Improve availability & performance (with multiple paths to any piece of equipment) ● Provides scalability ● Provides deterministic routing: advance determination of the routes between given pairs of nodes ● Avoids unmanaged ad-hoc data streams

* Example refers to images

Spines and Leaf topology

Definition	A modern datacenter network architecture designed to address the limitations of traditional three-tier hierarchical designs
Natures	● The spine switches are not interconnected ● Each leaf switch is connected to all spine switches ● Each server is connected to two leaf switches ● The connections between spine and leaf switches typically have 10 times the bandwidth of the connectivity between the leaf switches and the servers
Key components	Spine Switches

	• Function
	  ○ Form the core of the network
	• Nature
	  ○ Interconnected in a full-mesh topology, providing redundancy & load balancing
	• Working (Retail Industry)
	  ○ The spine layer serves as the core of the network, connecting all leaf switches
	
	Leaf Switches
	• Function
	  ○ Connect to servers and other end devices
	• Nature
	  ○ Each leaf switch connects to all spine switches, creating multiple paths for traffic
	• Working (Retail Industry)
	  ○ Leaf switches act as access points, connecting servers, point-of-sale (POS) systems, inventory management systems, and other devices to the network

Working	A simple physical network is used that can be programmed to act as a complex virtual network. Such a network can be organized in a spine and leaf topology
Benefits	• Highly scalable: There are no interconnects between the spine switches • Simple to scale: Just add spine or leaf servers • Physical servers can be connected using relatively few switches • Predictable latency: Each server is always exactly four hops away from every other server

* Example refers to images

Network Teaming

A.k.a.	Link aggregation, port trunking, network bonding
Definition	A method of **combining multiple network interface cards (NICs) into a single logical interface**
Objective	• Provides a virtual network connections using multiple physical cables • To achieve high availability and increased bandwidth • To improve network performance and reliability
Working	Bonds physical NICs together to form a logical network team: • Sends traffic to the team's destination to all NICs in the team • Allows a single NIC, cable or switch to be unavailable without interrupting traffic

* Example refers to images

Spanning Tree Protocol (STP)

Definition	An Ethernet level protocol that runs on switches
Working	• Guarantees only one path is active between two network endpoints at any given time • Redundant paths are automatically activated when the active path experiences problems • Ensures no loops are created when redundant paths are available in the network

Pros	• Shortest Path Bridging (SPB) allows all paths to be active simultaneously, enables much larger topologies, supports faster convergence times • Improves efficiency by allowing traffic to be load balanced across all paths. While STP can take 30 to 60 seconds to respond to a topology change, SPB can respond to changes in less than a second
Cons	It is not using half of the network links in a network, since it blocks redundant paths

* Example refers to images

Multihoming

Definition	Connecting a network to two different Internet Service Providers (ISPs)
Methods	• Single router with dual links to a single ISP • Single router with dual links to two separate ISPs • Dual routers each with its own link to a single ISP • Dual routers each with its own link to a separate ISP
Pros	• Improve availability
Cons	It is not always guaranteed that multiple network paths actually run on a different set of cables. Cables are used by multiple carrier providers.

Networking Performance

Factors affect the speed of a connection

- Throughput and bandwidth

Definition	Amount of data that is transferred through the network during a specific time interval
Constraint	Throughput is limited by the available bandwidth
Working	When an application requires more throughput than a network connection can deliver: • Queues in the network components temporarily buffer data • Buffered data is sent as soon as the network connection is free again • When more data arrives than the queries can store in the buffer, packet loss occurs

- Latency

Definition	The time from the start of packet transmission to the start of packet reception
Depend on	• The physical distance a packet has to travel • The number of switches and routers the packet has to pass
Rules	• 6 ms latency per 100 km • WANs: Each switch in the path adds 10 ms to the one-way delay • LANs: Add 1 ms for each switch
Types	One way latency • The time from the source sending a packet to the destination receiving it Round-trip latency • The one-way latency from source to destination plus the one-way latency from the destination back to the source A "ping" can be used to measure round-trip latency

- Quality of Service (QoS)

Definition	Ability to provide different data flow priority to different applications, users or types of data
Nature	Allows better service to certain important data flows compared to less important data flows
Used for	Real-time applications like video and audio streams and VoIP telephony
Implementations methods	Congestion Management • To prioritize traffic • Defines action when the amount of data to be sent exceeds the bandwidth of the network • Packets can either be dropped or queued Queue Management • To make the wait time more manageable and transparent • Defines criteria for dropping packets that are of lower priority before dropping higher priority packets • When queue are full, packets will be dropped Link efficiency • To ensure efficient use of available resources • Ensures the link is used in an optimized way • By fragmenting large packets with low QoS, allowing packets with a high QoS to be sent between the fragments of low QoS packets Traffic shaping • To prioritize certain types of traffic over others • Limit the full bandwidth of streams with a low QoS • Have QoS streams have a reserved amount of bandwidth

- WAN Link Compression

Definition	Data compression reduces data size before it is transmitted
Pros	WAN acceleration appliances: • Provide compression • Perform some caching of regularly used data at remote data

* Example refers to images

Networking Security

Network Encryption

Definition	Encryption is often a feature in the datacenter
Types	• Encrypting data in transmit: Encrypting data on the network • Encrypting data at rest: Encrypting data in the storage • End-to-end encryption: Encrypting data between 2 end-points, with network traffic encryption

Firewalls

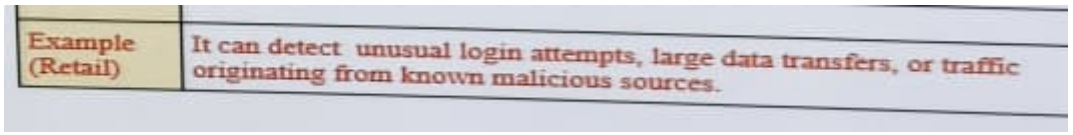
Definition	• Firewalls separate 2 / more LAN / WAN segments for security reasons • Firewalls block all unpermitted network traffic between network segments • Permitted traffic must be enabled by configuring the firewall to allow it

Implementation	• In hardware appliances • As an application on physical servers • In virtual machines
Host based firewall	• Protect a server or end user computer against network based attacks • Part of the operating system
Traffic control methods	Packet Filtering • Data packets are analyzed using preconfigured filters • This function is always available on routers & most OS Proxy • A proxy terminates the session on the application level on behalf of the server (proxy) or the client (reverse proxy) and creates a new sessions to the client or server Stateful Inspection • Inspects the placement of each individual packet within a packet stream • Maintains records of all connections passing through the firewall and determines whether a packet is the start of a new connection, part of an existing connection, or is an invalid packet

Intrusion Detection System (IDS) or Intrusion Prevention System (IPS)

Definition	Detects & prevents activities that compromises system security (a hacking attempt)
Working	• Monitors for suspicious activity • Alerts the systems manager when these activities are detected • Stop attacks by changing firewall rules on the fly
Types	Network-based IDS (NIDS) • Placed at a strategic point in the network • Monitors traffic to and from all devices on that network • The NIDS is not part of the network flows, but just "looks at it", to avoid detection of the NIDS by hackers Host-based IDS (HTDS) • Runs on individual servers or network devices • It monitors the network traffic of that device • It also monitor user behavior and the alteration of critical (system) files

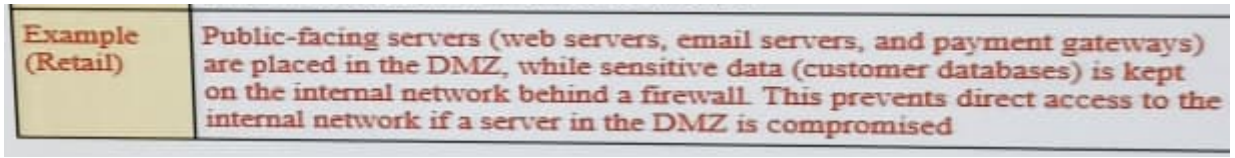
* Example



De-Militarized Zone (DMZ)

Definition	A DMZ is a network that serves as a buffer between a secure protected internal network and the inaccurate internet

* Example



Remote Authentication Dial In User Service (RADIUS)

Definition	RADIUS is a networking protocol that provides centralized user and authorization management for network devices

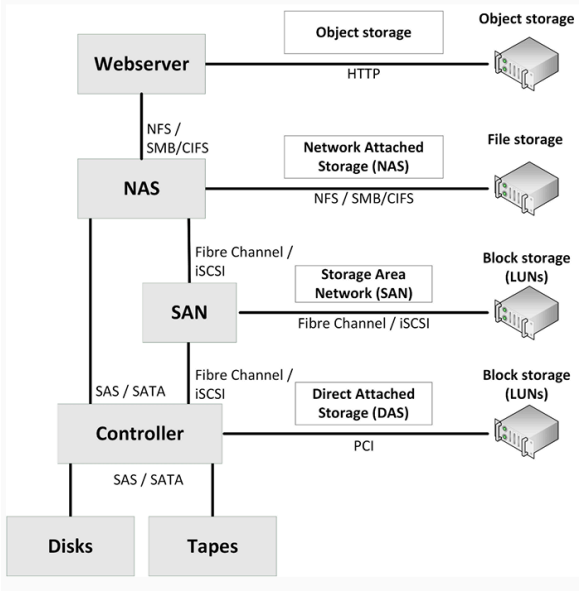
Example (Retail)	Many retail stores offer free Wi-Fi to customers to enhance their shopping experience. RADIUS can be used to authenticate customers attempting to

* Example

C5: Storage

Storage Model

Server: external + internal



5.1 Storage building blocks

1 Disks – command sets

Advanced Technology Attachment (ATA)	• Aka IDE • simple hardware and communication protocol (disks to PCs) • (Common interface standard for connecting storage devices like HDDs and SSDs to computers.)
Small Computer System Interface (SCSI)	• set of standards for physically connecting and transferring data between computers (mostly servers) and peripheral devices, like disks and tapes • (High-performance command set used mainly in servers and enterprise systems.)
command sets (ATA or SCSI) stayed mostly the same, but the physical connection type changed from parallel interface(many wires at once) to serial interface(one wire sends data in order) to improve speed, reliability, and scalability	

1 Disks - type (2)

Mechanical Hard Disk (HDD)	SATA (Serial ATA): Standard interface for consumer PCs; cost-effective and widely used. SAS (Serial Attached SCSI): Enterprise-grade interface offering higher speed and reliability. NL-SAS (Nearline SAS): Combines large capacity of SATA with reliability of SAS; used for backup or archival storage.
Solid State Drive (SSD)	Uses flash memory with no moving parts. Provides faster data access, lower latency, and higher durability compared to HDDs.

2 Tape (2)

Cheapest option for long-term storage and archiving.

Life expectancy up to 30 years (e.g., DLT, SDLT, LTO).

Disadvantages: fragile, slow sequential access, prone to damage.

Tape Cartridge Formats	SDLT (Super Digital Linear Tape): Up to 300 GB. LTO (Linear Tape Open): Most common; 80%+ market share. LTO-9: 18 TB uncompressed, ~200 MB/s throughput.
Library Types	Tape Library: Automates tape handling using drives, slots, barcode/RFID, and robotic loaders. Virtual Tape Library (VTL): Disk-based backup emulating tape systems; faster, supports parallel processing; typically uses SATA/NL-SAS arrays.

3 Controller Implementation

Connect disks/tapes to servers via PCI or network (NAS/SAN).

Provide virtualization, high availability, deduplication, cloning, and thin provisioning

RAID (Redundant Array of Independent Disks)

- Improves performance and data availability using **multiple disks**.
- Implemented in hardware (controller) or software (OS).

Common RAID Levels:

- RAID 0 (Striping): Speed only; no redundancy.
- RAID 1 (Mirroring): Two copies; high reliability, 50% efficiency.
- RAID 10: Combines striping + mirroring; high performance & availability.
- RAID 5: Striping + single parity; balanced speed and protection.
- RAID 6: Dual parity; tolerates two disk failures.

RAID Level	Description	Benefits	Drawbacks	Use Case
RAID 0 (Striping)	Data split evenly across multiple disks (no redundancy)	Increases performance	No redundancy; if one disk fails, all data is lost	Temporary or non-critical data where speed matters
RAID 1 (Mirroring)	uses two disks that contain the same data Duplicates data exactly	High availability; most reliable	Uses 50% of disk space for redundancy; expensive	Critical data needing high reliability
RAID 10 (Striping + Mirroring)	Combination of RAID 0 and RAID 1	High performance and availability	Uses 50% disk space for redundancy; costly	Databases and critical systems needing speed + redundancy
RAID 5 (Striping + Distributed Parity)	Data and parity blocks striped across disks	Good balance of performance, availability, and storage efficiency	Single disk failure tolerated; slower write speed due to parity calculations	General purpose with moderate reliability needs
RAID 6 (Striping + Double Parity)	Like RAID 5 but with two parity blocks	Can tolerate two disk failures	More overhead than RAID 5; slower writes	Systems requiring extra fault tolerance during rebuilds

Compression	• Reduces storage needs (2–2.5× typical), depending on data type.
Data Deduplication	Removes duplicate data segments to save space. Inline deduplication: immediate but slower. Post-process deduplication: done later, reduces performance impact.
Cloning & Snapshots	Clone: full copy of data (like RAID 1). Snapshot: point-in-time version; read-only original, writes go to a new volume. Used for backups, testing, or quick restore.
Thin Provisioning	Allocates more logical space than physically available. Dynamically assigns real capacity as needed; ideal for home directories, email storage.

4 Storage Architectures (4)

Direct Attached Storage (DAS)	• Local disks connected via SATA/SAS controller through PCI bus. • OS manages LUNs and file systems. • Simple and low-cost but limited scalability and sharing
Storage Area Network (SAN)	• Dedicated high-speed network connecting servers and storage (block-level). • Uses Fibre Channel, FCoE, or iSCSI. • Centralized storage with large disk arrays (petabyte scale). Key Components: • Fabric: SAN switch network. • HBA (Host Bus Adapter): Server interface for Fibre Channel or iSCSI. Protocols: Fibre Channel (FC): High reliability, 1–128 Gbps, zero data loss. Topologies: Point-to-Point, Arbitrated Loop, Switched Fabric (most common). FCoE (Fibre Channel over Ethernet): Runs FC over Ethernet ≥10 Gbps using DCB/CEE extensions. iSCSI: SCSI over TCP/IP; cost-effective and easy to deploy; lower performance but closing gap with 10/40 Gbps Ethernet.
Network Attached Storage (NAS)	• Provides shared file-level access over TCP/IP (NFS, SMB/CIFS). • Acts as a file server, may use SAN as backend storage. • Supports file-level snapshots and permissions (LDAP/AD integration). • Clustered NAS: Distributes data across multiple nodes for scalability. • Cloud examples: AWS File Gateway, Azure Storage Account, GCP Filestore.
Object Storage	**Stores data as objects (file + metadata + unique ID) accessed via REST API (HTTP).** • **Immutable — modified files are replaced with new versions.** Ideal for static content (archives, media, backups). Used by AWS S3, Azure Blob, GCP Object Storage. Supports massive scalability and geographic redundancy.

5 Software Defined Storage (SDS)

- Separates control plane (management) from data plane (physical storage).
- Pools all physical storage into a virtualized shared resource.
- Works across SAN, NAS, and Object systems using commodity servers.
- All cloud providers' storage systems are SDS-based.

Functions:

- Provides data services: deduplication, compression, caching, snapshotting, cloning, replication, and tiering.
- Enables policy-based provisioning (e.g., automatic setup for databases with snapshots and tiering).

- Managed via API, CLI, or GUI, ensuring desired performance, availability, and security.

5.2 Storage: Availability, Performance & Security

1- Improve Storage Availability (4)

Refers to how often storage systems remain accessible for data operations (store, retrieve, manage).

1	RAID (Redundant Array of Independent Disks) • Combines multiple disks to provide redundancy and fault tolerance.
2	Redundancy and Data Replication (2) Synchronous replication: • Data is written to both primary and secondary storage before confirming write completion. • Ensures data consistency; risk of latency or downtime if connection fails. Asynchronous replication: • Data is first written to primary storage and later copied to secondary storage. • Reduces latency but may risk small data loss if failure occurs.
3	Backup and Recovery • Protects data from deletion, corruption, or disasters. Follows the 3-2-1 rule: 3 copies, 2 media types, 1 offsite. Backup types: i) Full Backup – Complete data copy. ii) Incremental Backup – Changes since last backup. iii) Differential Backup – Changes since last full backup. iv) Incremental Forever Backup – One full backup + continuous incrementals. v) Continuous Data Protection (CDP) – Captures every data change in real time (zero RPO).
4	Archiving Long-term storage for compliance and regulations. Data is read-only, encrypted, and stored on durable media (e.g., WORM tapes, optical disks). Use open formats (e.g., XML) for future readability; migrate to new media every 10 years.

2- Storage Performance (3)

Refers to how often storage systems remain accessible for data operations (store, retrieve, manage).

To improve storage availability:

1	Factors affecting disk performance: i) Disk Rotation Speed (RPM): Faster rotation → lower delay (e.g., 5400 RPM = 5.6ms; 15000 RPM = 2ms). ii) Seek Time: Time for disk head to locate track (3–9ms typical). iii) Interface Protocol: Determines data transfer rate (e.g., SATA 6Gb/s, SAS 12Gb/s, NVMe 7GB/s).
2	Metrics to measure performance: IOPS (Input/Output Operations Per Second): Number of reads/writes per second. • HDD: <500 IOPS • SSD: ~400,000 (read), 150,000 (write) • NVMe: up to 1,000,000 (read)
3	To improve storage performance: i) Cache: • Read cache stores frequently accessed data.

	• Write-back cache speeds up write operations. ii) Storage Tiering: • Assigns data to storage based on importance and usage (e.g., SSD for Tier 1, SAS for Tier 2, tape for Tier 4). • Automated tiering moves data between tiers based on access patterns. iii) Load Optimization: • Balances workloads to prevent bottlenecks. • Example: Oracle uses a mix of RAID 1 & 5 for database optimization.

3- Improve Storage Security(2)

1	Data at Rest (on disk or tape) **Data Encryption:** Prevents unauthorized access without a decryption key. **Self-Encrypting Drives (SEDs):** Built-in encryption requiring password at startup. **Cryptographic Disk Erasure (CDE):** Deletes encryption keys to render data unreadable.
2	Data in Transit **SAN Zoning:** Divides Fibre Channel SANs into logical groups; only devices in the same zone can communicate. **SAN LUN Masking:** Restricts Logical Unit Numbers (LUNs) to specific hosts; implemented at HBA level. Combining zoning and masking enhances access control.

C6: Compute

6.1 Compute building blocks

1

Compute Housing

Tower (Pedestal): Standalone; placed on floor.
Rack Servers: Standardized frames for multiple systems.

Blade Servers: Shared enclosure components (power, fans, network, SAN); reduced wiring, cost, and failure points. Blade enclosure: houses 8–16 blades; shared redundant power, Ethernet, SAN, and management modules.

- **Reduced wiring then fewer failure points**: it only need to run one cable to the enclosure/chassis机箱,外壳, making it **easier to manage and avoid safety concerns from tangled wires**电线缠绕.
- **Lower initial deployment costs**: it **uses the enclosure's shared components like power supplies and fans, also shared redundant components** (Power supplies, Backplane for interconnection, Network switches (redundant Ethernet connections), SAN switches (redundant Fibre Channel connections)).

2

Processor

1 Central Processing Unit (CPU)

- Executes program instructions (arithmetic, logic, I/O).
- Works using an instruction set (machine code).
- Operates via clock cycles (GHz = billions of ticks/sec).
- Word size: data handled per operation (modern = 64-bit).

Processor Families
Intel x86: Standard CPU architecture.
AMD x86: Competitor; Ryzen 9950X (16 cores, 5.7 GHz).
x86-64: 64-bit architecture used by Intel & AMD.
ARM: RISC-based; used in smartphones, tablets, IoT; licensed to many manufacturers (e.g., Apple M1–M3, AWS Graviton, Microsoft Ampere, Qualcomm Snapdragon X).

2 Graphics Processing Unit (GPU)

- Thousands of cores for parallel processing.
- Accelerates AI/ML workloads and graphics rendering.

Example: NVIDIA Tesla GP100 (3840 cores, 150B transistors).

#GPU usage increases?
growing demand for the artificial intelligence, machine learning, big data analytics application
GPUs are **highly parallel processors** that much faster than CPUs for workloads of these application

3

Memory

Evolution
Early: vacuum tubes → relays → magnetic core memory → RAM chips.
Core memory was replaced by transistor-based RAM in the 1970s.

1 **RAM (Random Access Memory)**
Volatile; data lost without power.
SRAM(Static RAM): 6 transistors/bit, implement for cache

- fast, costly.

DRAM(Dynamic RAM): 1 transistor + 1 capacitor/bit, implement for main memory
DRAM **loses its data after a short time** due to the leakage of the capacitors, **refresh regularly to keep data available, cheaper**.

2 **BIOS (Basic Input/Output System)**

	Firmware initializing hardware before OS loads. Stored in flash memory; regularly updated (BIOS flashing).
4	Interfaces External Interfaces: 1 USB: Replaced older interfaces (since 1996). USB 3.1 = 10 Gbps; USB-C = 40 Gbps (USB 4) & up to 20V power. 2 Thunderbolt (Light Peak): Up to 80 Gbps; 100W power; uses USB-C connector. Internal Interfaces: 3 **PCI**: (cheaper and wide adoption) slower, uses shared parallel bus, limited bandwidth. 4 **PCIe**: (routed by a hub that allows multiple devices to communicate **simultaneously**) faster, uses point-to-point serial links, and scalable bandwidth

6.2 Compute Virtualization

LPAR (Logical Partition)	• **Hardware-based virtualization** • full and can different OS per LPAR • divides single physical sys into multiple isolated env. • Common on IBM mainframes and Power Systems.	Enterprise systems, mainframes, midrange
VM (Virtual Machine)	• **Software-based virtualization** • full and can different OS per VM (*Hypervisor* create vm) • creates and manages multiple virtual machines (VMs) on one physical host.	Data centers, servers
Container	• **OS-level** • Shared host OS (Container Engine - Docker,) • Pay for container runtime or host instance	Cloud-native apps
Serverless	• **Function-level** • by third-party cloud providers (AWS,) • Pay per function execution	Event-driven cloud tasks

1 Compute Virtualization vs. Public Cloud

Aspect	**Compute Virtualization (Virtual Machine)**	**Public Cloud (include serverless)**
Definition	Uses **hypervisor虚拟机管理程序** to create **multiple virtual machines (VMs) on a single physical server.**	Provides virtualized compute resources over the Internet, managed by cloud providers.
Ownership	Managed by organization on its own servers. **Full control** over infrastructure.	**Managed by third-party providers** (AWS, Azure, GCP).
Scalability	Limited by on-premise hardware capacity.	Highly scalable on demand.
Cost Model	High initial cost, lower long-term cost.	Pay-as-you-go model, no upfront hardware.

	Use Case (Takoyaki Example)	A local takoyaki shop uses virtualization to run POS and inventory VMs on one physical server.	A takoyaki franchise uses cloud VMs to host its online ordering system that scales automatically during lunch rush hours.

2 Compute Virtualization Technology

1 Emulation

- Simulates different hardware architecture entirely through software.
- Allows running software for one system on another (e.g., PlayStation emulator on PC).

Example (Takoyaki): A takoyaki shop runs an old POS app made for PowerPC on a modern x86 server using emulation.

2 Logical Partitions (LPARs)

- Hardware-based virtualization that divides a single physical system into multiple isolated env.
- Common on IBM mainframes and Power Systems.
- Each LPAR can run its own OS.

Example (Takoyaki): A large takoyaki chain uses an IBM Power server — one LPAR for sales, another for payroll, another for supply management — all isolated but sharing hardware.

3 Hypervisor

- Software-based virtualization layer that creates and manages multiple virtual machines (VMs) on one physical host.
- Each VM has its own virtual CPU, memory, storage, and OS.

Types:

- Type 1 (Bare Metal): Runs directly on hardware (e.g., VMware ESXi, Microsoft Hyper-V).
- Type 2 (Hosted): Runs on top of existing OS (e.g., Oracle VirtualBox).

Example (Takoyaki): The main takoyaki shop uses a hypervisor to host separate VMs — one for accounting, one for recipe management, and one for online orders.

6.3 Types of Compute

1 Compute: Cloud – Containers Technology

i) Technology: **Container**

- Containers package applications with their dependencies to ensure consistency across environments.

ii) Implementation

- Chroot or Jail: Isolate processes and file systems for lightweight containment.
- Namespaces: Provide process isolation by separating system resources (PID, network, users).
- Cgroups: Control and limit resource usage (CPU, memory, I/O) for containers.

iii) Container Orchestration

- Tools like Kubernetes automate deployment, scaling, and management of containerized applications.

2 Compute: Cloud – **Serverless Computing**

- Allows running code without managing servers.
- Automatically scales by providers based on demand
  - charges only for actual execution time.

#Serverless computing means running code without managing servers yourself. The cloud provider (like AWS Lambda, Azure Functions, or Google Cloud Functions) handles server provisioning, scaling, and maintenance automatically. Just upload your function/code → provider runs it when triggered → you pay only for runtime.

3	**Mainframe** #x86 = general-purpose; Mainframe = high reliability, enterprise-scale workloads. i) Components ● PU (Processing Unit): Executes instructions and handles computations. ● Memory: Stores active data and instructions. ● I/O Channels: Manage data transfer between mainframe and peripherals. ● Control Units: Direct and coordinate input/output operations. ii) Mainframe Virtualization: Logical Partitions (LPARs) ● Divides mainframe hardware into multiple isolated logical systems, each running its own OS.
4	**Midrange Systems** i) Midrange Virtualization: Logical Partitions (LPARs) Similar to mainframes but designed for medium-scale workloads in enterprises.
5	**X86 Servers** Industry-standard servers based on x86 architecture, commonly used for virtualization, cloud, and enterprise workloads.
6	**Supercomputers** #Supercomputer = classical parallel processing; Quantum = quantum bit-based, experimental. Extremely powerful systems designed for intensive scientific and engineering computations requiring parallel processing.
7	**Quantum Computers** Use quantum bits (qubits) and quantum mechanics principles to perform complex calculations exponentially faster than classical computers.

6.4 Compute: Availability, Performance & Security

1	Improve Compute Availability ● **Hot swappable components** ○ Components (e.g., memory, CPUs, interface cards, power supplies) can be installed, replaced, or upgraded **while the server is running**. ○ Why it improves availability: It prevents downtime because the system stays operational during ● **Parity memory** 奇偶校验内存 ○ Uses parity bits as a simple error detection code to detect memory data corruption. ○ Why it improves availability: Early detection of memory faults prevents crashes and data corruption that could cause service interruptions. ○ **it can only detect errors, not correct them.** ● **Error-Correcting Code (ECC) memory** ○ Detects and corrects single-bit memory errors automatically using Hamming Code or TMR. ○ Why it improves availability: **Continuous error correction keeps servers stable** and prevents system failure due to memory faults, which are more frequent in 24/7 environments. ● **Virtualization availability** ○ Provides failover clustering where VMs automatically restart on another host if a hardware or OS failure occurs. ○ • Why it improves availability: Ensures minimal service downtime and automatic recovery during failures. ○ • Clustering requires spare or underloaded hosts to take over workloads, maintaining service continuity.

2

Performance

Factors affect compute performance

- Architecture of the server
  - Refers to CPU design, memory hierarchy, and bus interconnections.
  - • How it affects performance: Efficient architecture improves data throughput and reduces bottlenecks between components.
- Speed of the memory and CPU
  - Determines how fast data is processed and accessed.
  - • How it affects performance: Faster CPUs execute instructions more quickly; faster memory reduces data wait time.
- Bus speed
  - Refers to the data transfer rate between CPU, memory, and I/O devices.
  - • How it affects performance: A faster bus minimizes latency when moving data between components.

To improve compute performance

Moore's Law

• States that transistor count doubles every ~2 years.

• Why it affects performance: More transistors enable more processing units and higher efficiency, increasing CPU power — although physical limits are being reached.

- Increasing clock speed (more instructions per second)
  - CPU executes instructions at each clock pulse.
  - • Why it improves performance: Higher clock rates mean more instructions per second, improving compute speed — limited by heat and power constraints.
- Cache memory
  - Small, high-speed memory located on the CPU (L1/L2/L3).
  - • Why it improves performance: Stores frequently accessed data, reducing delays from slower main memory access.
- Pipeline 多个指令阶段可以重叠执行
  - Allows overlapping execution of multiple instruction stages.
  - • Why it improves performance: Increases instruction throughput by utilizing CPU resources continuously.
- Prefetching加载即将到来的指令 and branch prediction预测猜测下一个执行路径
  - Prefetching loads upcoming instructions; branch prediction guesses the next execution path.
  - • Why it improves performance: Reduces idle CPU cycles and cache misses, enabling faster instruction delivery.
- Superscalar CPUs
  - Executes multiple instructions per clock cycle via parallel execution units.
  - • Why it improves performance: Maximizes CPU utilization and throughput.
- Multi-cores CPUs
  - Integrates multiple cores into a single chip.
  - • Why it improves performance: Enables true parallel processing; improves multitasking and workload distribution while reducing heat and energy per operation.
- Hyper-threading
  - Allows one core to execute multiple threads simultaneously.
  - • Why it improves performance: Keeps execution pipelines active, improving CPU efficiency for multi-threaded workloads.
- Virtualization performance
  - Multiple VMs share one physical machine.
  - • Why it affects performance: Efficient consolidation reduces idle time but can cause I/O bottlenecks if overloaded.
  - • High CPU/memory capacity and fast storage improve performance.
  - • Overhead from hypervisor processing is typically <10%.
  - • Databases are I/O-intensive and may require dedicated physical servers or Raw Device Mapping to maintain speed.

3 Security

Physical security

i) Physical server
- USB port
- BIOS setting
- Server housing

Disable external USB ports and **secure BIOS with passwords.**

• Why it improves security: Prevents unauthorized access, data theft, or malware introduction via external devices.

• Detecting case openings can trigger alerts for tampering attempts.

ii) Data in use
- Trusted Execution Environment (TEE)
- Virtualization security

i) To minimize attacks
- Firewalls
- IDS
- Patching
- Minimize complexity of hypervisor

ii) To improve virtualization security
- **De-militarized Zone (DMZ)**
  - The DMZ **isolates external-facing services** (e.g., web servers) **from the internal private network**.
  - How it affects virtualization: It **prevents direct access from the internet to virtual machines** in the internal network, reducing the attack surface and protecting sensitive data or management interfaces.

- **Systems management console**
  - Provides **centralized control and monitoring of all virtual machines and hosts**.
  - How it affects virtualization: **Enables administrators to manage** user permissions, track configurations, detect unauthorized changes, and apply patches or updates consistently, ensuring the virtualized environment stays secure and compliant.

Chapter 7: Operating systems

7.1 Popular Operating Systems

IBM z/OS
IBM i (OS/400)
UNIX
Linux
Berkeley Software Distribution (BSD)
Windows
MacOS
OS for mobile
Special purpose OSs

7.2 Operating systems building blocks

1	OSs building blocks • An operating system allows multiple users, processes, and applications to share hardware and hides hardware complexity. Kernel • The heart of the OS — starts/stops programs, manages files, and controls hardware access to prevent conflicts. Drivers • Connect hardware devices (printers, NICs, keyboards, video) to the kernel. Utilities • Built-in tools like user interfaces, editors, loggers, and update managers. Applications • Software that communicates with the OS through system calls and APIs.
2	OSs functions Process Scheduling • Simulates parallelism by rapidly switching between processes (preemptive multitasking). • Ensures fair CPU allocation through complex scheduling algorithms. File Systems • Organize data in directories and files, managing disk communication and permissions. • Support multiple file system types: FAT, NTFS, UFS, VxFS, Ext (Linux). • Journaling file systems track changes for fast recovery. • File systems must be mounted before use and may use drive letters (Windows) or mount points (UNIX/Linux). • File sharing via NFS (UNIX) or SMB/CIFS (Windows). APIs and System Calls • System calls provide a hardware-independent interface for applications. • APIs define how software can use these system calls (e.g., POSIX for UNIX/Linux, Windows API). Device Drivers • Software that allows the OS to communicate with hardware components through defined APIs. Memory Management

• Allocates/deallocates memory for applications.
• Includes cache, paging, swapping, and DMA (Direct Memory Access) for efficient transfers.
• Prevents memory shortage by moving pages to/from disk as needed.

Shells, CLIs, and GUIs
• User interfaces to interact with the OS.
  – CLI: text-based (bash, sh, cmd.exe).
  – GUI: graphical (Windows, X Windows).

OS Configuration
• Stores settings in databases or text files (Windows Registry, Linux /etc, AIX ODM).
• Editable via configuration tools that simplify text-based settings.

7.3 Operating systems: Availability, Performance & Security

Availability

Failover Clustering – Improves system uptime by automatically transferring workloads from a failed node to a standby node.

- Failover cluster is a group of independent servers running identical operating systems, known as nodes, connected through a network and managed by cluster software.
- Every active application has a standby counterpart on a passive node, which remains idle until a failover occurs.
- When a failure happens, the standby application automatically becomes active and continues serving clients [ensures minimal downtime and uninterrupted service].
- The cluster manages each running application within a node as a package of application components, called a resource pool or application package [organizes applications for automatic monitoring and failover control].

i) Shared storage
- All nodes can access the same data; ensures data continuity during failover.

ii) Shared node
- Provides redundancy; another node takes over automatically during failure.

iii) Voting and quorum disks
- Prevents "split-brain" errors by deciding which node stays active during network disconnections.

iv) Cluster-aware applications
- Run on multiple nodes simultaneously; improve both scalability and failover recovery time.

Performance

Factors affecting OS performance:
i) Hardware performance: Faster CPU, RAM, and storage directly enhance OS responsiveness.
ii) Application load: Heavy workloads slow down the OS; optimization ensures smoother multitasking.
iii) OS configuration: Inefficient settings or unnecessary services waste system resources.

To improve performance:
i) Increase memory: Reduces paging/swapping to disk, enabling faster data access and smoother multitasking.
ii) Decrease kernel size: Frees up RAM, shortens boot time, reduces crash risk, and lowers attack surface.

Why it improves performance:
Optimized memory and kernel management maximize hardware utilization, making the system faster and more stable.

Security

To improve OS security:

- **Patching**: Fixes vulnerabilities, bugs, and design flaws缺陷 to close potential attack vectors.
- **Hardening**: Disables unnecessary services, users, and protocols to reduce attack surface.
    - step by step process of configuring an operating system to protect it against security threats
    - used to instantiate new operating systems. Ensure security is optimal and is consistent in all deployment
- Virus Scanning: Detects and removes malicious software that can harm data or performance.
- Host-Based Firewalls: Filters incoming/outgoing traffic to block unauthorized access.
    - Most operating systems, including Windows, Linux, and UNIX, provide a built-in host-based firewall
- Limiting User Accounts: Minimizes risk from privileged misuse; enforces principle of least privilege.
- Hashed Passwords: Protects credentials by preventing recovery of original passwords.

Why it improves security:

Each measure reduces the likelihood of unauthorized access, malware infection, and privilege abuse, ensuring system integrity and reliability.

Chapter 8: End User Devices

8.1 End user devices building blocks

End user devices are tools humans use to interact with applications.
Examples: desktops, laptops, virtual desktops, mobile devices, printers.
[They connect users to IT infrastructure and deliver input/output functions.]

Desktop & laptop

i) Desktop

- High performance, can store large data and run complex software.
- Issues: complex management, high maintenance cost, and local security risks.
- [Best for stationary, high-power tasks.]

ii) Laptop

- Portable and as powerful as desktops.
- Common risks: theft, damage, and malware from external networks.
- Usually connected to docking stations for power and peripherals.
- [Ideal for mobility but less secure.]

Mobile devices

- Examples: smartphones, tablets, smartwatches, cameras, cars.
- Connect via wireless networks (UMTS/LTE/Wi-Fi).
- Features: small form factor, low bandwidth, variable reliability.
- [Enable mobile access but require adaptive app design and secure connectivity.]

Bring Your Own Device (BYOD)

- Users bring personal devices to access organizational data, instead of using only company-provided devices.
- user paid for the device (they brought their own device), it will not be acceptable to:
  - Have systems managers erase the device (including all family photos or purchased music) in case of an incident
  - Have personal data visible to the systems managers
- Conflict: user freedom vs. IT security control.
- **Solution**:
  - **virtualization(personal + work vm)** + Mobile Device Management (MDM) → separates personal and work environments.
  - **Mobile Device Management (MDM)**
    - to monitor organization data on personal devices.
    - It can enforce password strength, device encryption, and **remote wipe** in case of device loss or theft.
  - **Network Access Control and segmentation**
    - require devices to **meet security compliance standards (updated OS**, antivirus, encryption) before connecting to internal systems.
    - **Provide DMZ or VLAN for personal devices to connect,** instead of the core banking network.

Printers

1. Laser Printers: use toner + drum + laser → high quality, fast.
2. Inkjet Printers: spray ink droplets → cheaper, energy-efficient, high-quality color prints.
3. Multi-Functional Printers (MFPs):
   combine printer, scanner, copier, fax; include OS, storage, and network features → require patching and data protection.
4. Specialized Printers:
   a. Dot Matrix: uses pins; reliable, low cost, noisy.
   b. Line Printer: prints full lines; durable for industrial use.
   c. Thermal Printer: uses heat-sensitive paper; quiet, compact, fast, but prints fade.

8.2 Desktop virtualization

Technology	Runs On	User Rss	Cost Model	Key Benefit
Application Virtualization	Local or Server	Shared OS	License-based	Compatibility & easy deployment
SBC	Server	Shared VM	Per-user/server license	Centralized management
VDI	Server(VM per user)	Dedicated VM	Pay-per-use (cloud)	Full OS isolation
Thin/Zero Client	Minimal hardware	Server resources	Low device cost	Easy to manage, low maintenance
PXE Boot	Network OS image	Network-based	Shared infra	Diskless, fast deployment

Application virtualization

- Runs apps in isolated virtual environments.
- OS resources are virtualized, not the app itself.
- Requests and file operations are redirected to virtualized locations.
- **[Allows running old/incompatible apps side-by-side and easier migration.]**
- Examples: Microsoft App-V, VMware ThinApp.

Server based computing (SBC)

- Applications/desktops **run on remote servers**.
- **Only display updates, keyboard, and mouse data are exchanged**.
- [Low bandwidth, centralized management.]
- Products: Windows RDS, Citrix XenApp.
- Advantages: Easier maintenance and consistent configuration.
- Disadvantages: **Depends on network latency**, **limited local performance**.

Virtual Desktop Infrastructure (VDI)

- Each user runs a full desktop OS in their own VM.
- Managed by a hypervisor that allocates resources.
- **[Provides isolation but uses more CPU/storage than SBC.]**
- Available as cloud services (Azure Virtual Desktop, Amazon WorkSpaces, Google Cloud VDI).
- Issue: ***"Logon storm" when many users start VMs simultaneously***.

Thin clients (2)

- Lightweight devices connecting to SBC/VDI servers.
- **No local storage or configuration**; easy to replace.
- [Reduce cost, maintenance, and security risk.]

i) **Zero client**

- No local OS; relies entirely on the server for boot and operation.

ii) **Preboot eXecution Environment (PXE) boot**

- Boots OS over network instead of local disk; used for diskless workstations.
- Requires constant network connection.
- [Useful for managed office environments but unsuitable for mobile devices.]

8.3 End user devices: Availability, Performance & Security

Availability

To improve end user device's availability:

- Use **high-quality components** to reduce failure rates.
- Perform **regular maintenance** (software updates, cleaning, battery checks).
- Add **redundancy** (spare devices, backup data copies).

Performance

To improve end user devices' performance:

- **Add more RAM** – improves multitasking and system speed.
- **Use SSD instead of HDD** – faster data access and startup.
- **Ensure sufficient network bandwidth** for all users.
- **Optimize mobile apps** for low-bandwidth or offline operation.
- Use **Server-Based Computing (SBC)** to reduce the effect of slow connections.

Security

Challenges:

- Devices are **spread across offices, homes, and client sites**.
- Not locked in a datacenter, thus prone to **theft, malware, and misuse**.

To improve security:

i) General Protection

- Use **laptop cable locks** to prevent theft.
- Install **malware protection** (antivirus, firewall).
- Enable **full-disk encryption** to protect data.
- Erase disks completely before disposal or repair.

ii) Mobile Device Management (MDM)

- **Monitor, maintain, and secure** mobile devices remotely.
- **Remote wipe** feature to delete data on lost or stolen devices.
- **Locate** stolen devices via tracking software.

iii) End User Authorization & Awareness

- Restrict user privileges (no admin rights).
- Set **BIOS passwords** and disable USB/DVD booting.
- Conduct **security awareness training** (social engineering, strong passwords, handling sensitive data).

iv) Network Access Control (NAC)

- Control network access based on:
    - **Device identity** (known/trusted device)
    - **User/group roles**
    - **Compliance status** (latest antivirus, OS patches)
- Non-compliant devices are placed in an **isolated network** until updated.

C9: Infrastructure Management

9.1 Infrastructure Deployment Options

Infrastructure deployment models (4)

On-Premises You own and manage all hardware and software in your own data center. Requires sufficient space, UPS, cooling, fire safety, redundant networks, and strong floor load capacity. • Large enterprises with strict data control needs • Industries with regulatory requirements (e.g., finance, defense)	✅ Full control and high security ❌ 1. High Costs – Expensive setup and maintenance (hardware, licenses, IT staff, energy, cooling). Vendor lock-in risk. 2. Poor Scalability – Scaling requires physical installation and manual setup. 3. Limited Redundancy & Disaster Recovery – Single points of failure; complex and costly DR solutions. 4. Security Burden – Needs dedicated IT security, tools, and staff.
Public Cloud You rent computing resources (e.g., servers, storage) from providers like AWS, Azure, or GCP. Management level depends on the model (IaaS, PaaS, SaaS). • Startups or new businesses (greenfield projects) • Workloads that need rapid scaling or global reach	• Cost-effective and scalable • Fast innovation and deployment • Pay-as-you-go model • Less control and data security • Vendor dependency • Requires reliable internet connection
Private Cloud Infrastructure dedicated to one organization, either hosted internally or by a provider. Functions as a software-defined data center (SDDC) with automation and orchestration. • Enterprises requiring strict security and compliance • Organizations needing isolated environments with virtualization	• High control, privacy, and customization • Automated management (costing, reporting, scaling) • Comparable to IaaS model • High initial investment • Limited scalability • No true pay-per-use model
Hybrid Cloud Combines on-premises or private cloud with public cloud for flexibility and phased migration. Sensitive data stays private, while non-critical workloads use public resources. • Enterprises transitioning to cloud • Workloads needing both secure storage and dynamic scaling	• Flexibility and cost optimization • Enables gradual migration • Best of both environments • Complex integration and management • Requires dual-environment expertise • Mixed cost model (cloud + on-premises)

9.2 Infrastructure Automation

Configuration Management Tools

- Automate the configuration of servers, network devices, and infrastructure components, ensuring consistency.
- Continuously check systems against blueprints; if deviations occur (e.g., missing software, changed settings), tools automatically correct them.

Infrastructure as Code (IaC)

Treats infrastructure like software code — configurations are defined in code files to automate deployment and management.

Terraform (by HashiCorp) is an open-source IaC tool using HashiCorp Configuration Language (HCL) or JSON to manage infrastructure across cloud and on-premises environments.

Version Control

Tracks code changes over time using repositories that create new versions automatically when updates are pushed.

Tools: Git (distributed CLI tool), GitHub (web-hosted Git), GitLab (self-hosted Git platform).

Orchestration Tools

Coordinate and automate complex workflows across multiple infrastructure components.
Act like a conductor ensuring all parts work in harmony (e.g., for deployments or scaling).
Key Functions:

- Workflow Automation – Automates provisioning, configuration, deployment, and scaling.
- Dependency Management – Handles retries, rollbacks, and alternative paths when tasks fail.
- Resource Coordination – Manages resources efficiently across cloud and on-premises platforms.

Cloud Management Platforms (CMPs)

Provide centralized automation for provisioning, monitoring, and managing cloud resources.

9.3 Infrastructure Documentation

Infrastructure documentation tools / techniques

- Documentation ensures reliability, security, and effective management.
- Preserves knowledge when staff leave, aids troubleshooting, and supports disaster recovery by showing how infrastructure is configured and functions.

Configuration Management Database (CMDB)	Inventory of hardware, software, and networking components, detailing make, model, location, and function. Supports ITIL processes; should be updated regularly (preferably automated). Enables correlation of components to identify changes or root causes of failures.
Diagrams	Topology Diagrams – Show relationships and connections between components. Tools: Microsoft Visio, Diagrams.net. ArchiMate – Open standard for modeling enterprise architecture; describes IT systems, business processes, and information flows.
IaCs tools	IaC can serve as living documentation, showing how infrastructure is built and why. Documentation updates can be made alongside code changes. Advantage: Always up-to-date with modifications. Disadvantage: Requires reading code; doesn't provide instant visual overview like diagrams.
Documenting procedures	1. Procedures for routine tasks 2. Infrastructure naming convention 3. IP addressing plan 4. DNS naming convention 5. Fallback procedure 6. Disaster recovery plan