Information Technology Infrastructure

Microsoft CERTIFIED Systems Administrator

Microsoft CERTIFIED Systems Engineer

IT-602 MIDTERM

Topics (1-6 Weeks)

Compiled by : YASIR EJAZ

(MCP, MCSE, MCSA)

0321-5253058

0321-5620799

yasirejaz@gmail.com

Information Technology Infrastructure

Information Technology Infrastructure

- **Credit Hours:** 3
- **Lecturer: Shafaq Nisar**

- **Lecturing Style:** Video Lectures of short duration (5-7 minutes)

The definition of IT infrastructure

1. Introduction
2. IT Infrastructure
3. The Infrastructure Model
4. Non-functional Attributes
5. Non-functional Requirements

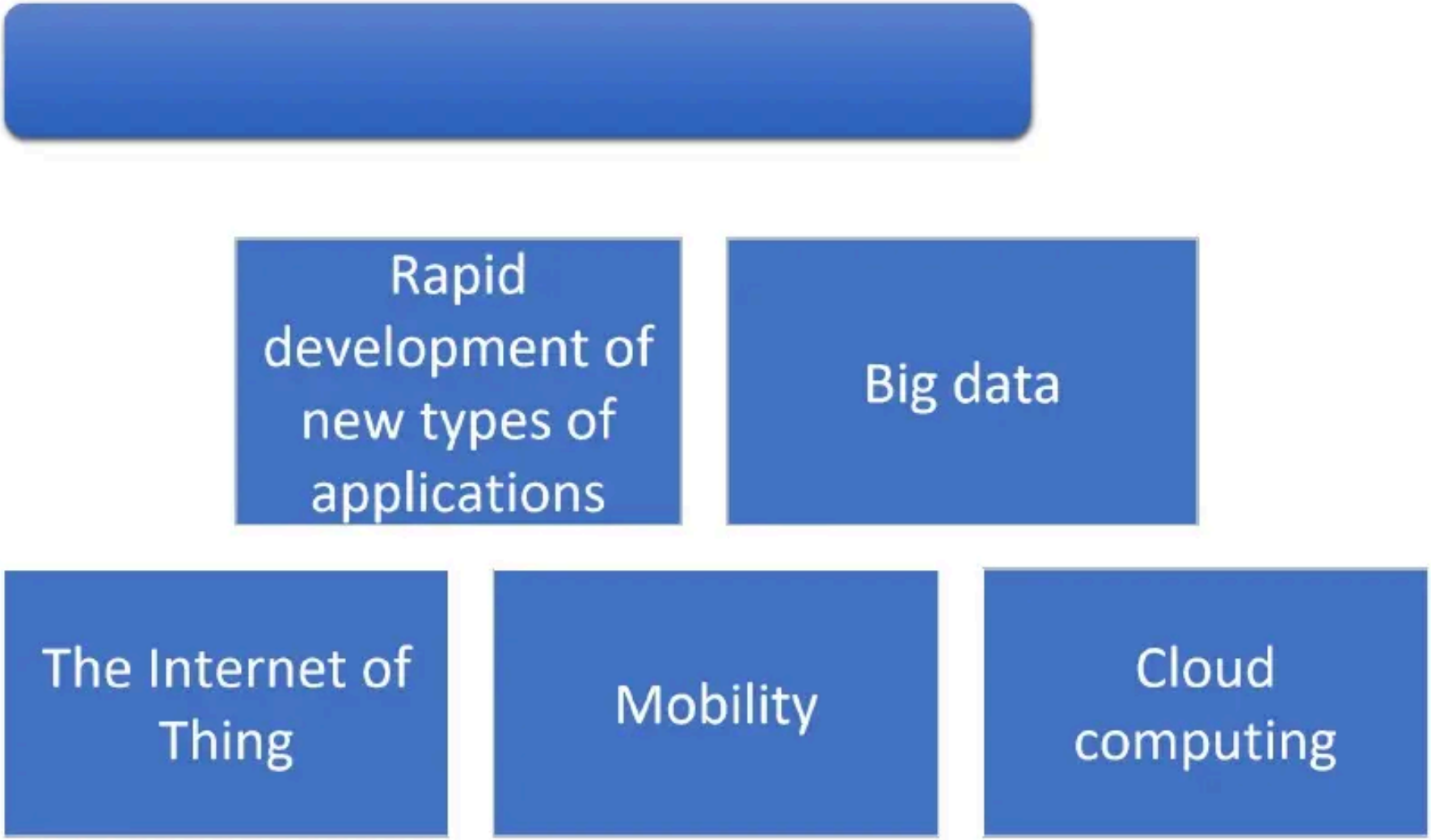
Topic 1
Introduction of IT Infrastructure

What is IT Infratsructure?

Infratsructure

Infrastructure

IT Infratsructure


Rapid development of new types of applications
Big data
The Internet of Thing
Mobility
Cloud computing

Various Kinds of Architecture

- Business architecture
- Enterprise architecture
- Data architecture,
- Application architecture and
- Infrastructure architecture

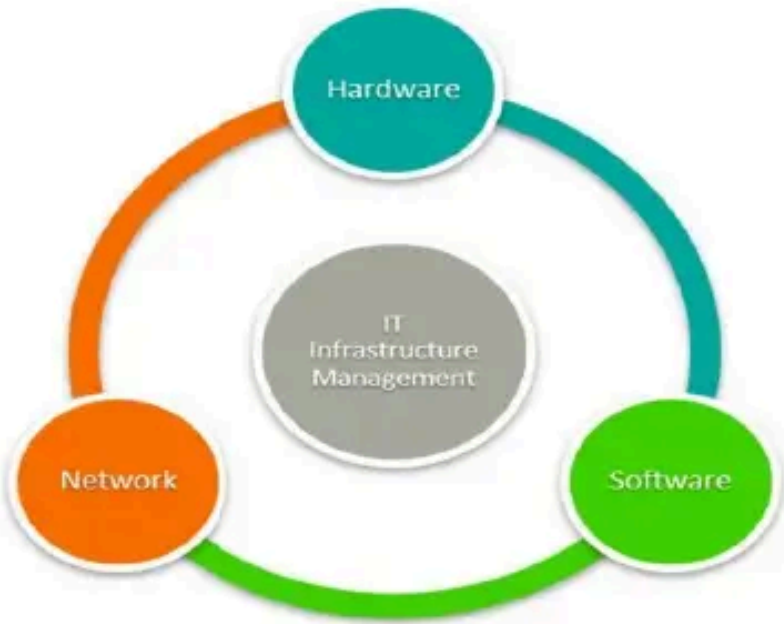
Is there a General Definition of IT Infrastructure?

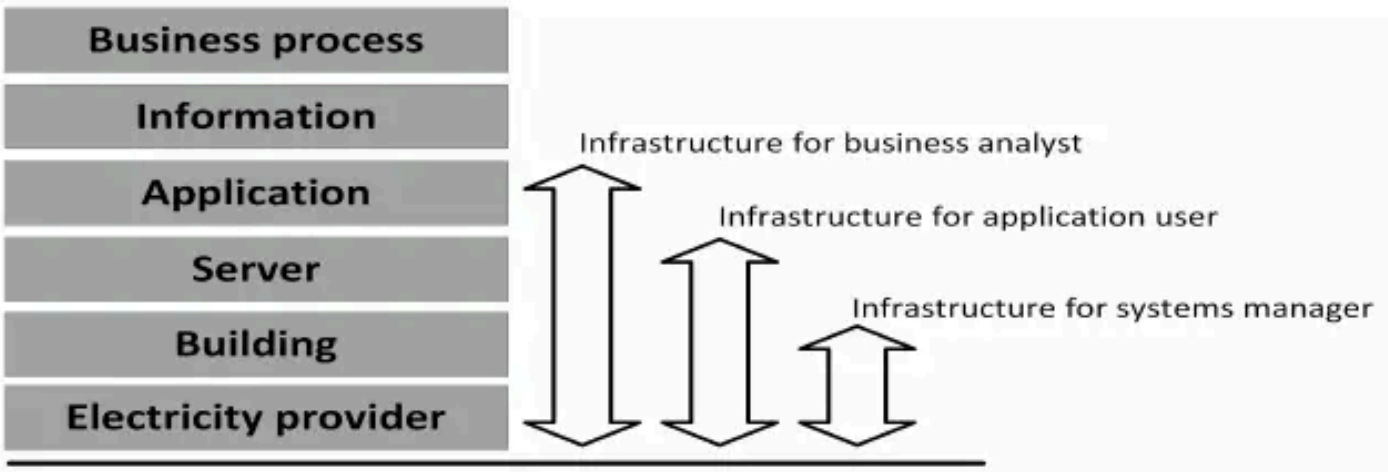
No generally accepted definition of IT infrastructure seems to exist

In literature, many definitions of IT infrastructure are described. Some of them are:

IT infrastructure consists of the equipment, systems, software, and services used in common across an organization, regardless of mission/program/project. IT Infrastructure also serves as the foundation upon which mission/program/project-specific systems and capabilities are built.

Traditional IT Infrastructure




Business process
Information
Application
Server
Building
Electricity provider
Infrastructure for business analyst
Infrastructure for application user
Infrastructure for systems manager

Information Technology Infrastructure

Information Technology Infrastructure

- **Credit Hours:** 3
- **Lecturer:** Shafaq Nisar

- **Lecturing Style:** Video Lectures of short duration (5-7 minutes)

Introduction to Non-Functional Attributes

- IT infrastructure provides services to applications
- Many of these services can be defined as functions such as
  - Disk space,
  - Processing,
  - Connectivity

  However most of these services are non functional in nature

Non functional attributes describe the qualitative behavior of the system rather than its specific functionality and these include

- Availability
- Security
- Performnace
- Recoverability
- Testtability
- Scalability

Handling Conflicting NFRs

It is unsual to encounter conflictingNFRs for instance users may want a system that is secure but not want to be bothered by passwords

- It is the task of the infrastructure architect to balance these NFRs, in some cases some NFRs may take priority over others and the architect must involve the relevant stakeholders

Information Technology Infrastructure

Information Technology Infrastructure

- **Credit Hours:** 3
- **Lecturer: Shafaq Nisar**

- **Lecturing Style:** Video Lectures of short duration (5-7 minutes)

1. Introduction
2. Calculation of Availability
3. Mean Time Between Failures (MTBF)
4. Sources of Unavailability
5. Types of erros
6. Environmental issues
7. Complexity of the infrastructure
8. Availability Patterns
9. Redundancy, Failover
10. Business Continuity

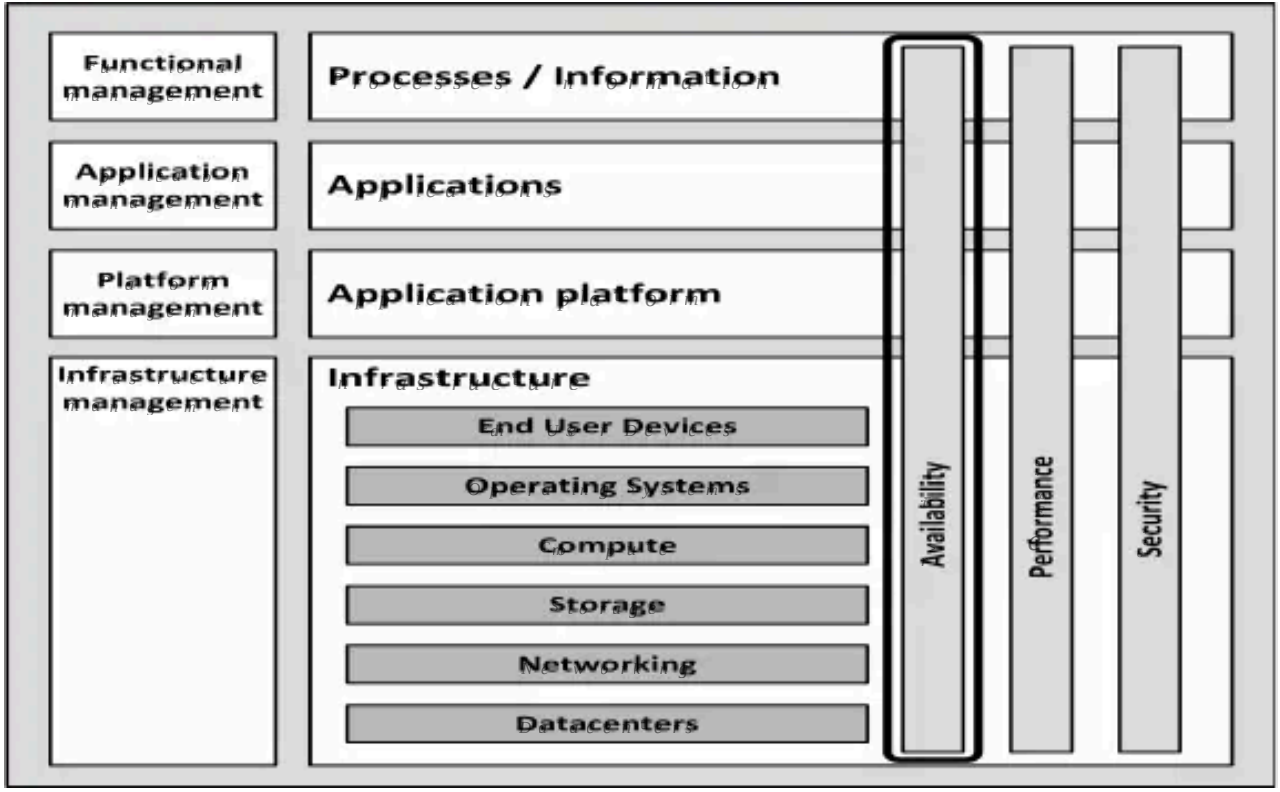
Introduction

AVAILABILITY CONCEPTS

Introduction

- Everyone expects their infrastructure to be available all the time
- A 100% guaranteed availability of an infrastructure is impossible
- **A fact of life**
  - There is always a chance of downtime.

Introduction



Calculation of Availability

- Availability can neither be calculated, nor guaranteed upfront
  - It can only be reported on afterwards, when a system has run for some years
- Over the years, much knowledge and experience is gained on how to design high available systems
  - Failover
  - Redundancy
  - Structured programming
  - Avoiding Single Points of Failures (SPOFs)
  - Implementing systems management

Calculation of Availability

- The availability of a system is usually expressed as a percentage of uptime in a given time period, usually one year or one month
- Example for downtime expressed as a percentage per year

Availability %	Downtime per year	Downtime per month	Downtime per week
99.8%	17.5 hours	86.2 minutes	20.2 minutes
99.9% ("three nines")	8.8 hours	43.2 minutes	10.1 minutes
99.99% ("four nines")	52.6 minutes	4.3 minutes	1.0 minutes
99.999% ("five nines")	5.3 minutes	25.9 seconds	6.1 seconds

Typical requirements used in service level agreements today are 99.8% or 99.9% availability per month for a full IT system

- The availability of the infrastructure must be much higher
  - ➢ Typically in the range of 99.99% or higher

Calculation of Availability

- 99.999% uptime is also known as carrier grade availability
  - For one component, higher availability levels for a complete system are very uncommon, as they are almost impossible to reach

Calculation of Availability

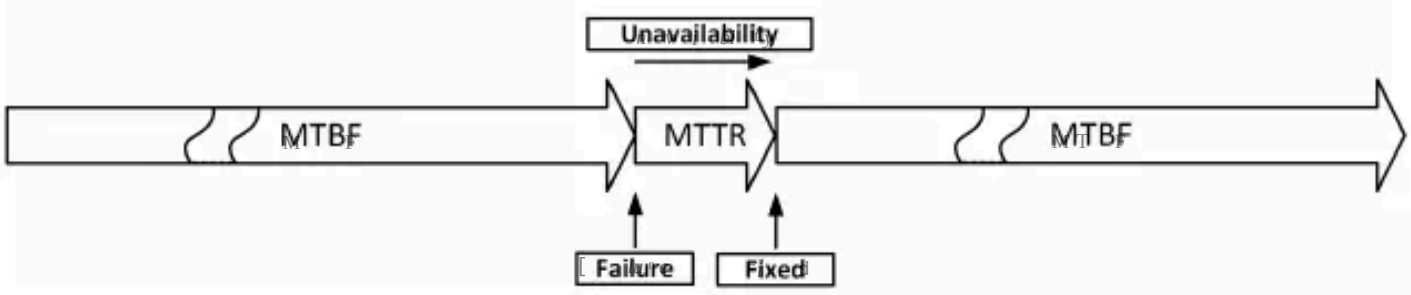
- It is a good practice to agree on the maximum frequency of unavailability

Unavailability (minutes)	Number of events (per year)
0 – 5	<= 35
5 – 10	<= 10
10 – 20	<= 5
20 – 30	<=2
> 30	<= 1

MTBF and MTTR

- Mean Time Between Failures (MTBF)
  - The average time that passes between failures

- Mean Time To Repair (MTTR)
  - The time it takes to recover from a failure



MTBF and MTTR

- Some components have higher MTBF than others

- Some typical MTB's:

Component	MTBF (hours)
Hard disk	750,000
Power supply	100,000
Fan	100,000
Ethernet Network Switch	350,000
RAM	1,000,000

- MTTR can be kept low by:
  - Having a service contract with the supplier
  - Having spare parts on-site
  - Automated redundancy and failover

MTTR

- Steps to complete repairs:
  - Notification of the fault (time before seeing an alarm message)
  - Processing the alarm
  - Finding the root cause of the error
  - Looking up repair information
  - Getting spare components from storage
  - Having technician come to the datacenter with the spare component
  - Physically repairing the fault
  - Restarting and testing the component

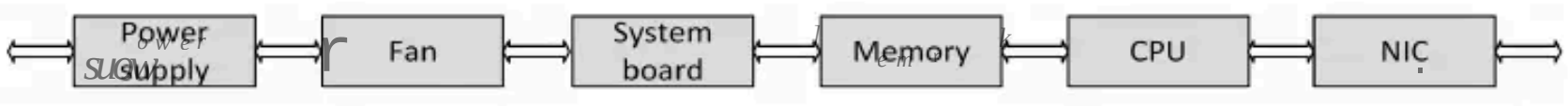
Calculation Examples

$$\text{Availability} = \frac{\text{MTBF}}{(\text{MTBF} + \text{MTTR})} \times 100\%$$

Component	MTBF (h)	MTTR (h)	Availability	in %	
Power supply	100,000	8	0.9999200	99.99200	
Fan	100,000	8	0.9999200	99.99200	
System board	300,000	8	0.9999733	99.99733	
Memory	1,000,000	8	0,9999920	99.99920	
CPU	500,000	8	0.9999840	99.99840	
Network Interface Controller (NIC)	250,000	8	0.9999680	99.99680	

Calculation Examples

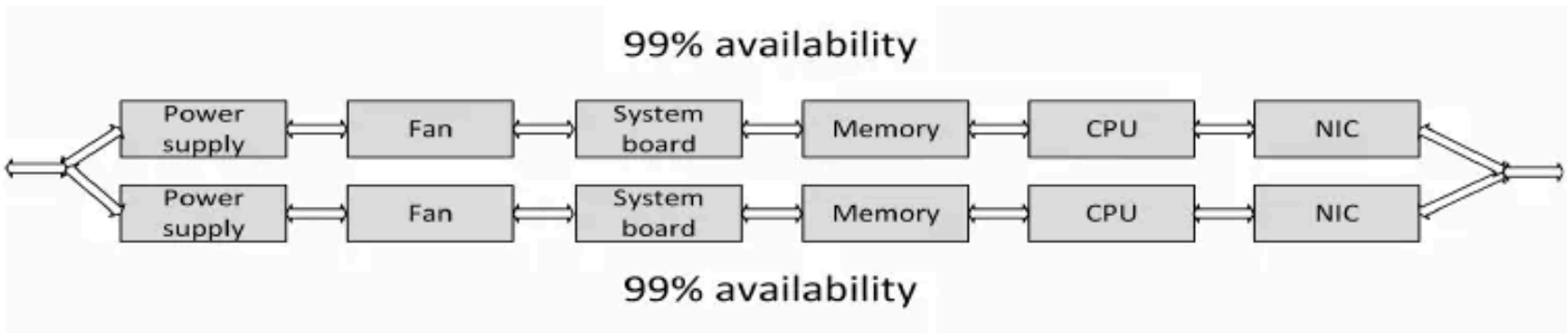
- Serial components: One defect leads to downtime



- Example: the above system's availability is:

$$0.9999200 \times 0.9999200 \times 0.9999733 \times 0.9999920 \times 0.9999840 \times 0.9999680 = 0.99977 = \square\square.\square\square\square\%$$

(each components' availability is at least 99.99%)


99% availability
Power supply
Fan
System board
Memory
CPU
NIC
Power supply
Fan
System board
Memory
CPU
NIC
99% availability

- 80% of outages impacting mission-critical services is caused by people and process issues
- Examples:
  - Performing a test in the production environment
  - Switching off the wrong component for repair
  - Swapping a good working disk in a RAID set instead of the defective one
  - Restoring the wrong backup tape to production
  - Accidentally removing files
    - Mail folders, configuration files
  - Accidentally removing database entries
    - Drop table x instead of drop table y

Sources of Unavailability - Software Bugs

- Because of the complexity of the software, it is nearly impossible (and very costly) to create bug-free software
- Application software bugs can stop an entire system
- Operating systems are software too
  - Operating systems containing bugs can lead to
    - corrupted file systems,
    - network failures, or
    - other sources of unavailability

Sources of Unavailability - Planned Maintenance

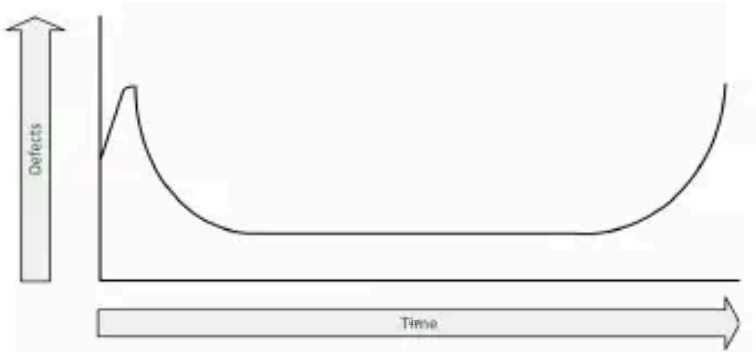
- Sometimes needed to perform systems management tasks:
  - Upgrading hardware or software
  - Implementing software changes
  - Migrating data
  - Creation of backups
- During planned maintenance the system is more vulnerable to downtime than under normal circumstances
  - A temporary SPOF could be introduced
  - Systems managers could make mistakes

Sources of Unavailability - Physical Defects

- Everything breaks down eventually
- Mechanical parts are most likely to break first
- Examples:
  - **Fans for cooling equipment** usually break because of dust in the bearings
  - **Disk drives** contain moving parts
  - **Tapes** are very vulnerable to defects as the tape is spun on and off the reels all the time
  - **Tape drives** contain very sensitive pieces of mechanics that can break easily

Sources of Unavailability - Bathtub Curve

- A component failure is most likely when the component is new
- Sometimes a component doesn't even work at all when unpacked for the first time. This is called a DOA component – Dead On Arrival.
- When a component still works after the first month, it is likely that it will continue working without failure until the end of its life



- Environmental issues can cause downtime. Issues with
  - Power
  - Cooling
  - External factors like:
  - Disasters
  - Fire
  - Earthquakes
  - Flooding

Sources of Unavailability - Complexity of the Infrastructure

- Adding more components to an overall system design can undermine high availability
  - Even if the extra components are implemented to achieve high availability
- Complex systems
  - Have more potential points of failure
  - Are more difficult to implement correctly
  - Are harder to manage
- Sometimes it is better to just have an extra spare system in the closet than to use complex redundant systems

Availability Patterns

- A single point of failure (SPOF) is a component in the infrastructure that, if it fails, causes downtime to the entire system.
- SPOFs should be avoided in IT infrastructures as they pose a large risk to the availability of a system.
- We just need to know what is shared and if the risk of sharing is acceptable.
- To eliminate SPOFs, a combination of redundancy, failover, and fallback can be used.

Redundancy

- Redundancy is the duplication of critical components in a single system, to avoid a single point of failure (SPOF)
- Examples:
  - A single component having two power supplies; if one fails, the other takes over
  - Dual networking interfaces
  - Redundant cabling

Failover

- Failover is the (semi)automatic switch-over to a standby system or component
- Examples:
  - Windows Server failover clustering
  - VMware High Availability
  - Oracle Real Application Cluster (RAC) database

- Fallback is the mannual switchover to an identical standby computer system in a different location
- Typically used for disaster recovery
- Three basic forms of fallback solutions:
  - Hot site
  - Cold site
  - Warm site

Business Continuity

- In case of a disaster, the infrastructure could become unavailable, in some cases for a longer period of time.
- Business continuity is about identifying threats an organization faces and providing an effective response.
- To handle the effect of disasters, following processes are
  - Business Continuity Management (BCM) and
  - Disaster Recovery Planning (DRP)

Business Continuity

- An IT disaster is defined as an irreparable problem in a datacenter, making the datacenter unusable

Natural disasters:	Manmade disasters:
Floods	Hazardous material spills
Hurricanes	Infrastructure failure
Tornadoes	Bio-terrorism
Earthquakes	

Information Technology Infrastructure

Information Technology Infrastructure

- **Credit Hours:** 3
- **Lecturer: Shafaq Nisar**

- **Lecturing Style:** Video Lectures of short duration (5-7 minutes)

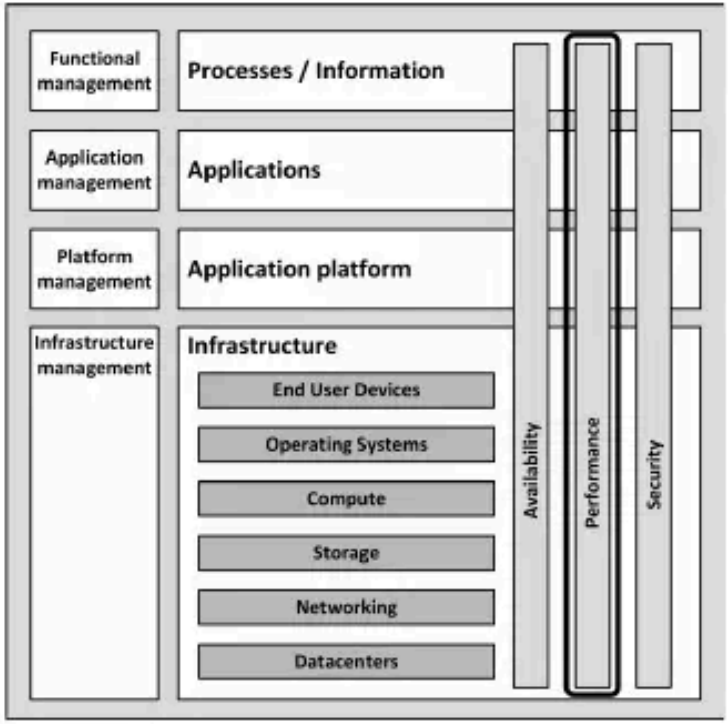
Performance Concepts

1. Introduction
2. Perceived Performance
3. Performance during Infrastructure Design
   - Benchmarking, User Profiling, Vendor experience
   - Prototyping
4. Performance of a Running System
   - Managing Bottlenecks and Performance Testing
5. Performance Patterns
   - Increasing performance on Upper layer
   - Caching
6. Grid Computing
7. Capacity management

Introduction

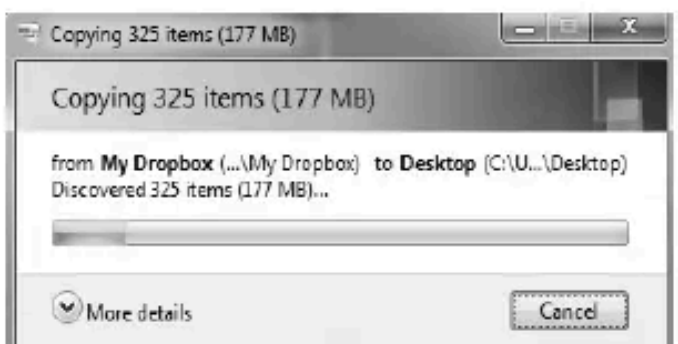
Performance Concepts

Introduction



Perceived Performance

- Perceived performance refers to how quickly a system appears to perform its task
- In general, people tend to overestimate their own patience
- People tend to value predictability in performance
  - When the performance of a system is fluctuating, users remember a bad experience
  - Even if the fluctuation is relatively rare


Copying 325 items (177 MB)
Copying 325 items (177 MB)
from My Dropbox (...\My Dropbox) to Desktop (C:\U...\Desktop)
Discovered 325 items (177 MB)...
More details
Cancel


Office
Word
Starten...

Performance during Infrastructure Design

- A solution must be designed, implemented, and supported to meet the performance requirements

  Even under increasing load

- Calculating performance of a system in the design phase is:

  Extremely difficult

  Very unreliable

Performance during Infrastructure Design

- Performance must be considered:
  - When the system works as expected
  - When the system is in a special state, like:
    - Failing parts
    - Maintenance state
    - Performing backup
    - Running batch jobs

Performance during Infrastructure Design

- Some ways to do this are:
  - Benchmarking
  - Using vendor experience
  - Prototyping and User Profiling

Benchmarking

- A benchmark uses a specific test program to assess the relative performance of an infrastructure component
- Benchmarks compare:
  - Performance of various subsystems
  - Across different system architectures

- Benchmarks comparing the raw speed of parts of an infrastructure
  - Like the speed difference between processors or between disk drives
  - Not taking into account the typical usage of such components
  - Examples:
    - Floating Point Operations Per Second – FLOPS
    - Million Instructions Per Second – MIPS of a CPU

Prototyping

- Also known as proof of concept (PoC)
- Prototypes measure the performance of a system at an early stage
- Building prototypes:
  - Hiring equipment from suppliers
  - Using data centre capacity at a vendor's premise
  - Using cloud computing resources
- Focus on those parts of the system that pose the highest risk, as early as possible in the design process

Vendor Experience

- The best way to determine the performance of a system in the design phase: use the experience of vendors
- They have a lot of experience running their products in various infrastructure configurations
- Vendors can provide:
  - Tools
  - Figures
  - Best practices

User Profiling

- Predict the load a new software system will pose on the infrastructure before the software is actually built
- Get a good indication of the expected usage of the system
- Steps:
  - Define a number of typical user groups (personas)
  - 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

Performance of a Running System

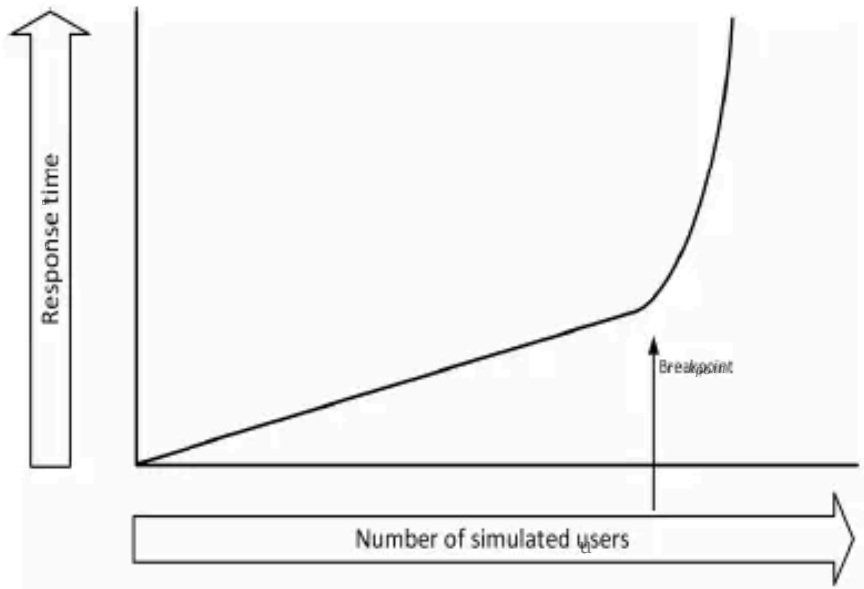
- The performance of a system is based on:
  - The performance of all its components
  - The interoperability of various components
- A component causing the system to reach some limit is referred to as the bottleneck of the system
- Every system has at least one bottleneck that limits its performance
- If the bottleneck does not negatively influence performance of the complete system under the highest expected load, it is OK

Performance Testing

- **Load testing** - shows how a system performs under the expected load
- **Stress testing** - shows how a system reacts when it is under extreme load
- **Endurance testing** - shows how a system behaves when it is used at the expected load for a long period of time

Performance Testing - Breakpoint

- Ramp up the load
  - Start with a small number of virtual users
  - Increase the number over a period of time
- The test result shows how the performance varies with the load, given as number of users versus response time.



Performance Testing

- Performance testing software typically uses:
  - One or more servers to act as injectors
    - Each emulating a number of users
    - Each running a sequence of interactions
  - A test conductor
    - Coordinating tasks
    - Gathering metrics from each of the injectors
    - Collecting performance data for reporting purposes

Performance Testing

- Performance testing should be done in a production-like environment
  - Performance tests in a development environment usually lead to results that are highly unreliable
  - Even when underpowered test systems perform well enough to get good test results, the faster production system could show performance issues that did not occur in the tests
- To reduce cost:
  - Use a temporary (hired) test environment

Performance Patterns

- 80% of the performance issues are due to badly behaving applications
- Application performance can benefit from:
  - Database and application tuning
  - Prioritizing tasks
  - Working from memory as much as possible (as opposed to working with data on disk)
  - Making good use of queues and schedulers
- Typically more effective than adding compute power

Disk Caching

- Disks are mechanical devices that are slow by nature
- Caching can be implemented i:
  - Disks
  - Disk controllers
  - Operating system
- Cache memory:
  - Stores all data recently read from disk
  - Stores some of the disk blocks following the recently read disk blocks

Caching

Component	**Time it takes to fetch 1 MB of data (ms)**
Network, 1 Gbit/s	675
Hard disk, 15k rpm, 4 KB disk blocks	105
Main memory DDR3 RAM	0.2
CPU L1 cache	0.016

Web Proxies

- When users browse the internet, data can be cached in a web proxy server
  - A web proxy server is a type of cache
  - Earlier accessed data can be fetched from cache, instead of from the internet
- Benefits:
  - Users get their data faster
  - All other users are provided more bandwidth to the internet, as the data does not have to be downloaded again

Grid Computing

- A computer grid is a high performance cluster that consists of systems that are spread geographically
- The limited bandwidth is the bottleneck
- Examples:
  - SETI@HOME
  - CERN LHC Computing Grid (140 computing centers in 35 countries)
- Broker firms exist for commercial exploitation of grids
- Security is a concern when computers in the grid are not under control

Capacity Management

- Capacity management guarantees high performance of a system in the long term
- To ensure performance stays within acceptable limits, performance must be monitored
- Trend analyses can be used to predict performance degradation
- Anticipate on business changes (like forthcoming marketing campaigns)

Information Technology Infrastructure

Information Technology Infrastructure

Credit Hours: 3

Lecturer: **Shafaq Nisar**

Lecturing Style: Video Lectures of short duration (5-7 minutes)

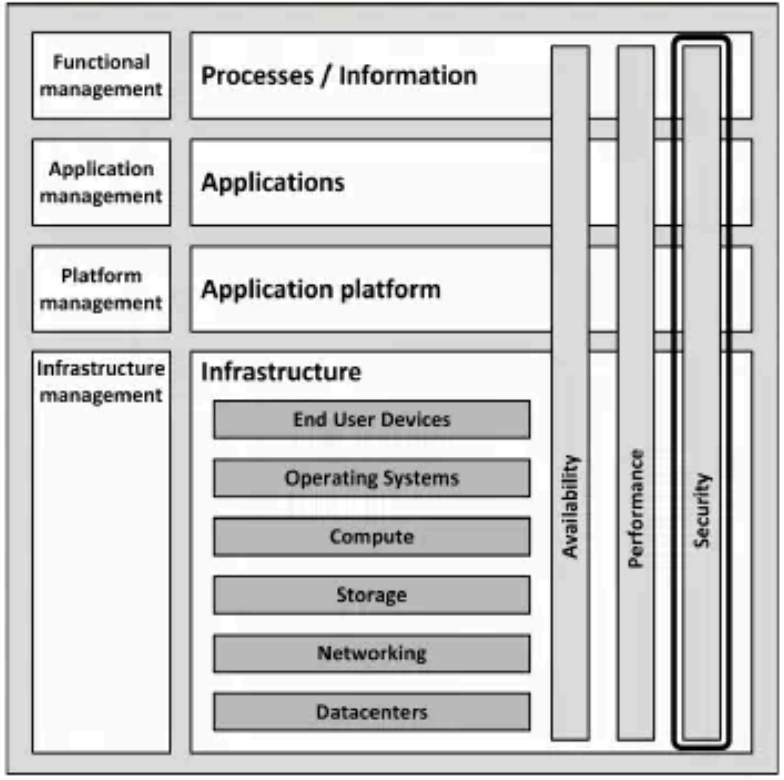
Security Concepts

1. Introduction
2. Computer Crimes
3. Risk Management
   - Risk Response
   - Exploits
4. Security Controls
   - Attack vectores
5. Security Patterns
   - Identity and Access management and Layered security
   - Cryptography and Cryptographic Attacks

Introduction

Security Concepts

Introduction



Computer Crimes

Reasons for committing crime against IT infrastructures:

- Personal exposure and prestige
- Creating damage
- Financial gain
- Terrorism
- Warfare

Personal Exposure and Prestige

In the past, the hacker community was very keen on getting personal or group exposure by hacking into a secured IT infrastructure. When hackers proved that they could enter a secured system and made it public, they gained respect from other hackers.

While nowadays most hacking activity is done for other reasons, there are still large communities of hackers that enjoy the game.

Creating Damage

Creating damage to organizations to create bad publicity

For instance, by defacing websites, bringing down systems or websites, or

making internal documents public

Financial Gain

For instance, by holding data hostage and asking for ransom money, stealing credit card data, changing account data in bank systems

OR

Stealing passwords of customers and ordering goods on their behalf

Terrorism

The main purpose of terrorism is creating fear in a society

A well-planned attack targeted at certain computer systems, like the

Computer system that manages the water supply

or

A nuclear power plant, could result in chaos and fear amongst citizens

Warfare

Certain governments use hacking practices as acts of war

Since economies and societies today largely depend on the IT infrastructures, bringing important IT systems down in a certain country could cause the economy to collapse.

Bringing down the internet access of a country for example means: no access to social media, no e-mails, no web shops, no stock trading, no search engines, etc.

Risk management

Risk management

- Managing security is all about managing risks
- The effort we put in securing the infrastructure should be directly related to the risk at hand
- Risk management is the process of:
  - Determining an acceptable level of risk
  - Assessing the current level of risk
  - Taking steps to reduce risk to the acceptable level
  - Maintaining that level

Risk list

A risk list can be used to quantify risks

Risk is calculated based on:

Asset name - component that needs to be protected

Vulnerability - weakness, process or physical exposure that makes the asset susceptible to exploits

Exploit - a way to use one or more vulnerabilities to attack an asset

Probability - an estimation of the likelihood of the occurrence of an exploit

Impact - the severity of the damage when the vulnerability is exploited

Risk Response

Controls can be designed and implemented based on identified severity of the risk in the risk list.

There four risk responses:

- Acceptance of the risk
- Avoidance of the risk - do not perform actions that impose risk
- Transfer of the risk - for instance transfer the risk to an insurance company
- Mitigation of the risk and accepting the residual risk

Exploits

Information can be stolen in many ways

Examples:

- Key loggers can send sensitive information like passwords to third parties
- Network sniffers can show network packages that contain sensitive information or replay a logon sequence
- Data on backup tapes outside of the building can get into wrong hands
- Disposed PCs or disks can get into the wrong hands
- Corrupt or dissatisfied staff can copy information
- End users are led to a malicious website that steals information (phishing)

Security Controls

CIA

Three core goals of security (CIA):

- Confidentiality
- Integrity
- Availability

Confidentiality - prevents the intentional or unintentional unauthorized disclosure of data

Integrity - ensures that:

- No modifications to data are made by unauthorized staff or processes
- Unauthorized modifications to data are not made by authorized staff or processes
- Data is consistent

Availability - ensures the reliable and timely access to data or IT resources

CIA

Example of confidentiality levels

Confidentiality Level	Description
1	Public information
2	Information for internal use only
3	Information for internal use by restricted group
4	Secret: reputational damage if information is made public
5	Top secret: damage to organization or society if information is made public

CIA

Example of integrity levels

Integrity Level	Description
1	Integrity of information is of no importance
2	Errors in information are allowed
3	Only incidental errors in information are allowed
4	No errors are allowed, leads to reputational damage
5	No errors are allowed, leads to damage to organization or society

CIA

Example of availability levels

Availability Level	Description
1	No requirements on availability
2	Some unavailability is allowed during office hours
3	Some unavailability is allowed only outside of office hours
4	No unavailability is allowed, 24/7/365 availability, risk for reputational damage
5	No unavailability is allowed risk for damage to organization or society

Controls mitigate risks

Security controls must address at least one of the CIA

Information can be classified based on CIA levels

Controls can be designed and implemented based on the identified risk level for CIA

Control	C 1	C 2	C 3	C 4	C 5	I 1	I 2	I 3	I 4	I 5	A 1	A 2	A 3	A 4	A 5
Standard security policy	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
Central archiving of documents		X	X	X	X										
User based password protection		X	X	X	X		X	X	X	X		X	X	X	X
Anti-virus measures		X	X	X	X		X	X	X	X		X	X	X	X
Screensaver lock when leaving workplace			X	X	X			X	X	X					
Webmail not allowed			X	X	X										
Logging of authentication and authorization requests			X	X	X			X	X	X			X	X	X
Secured datacenter and systems management room				X	X				X	X				X	X
Encrypted laptops				X	X										
Security key management				X	X										
Penetration hack-tests				X	X				X	X				X	X
IDS systems				X	X				X	X				X	X
Internet access limited to specific sites				X	X				X	X				X	X
Encrypted e-mail				X	X										
Printing only allowed in specific closed rooms				X	X										

Attack Vectors

Malicious code

Applications that, when activated, can cause network and server overload, steal data and passwords, or erase data

Worms

Self-replicating programs that spread from one computer to another, leaving infections as they travel

Attack Vectors

Virus

Self-replicating program fragment that attaches itself to a program or file enabling it to spread from one computer to another, leaving infections as it travels

Trojan Horse

Appears to be useful software but will actually do damage once installed or run on your computer

Attack Vectors

Denial of service attack

An attempt to overload an infrastructure to cause disruption of a service

Can lead to downtime of a system, disabling an organization to do its business

In a Distributed Denial of Service (DDoS) attack the attacker uses many computers to overload the server

Groups of computers that are infected by malicious code, called botnets, perform an attack

Preventive DDoS measures:

- Split business and public resources
- Move all public facing resources to an external cloud provider
- Setup automatic scalability (auto scaling, auto deployment) using virtualization and cloud technology
- Limit bandwidth for certain traffic
- Lower the Time to Live (TTL) of the DNS records to be able to reroute traffic to other servers when an attack occurs
- Setup monitoring for early detection

Attack Vectors

Phishing

A technique of obtaining sensitive information

The phisher sends an e-mail that appears to come from a legitimate source, like a bank or credit card company, requesting "verification" of information

The e-mail usually contains a link to a fraudulent web page

Security Patterns

Identity and Access Management (IAM)

The process of managing the identity of people and systems, and their permissions

The IAM process follows three steps:

Users or systems claim who they are: **identification**

The claimed identity is checked: **authentication**

Permissions are granted related to the identity and the groups it belongs to: **authorization**

Layered Security

Layered security (also known as a Defense-In-Depth strategy) implements various security measures in various parts of the IT infrastructure

- Instead of having one big firewall and have all your security depend on it, it is better to implement several layers of security

Preferably security layers make use of different technologies

- This makes it harder for hackers to break through all barriers, as they will need specific knowledge for each step

Disadvantage: increases the complexity of the system

Cryptography

The practice of **hiding information** using encryption and decryption techniques

Encryption is the conversion of information from a readable state to apparent random data

Only the receiver has the ability to **decrypt** this data, transforming it back to the original information

A **cipher** is a pair of algorithms that implements the encryption and decryption process. The operation of a cipher is controlled by a **key**

Block ciphers

- Input:
  - A block of plaintext
  - A key
- Output:
  - A block of cipher text
- Used across a wide range of applications, from ATM machine data encryption to e-mail privacy and secure remote access
- Standards:
  - Data Encryption Standard (DES)
  - Advanced Encryption Standard (AES)

Cryptography

Stream ciphers

- Create an arbitrarily long stream of key material
- Combines key stream with the plaintext bit-by-bit or character-by-character
- Used when data is in transit over the network
- RC4 is a widely-used stream cipher

Cryptographic Attacks

Every encryption method can be broken using a brute force attack

Except a one-time pad cipher with the key of equal or greater length than the message

A brute force attack consists of systematically checking all possible keys until the correct key is found

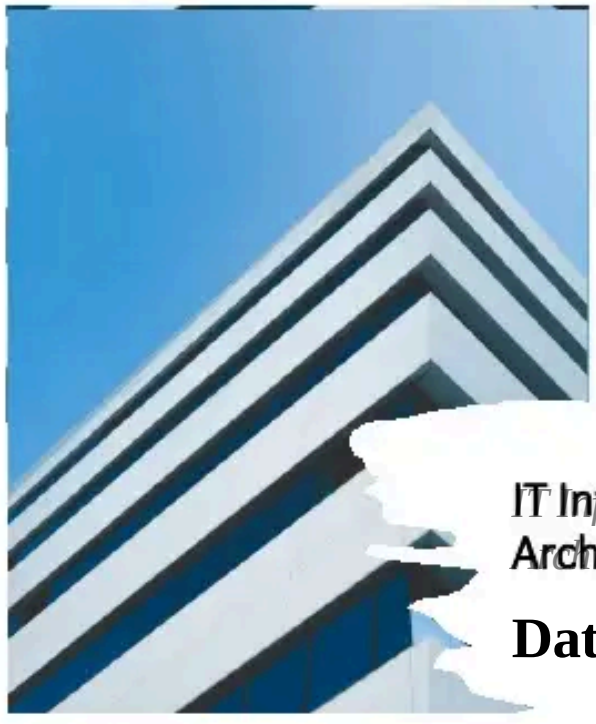
The amount of effort needed is exponentially dependent on the size of the key

Cryptographic Attacks

Effective security could be achieved if it is proven that no efficient method (as opposed to the time consuming brute force method) can be found to break the cipher

Most successful attacks are based on flaws in the implementation of an encryption cipher

To ensure a cipher is flawless, the source code is usually open source and thus open to inspection to everyone

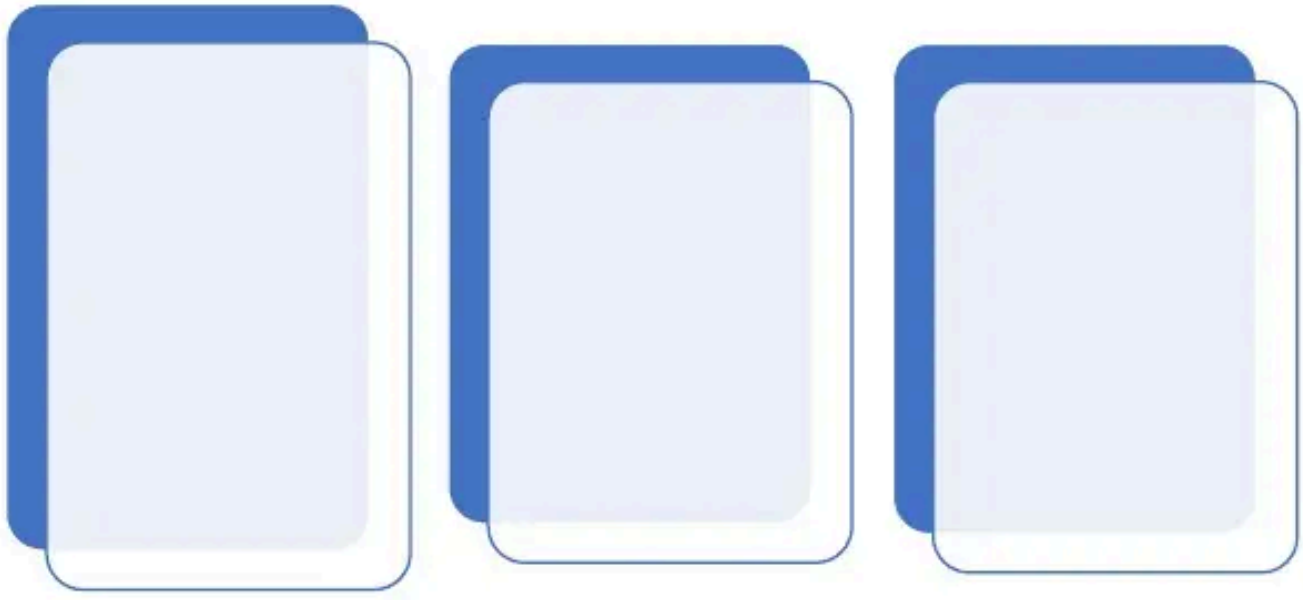


IT Infrastructure Architecture

Data Centers

Topic outline :

Data Center :



Today's Data Center :

Large datacenters contain shipping containers packed with thousands of servers each .

Data Center Categories :

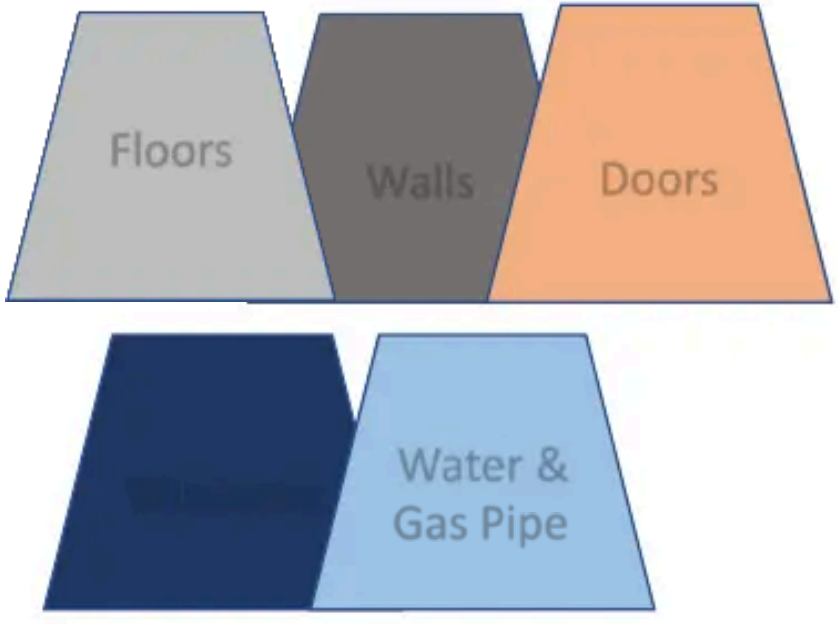
Sub Equipment Room (SER)

Main Equipment Room (MER)

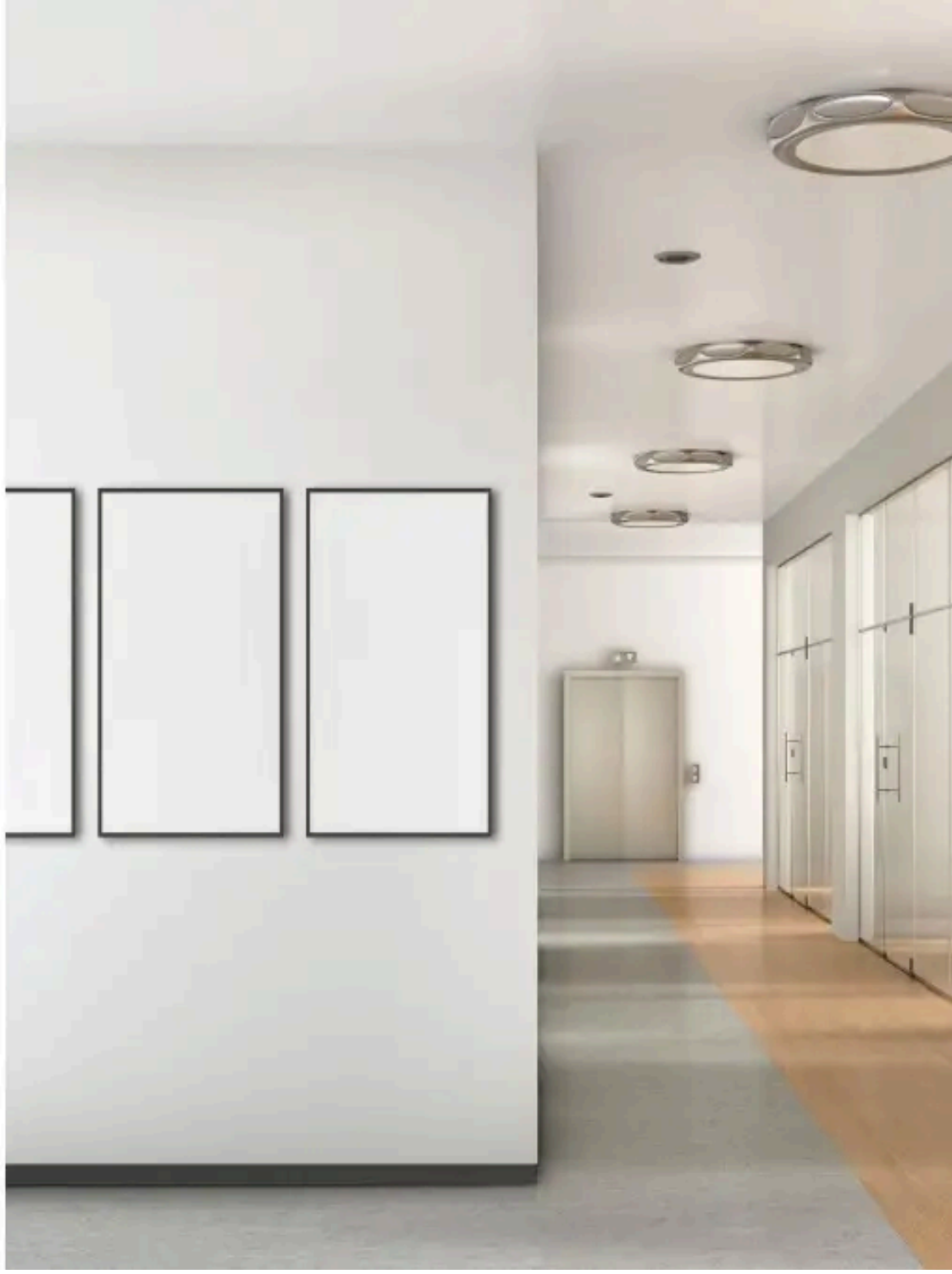
Organization owned datacenter

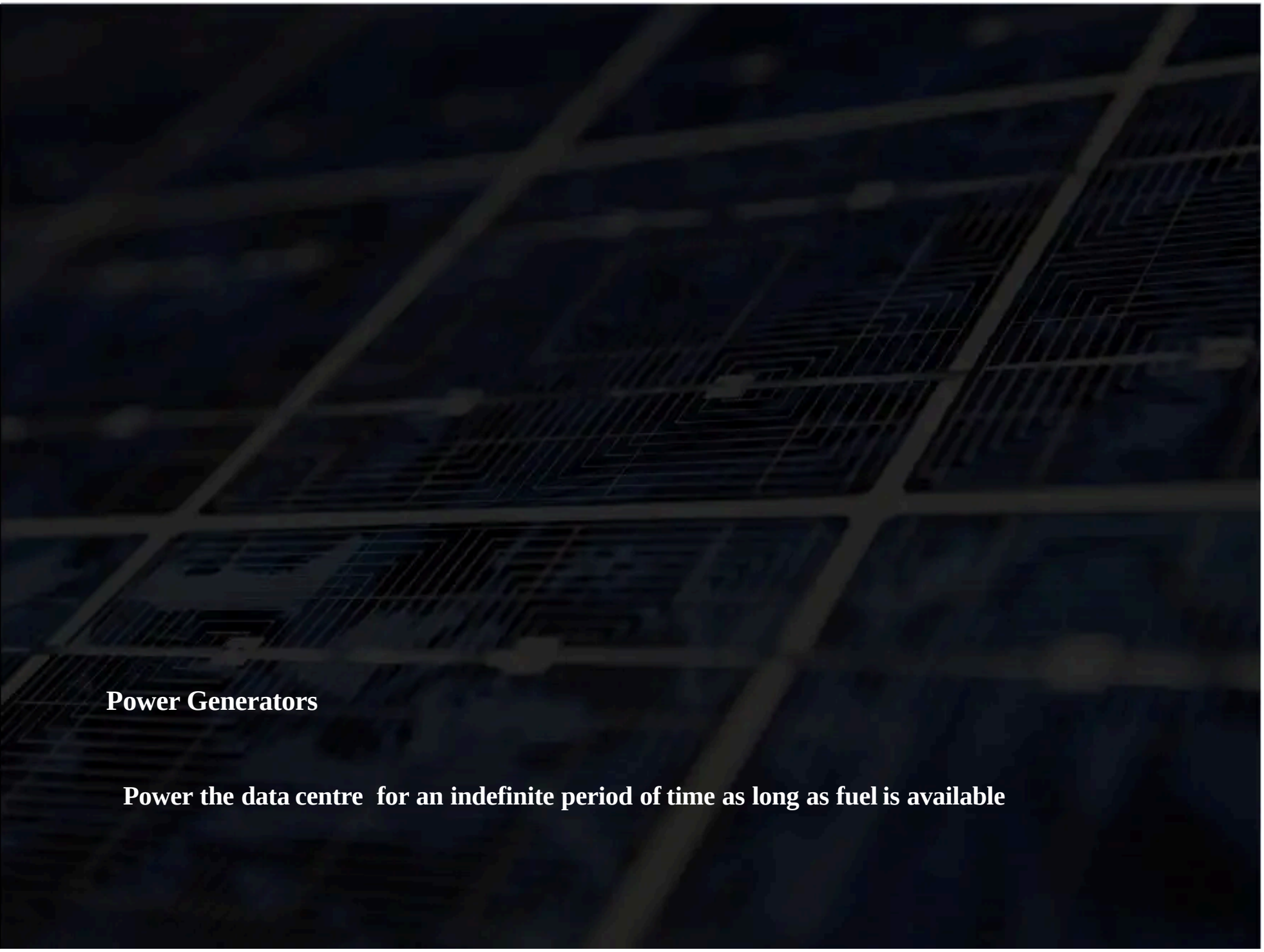
Multi-tenant datacenter


Environment
Visibility
Utilities
Located in foreign countries


Floors
Walls
Doors
Water &
Gas Pipe

Disadvantages



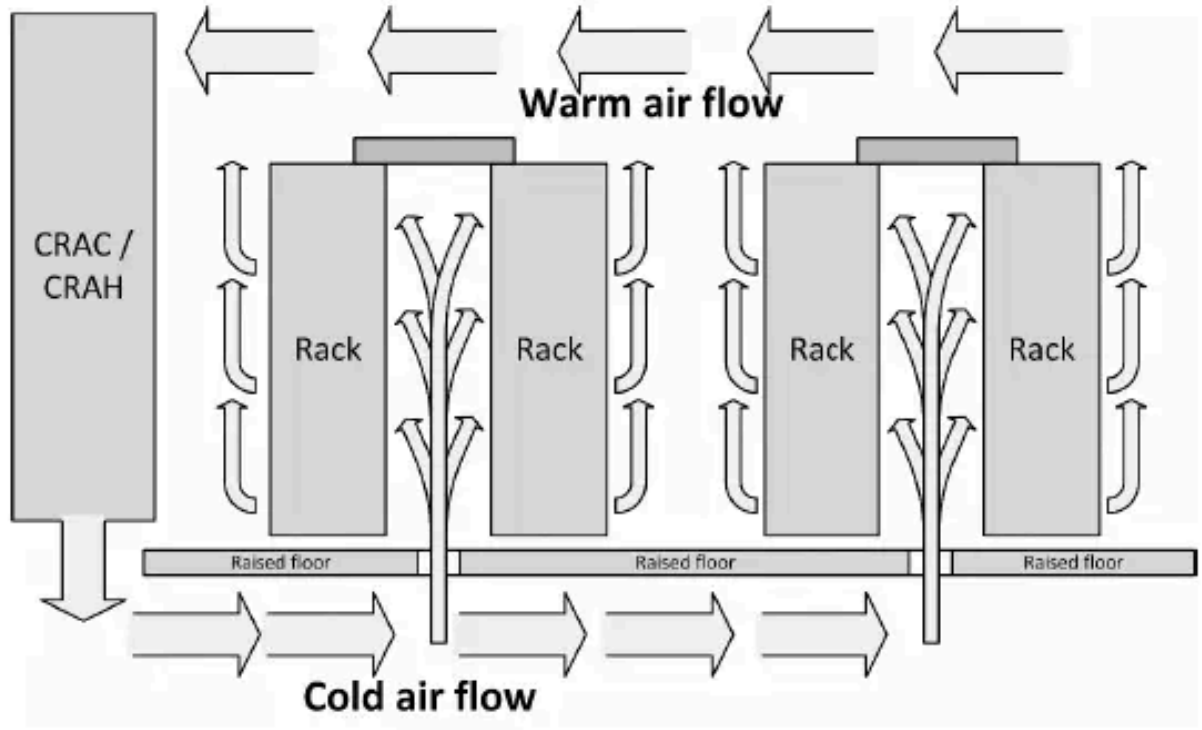


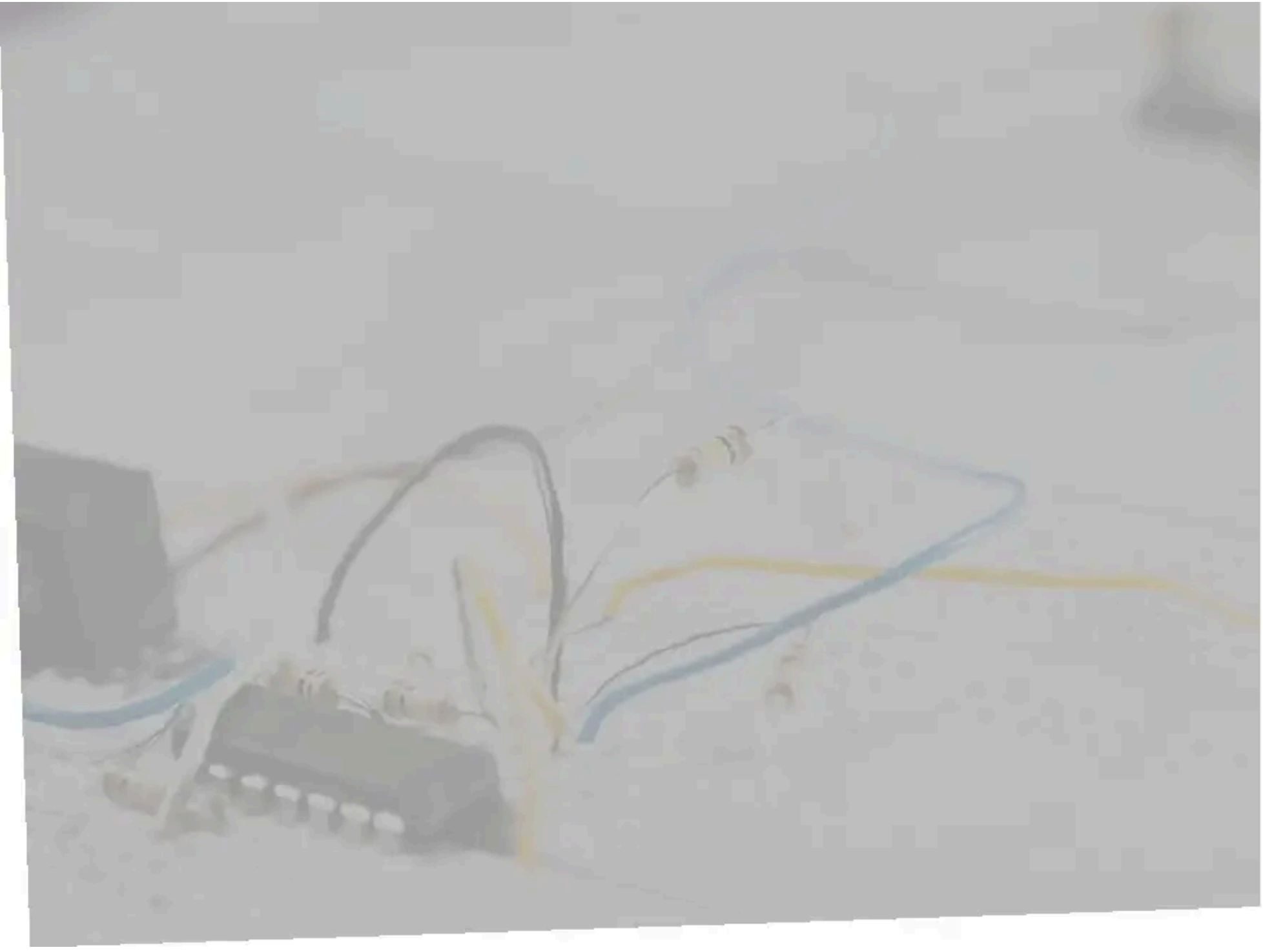
Power Generators

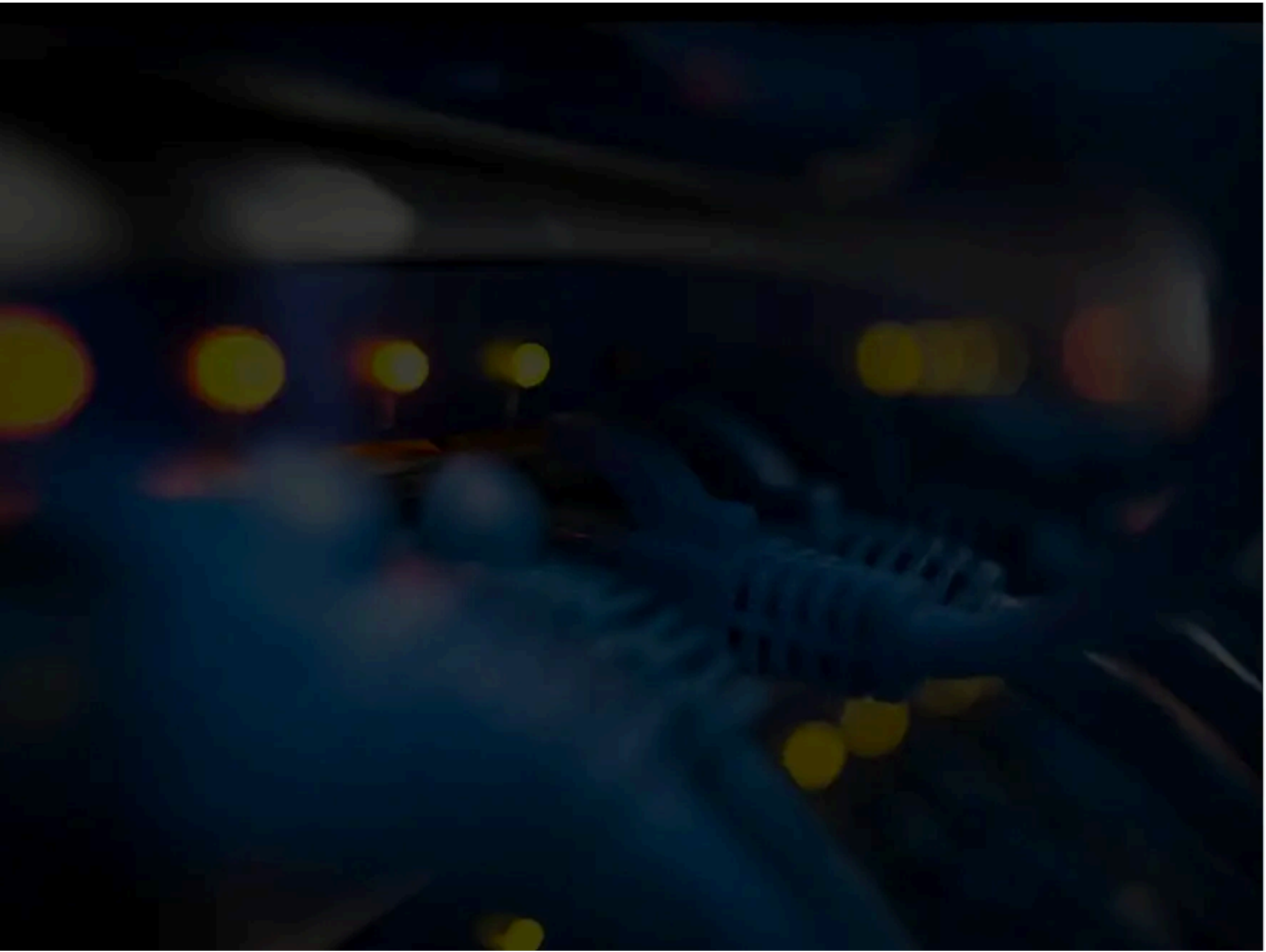
Power the data centre for an indefinite period of time as long as fuel is available

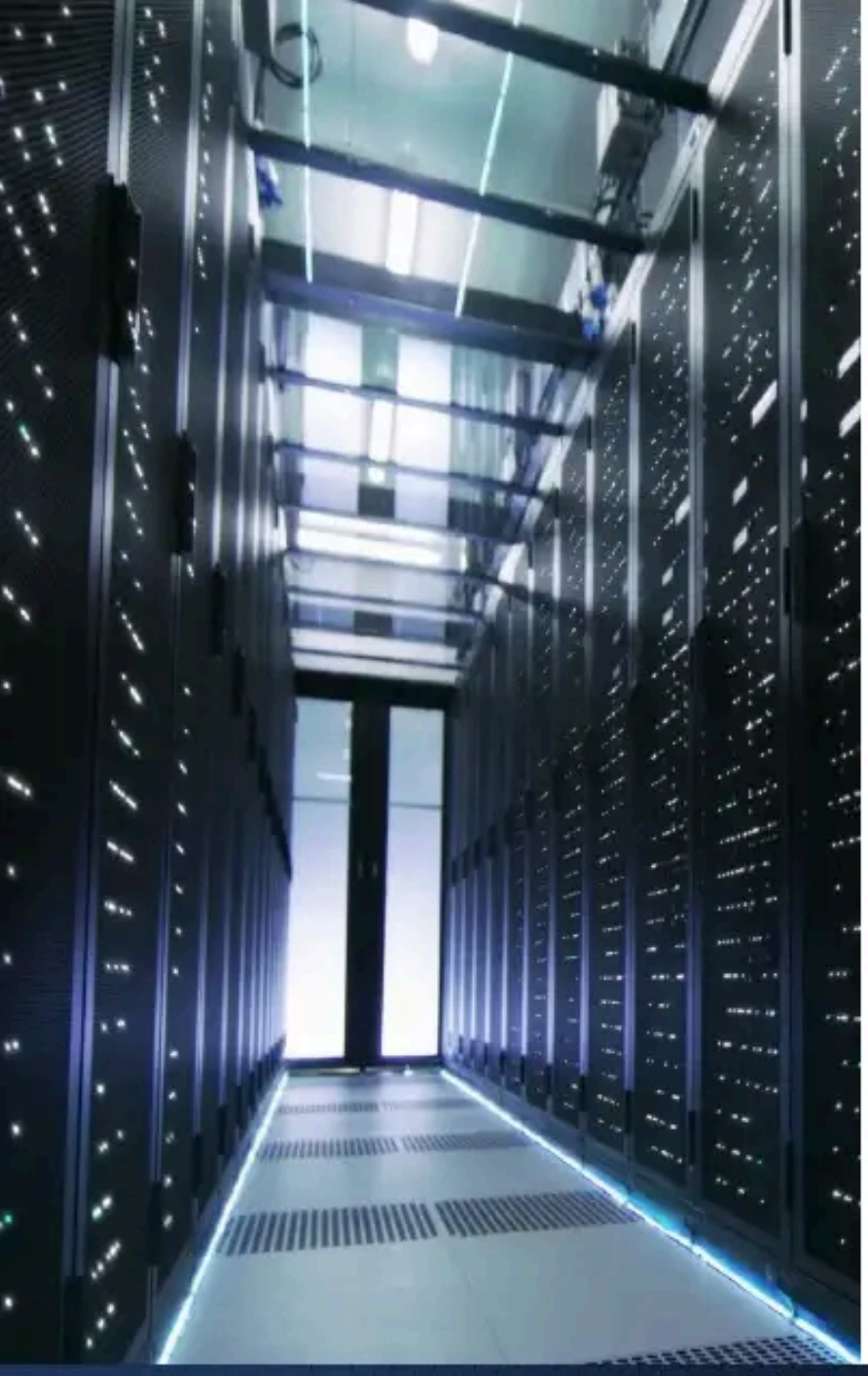
Computer Room Air Conditioners (CRAC)

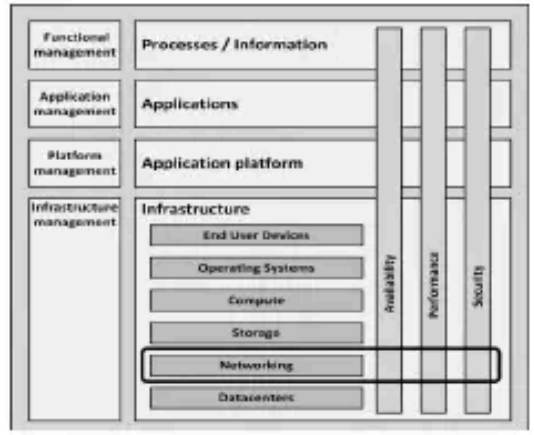
Computer Room Air Handlers (CRAH)


Warm air flow
CRAC /
CRAH
Rack
Rack
Rack
Rack
Raised floor
Raised floor
Raised floor
Cold air flow

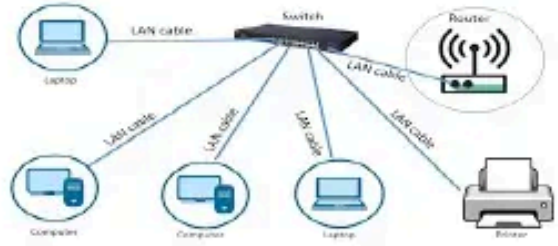




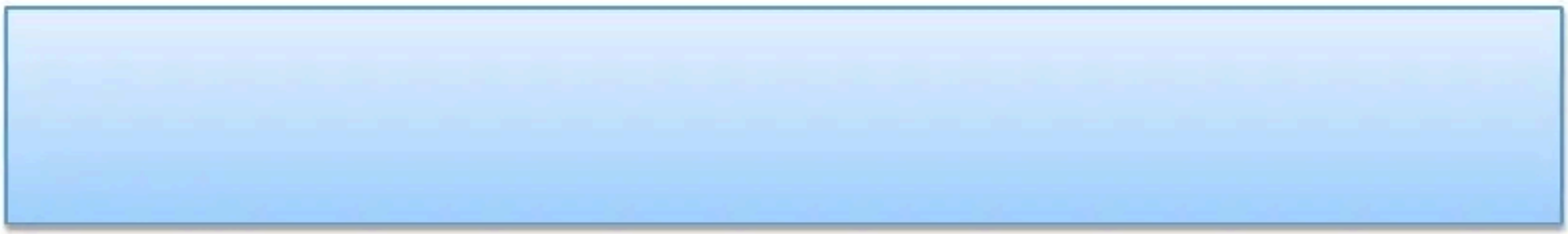




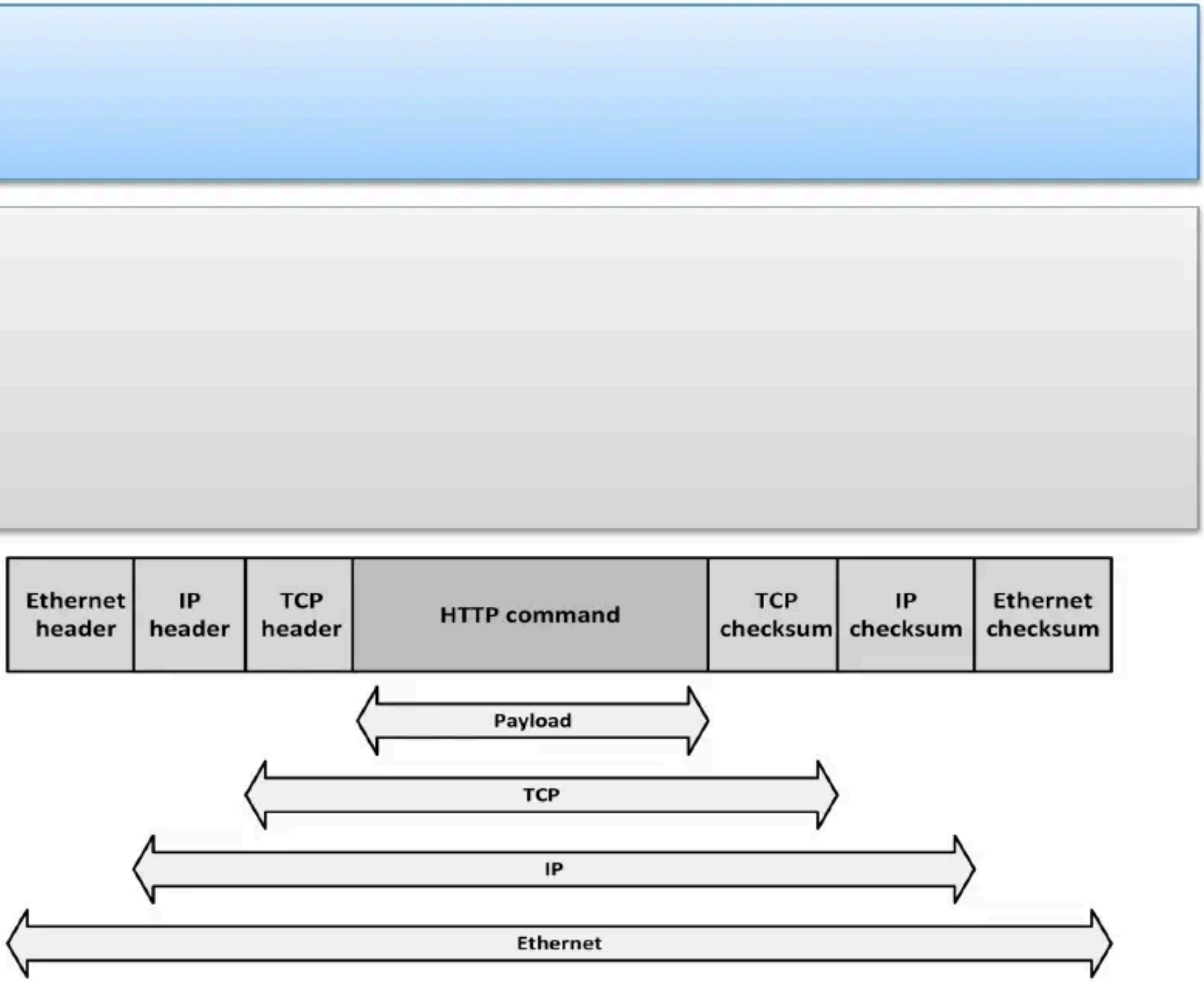
Mainframe Computers

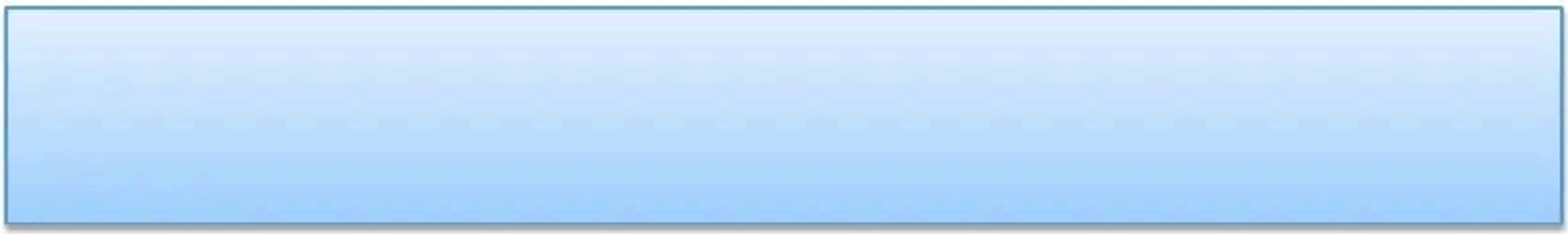


Local Area Network



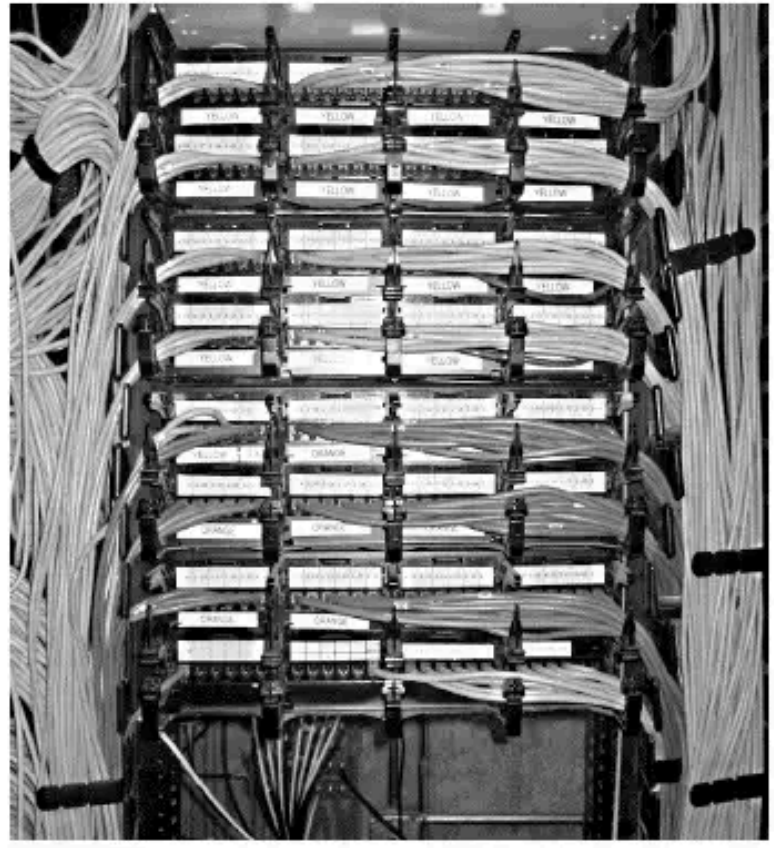

7 Layers of the OSI Model
Application
• End User layer
• HTTP, FTP, IRC, SSH, DNS
Presentation
• Syntax layer
• SSL, SSH, IMAP, FTP, MPEG, JPEG
Session
• Synch & send to port
• API's, Sockets, WinSock
Transport
• End-to-end connections
• TCP, UDP
Network
• Packets
• IP, ICMP, IPSec, IGMP
Data Link
• Frames
• Ethernet, PPP, Switch, Bridge
Physical
• Physical structure
• Coax, Fiber, Wireless, Hubs, Repeaters

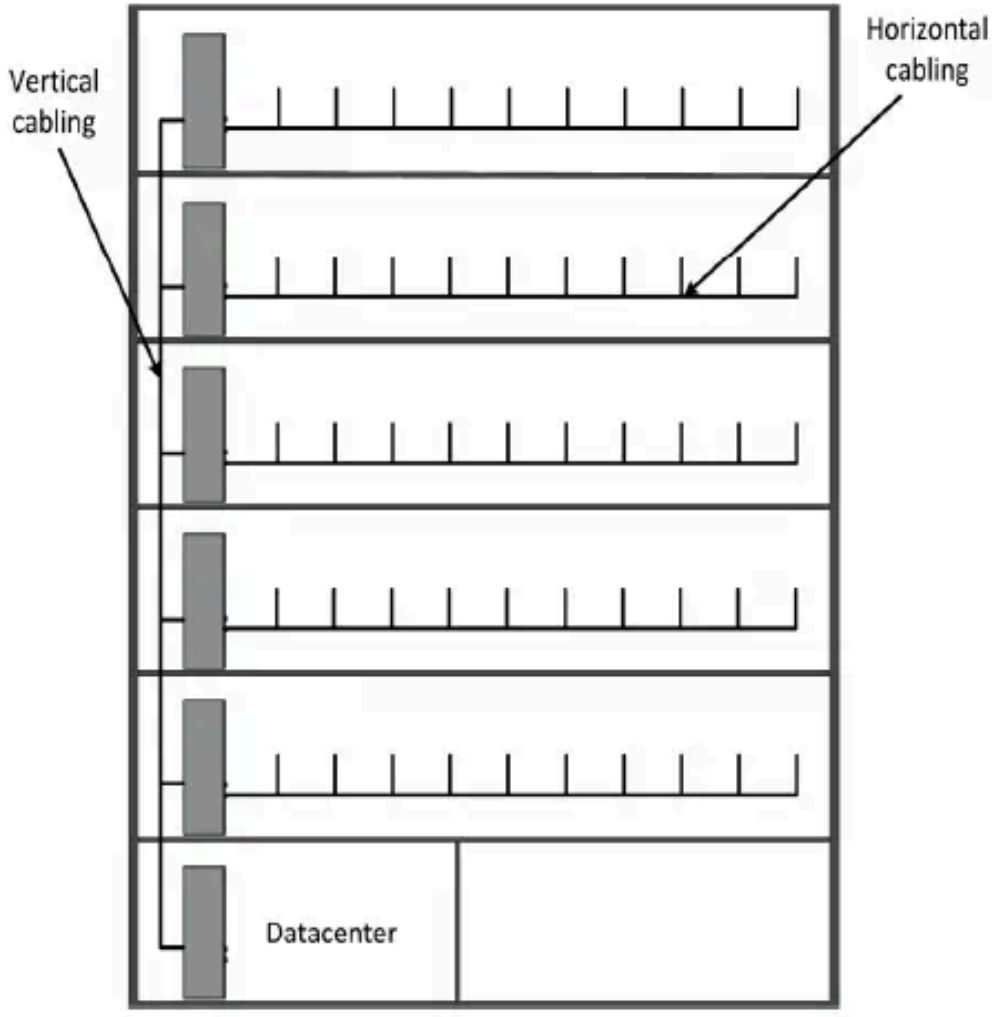

Ethernet header
IP header
TCP header
HTTP command
TCP checksum
IP checksum
Ethernet checksum
Payload
TCP
IP
Ethernet




MULTIMODE

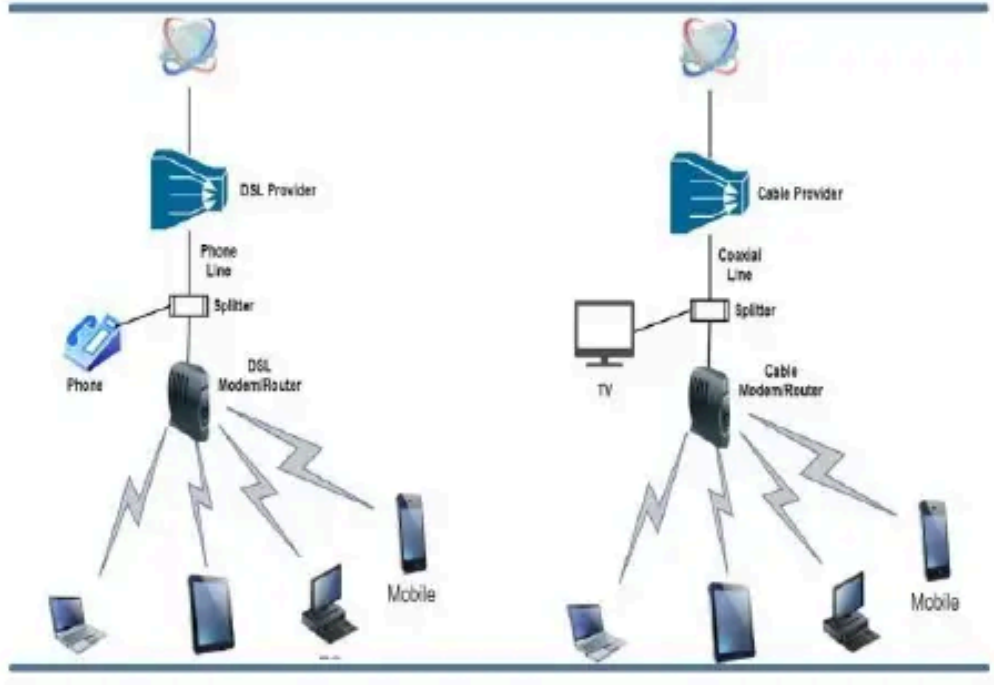

SINGLE MODE

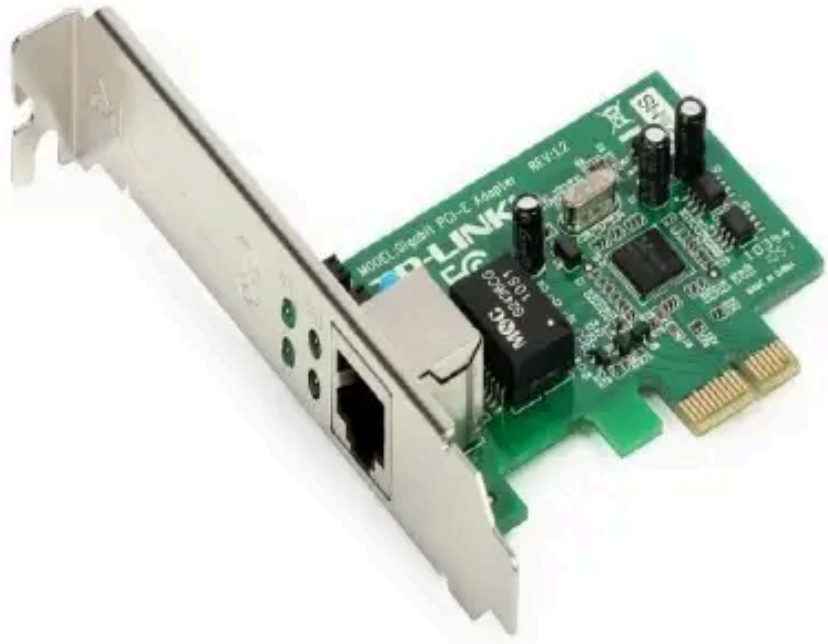

YELLOW
ORANGE

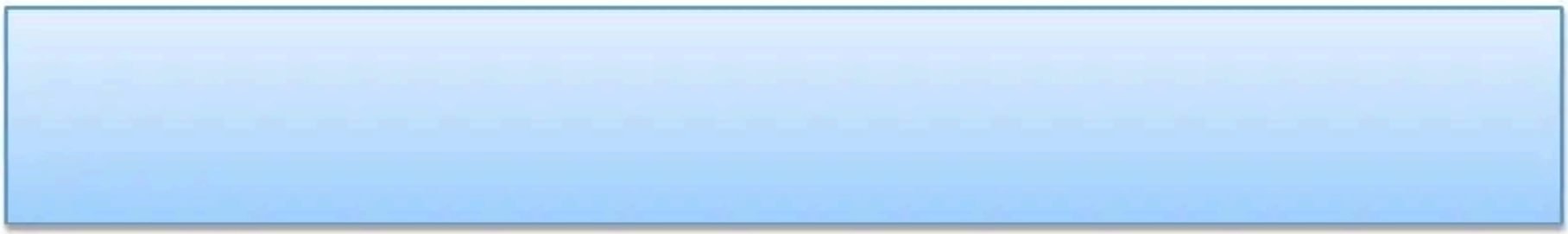

Vertical cabling
Horizontal cabling
Datacenter


Leased Line
Router
Corp NW
Local Exchange
Internet

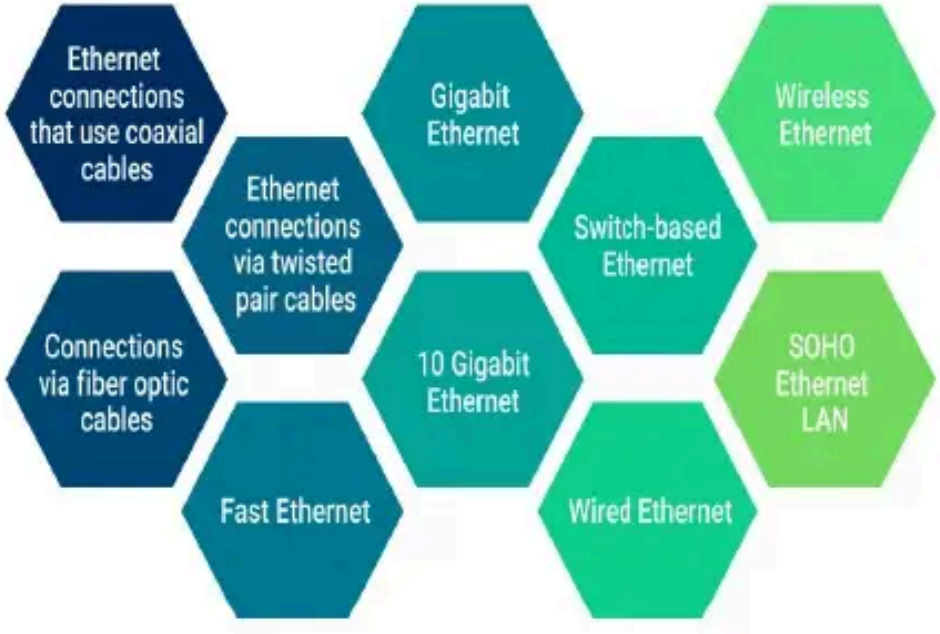
Leased Line Internet Connectivity



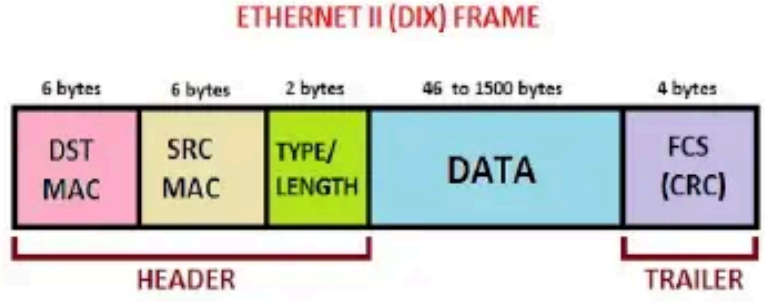

D-LINK



Types of Ethernet



spiceworks



Ethernet: CSMA/CD (continued)

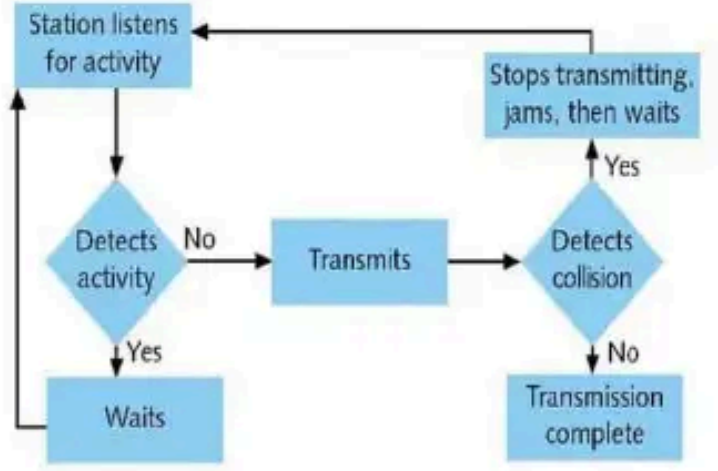
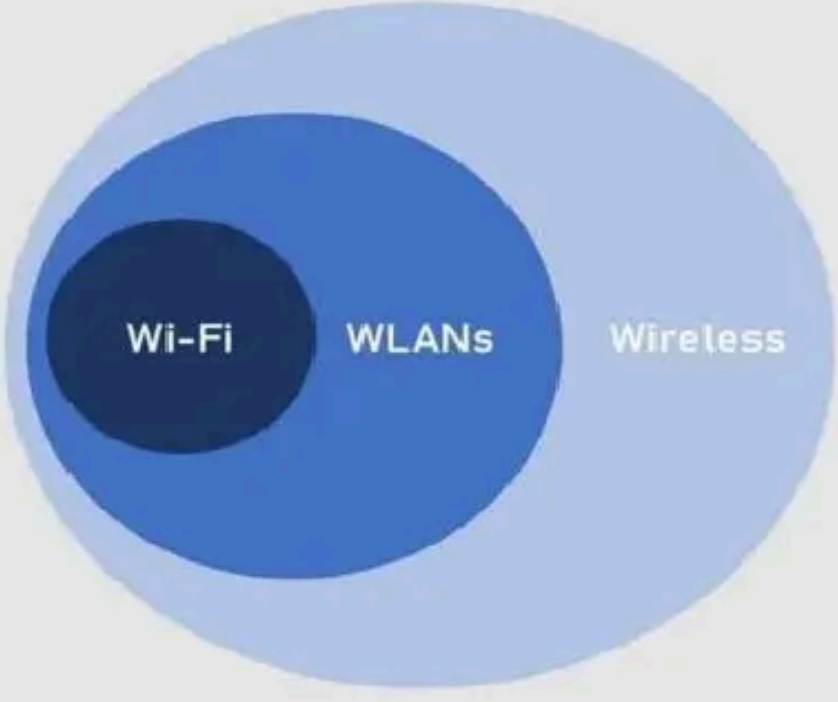


Figure 6-11: CSMA/CD process

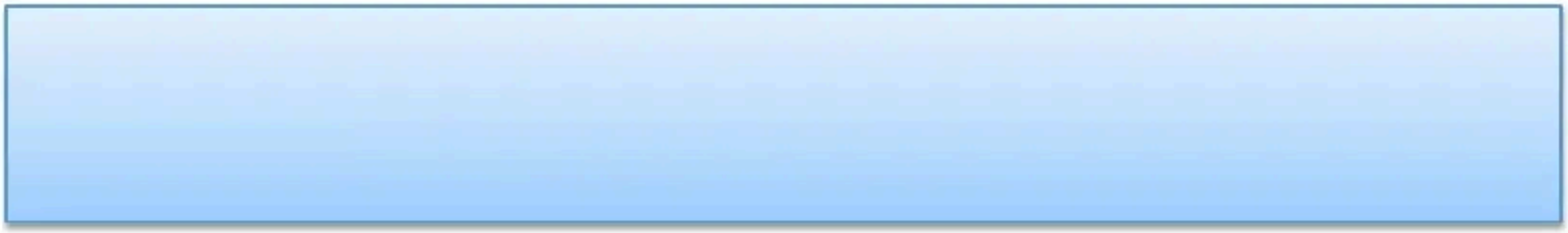


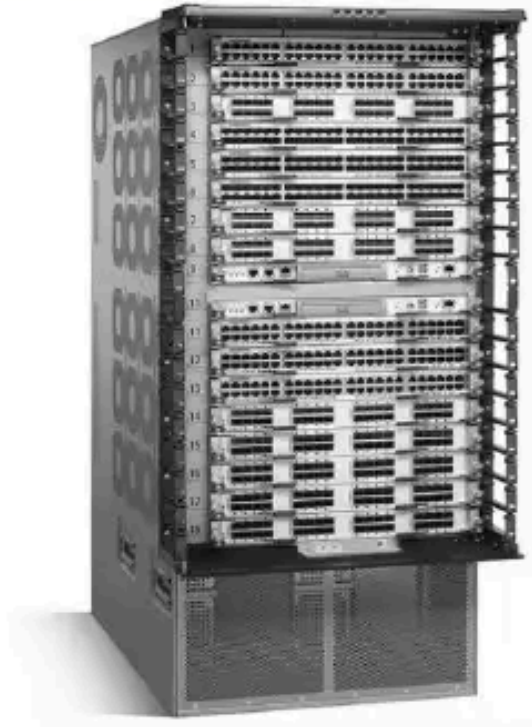
Wi-Fi vs. WLANs vs. Wireless

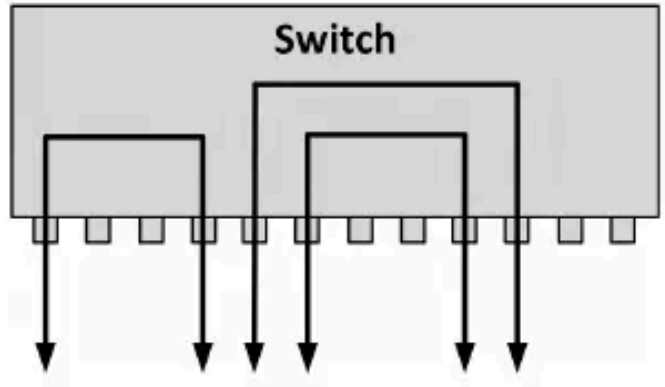
Wi-Fi, is a certain type of WLAN that uses specifications of the 802.11 wireless standard.

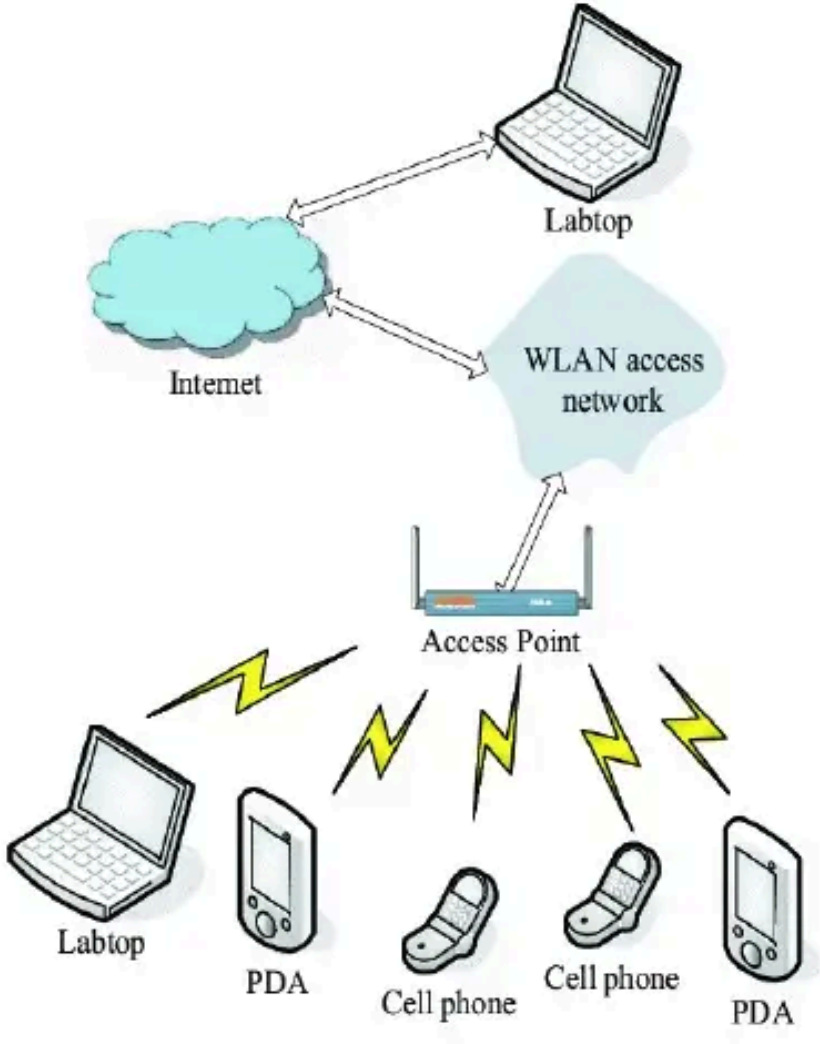
The terms WLAN and Wi-Fi are often linked and used interchangeably. A WLAN can be built on various wireless technologies.

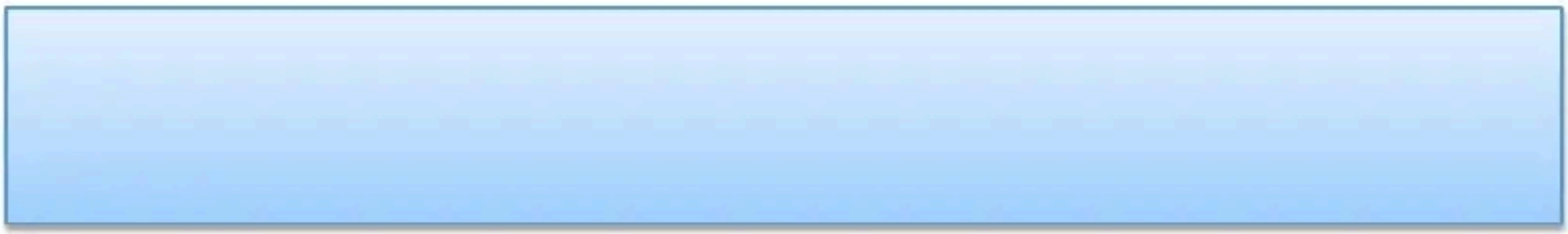
Wi-Fi networks are WLANs. But Wi-Fi is not the only type of WLAN.

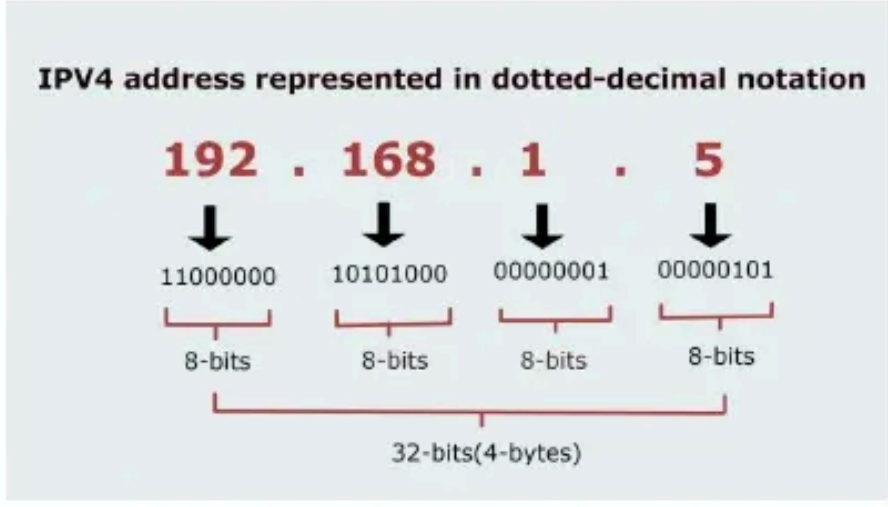





Switch

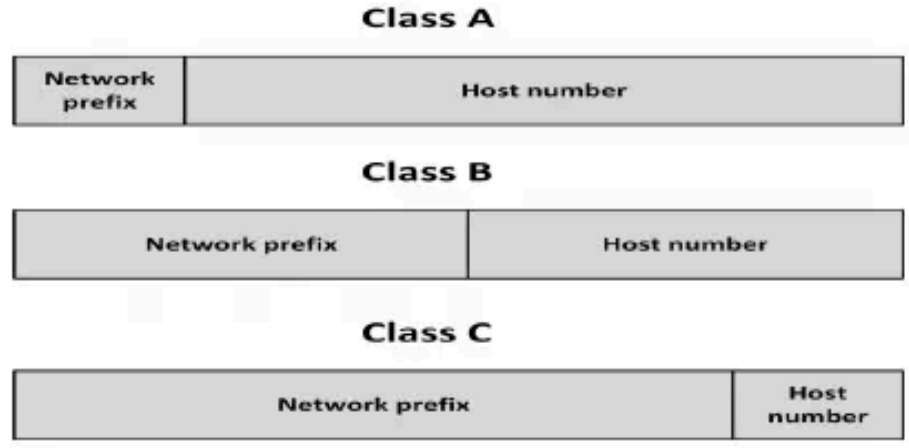

Labtop
Internet
WLAN access network
Access Point
Labtop
PDA
Cell phone
Cell phone
PDA



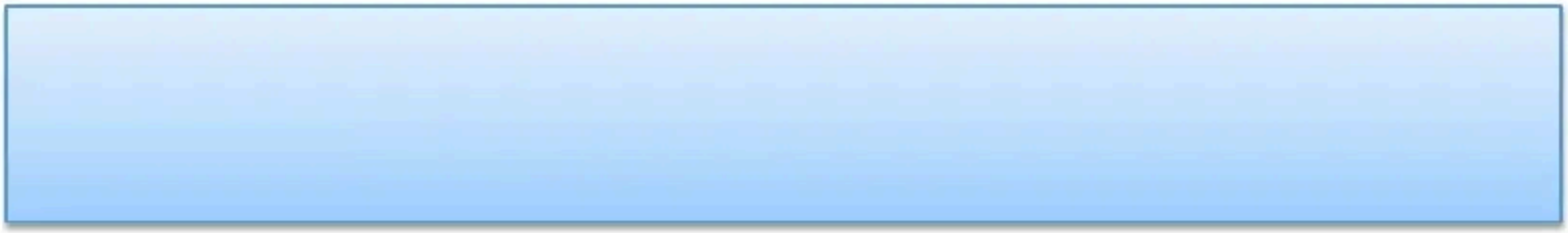


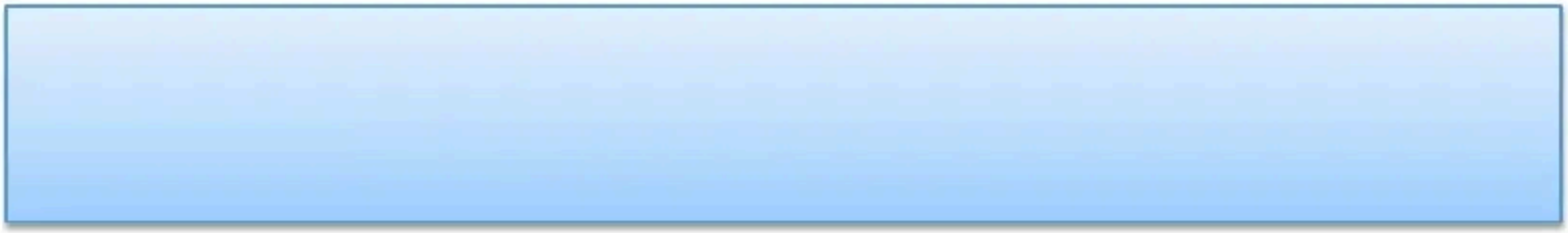
10 . 121 . 12 . 16

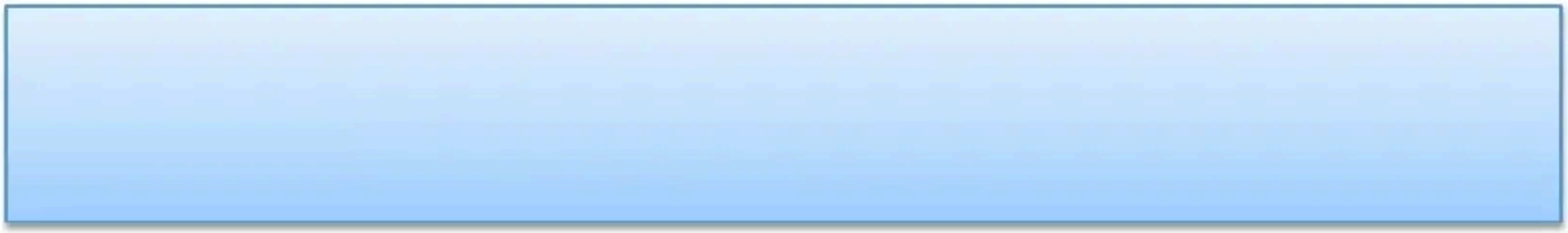
Network prefix	Host number

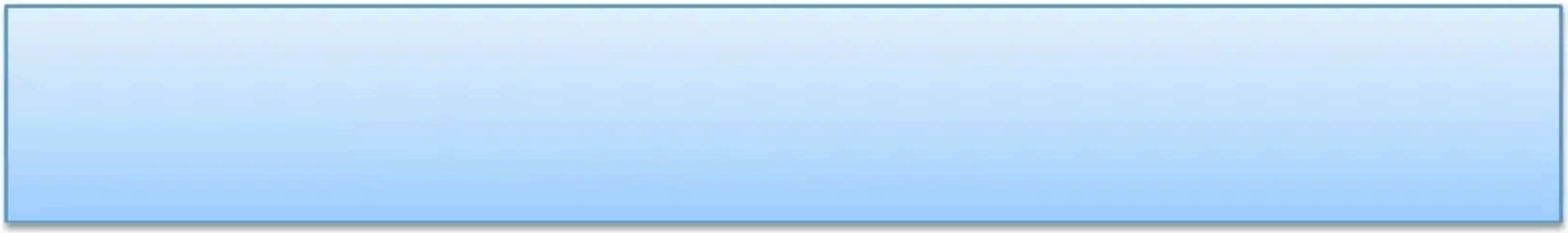

Class A
Network prefix
Host number
Class B
Network prefix
Host number
Class C
Network prefix
Host number

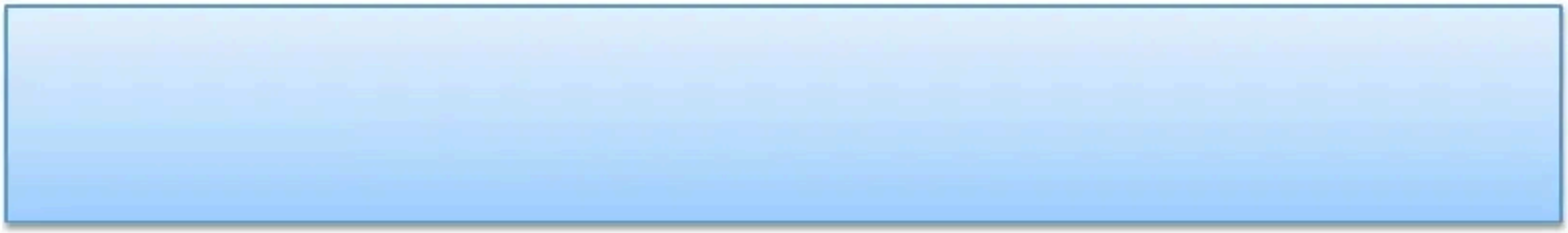
Address	**196**	.	**121**	.	**12**	.	**241**
	Network prefix				**Sub net**		**Host**
Subnet mask	**255**	.	**255**	.	**255**	.	**240**
Binary	**11111111**	.	**11111111**	.	**11111111**	.	**11110000**

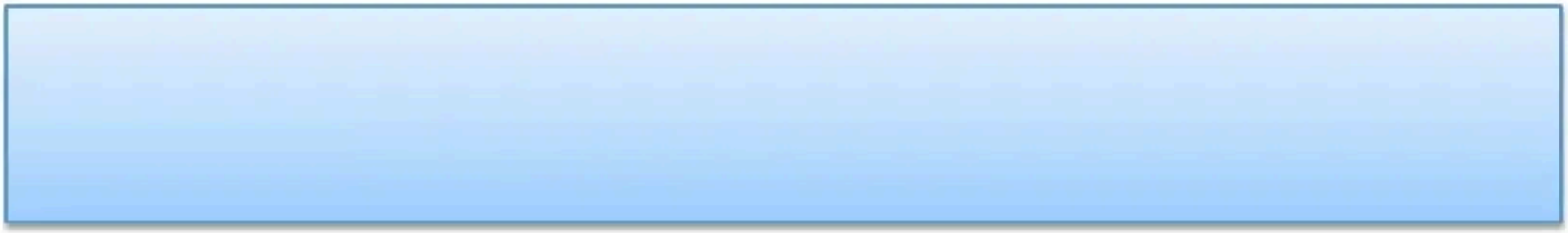


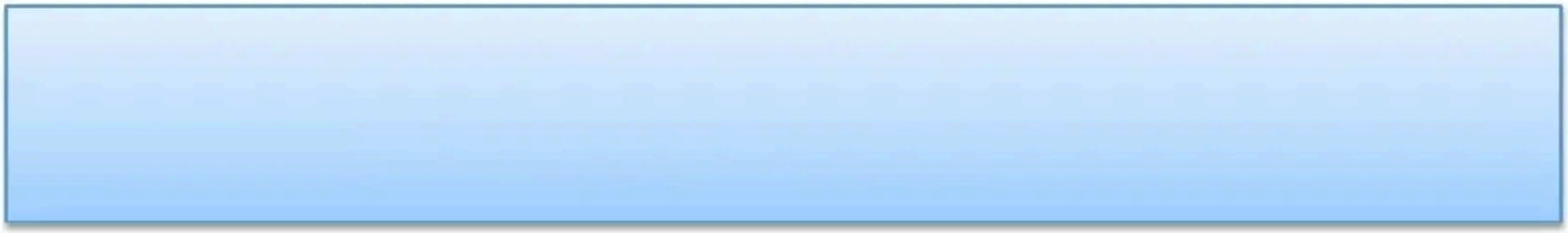


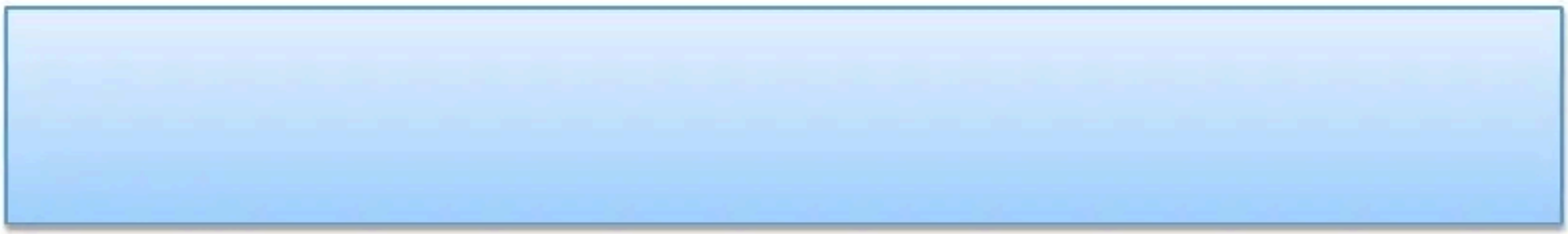


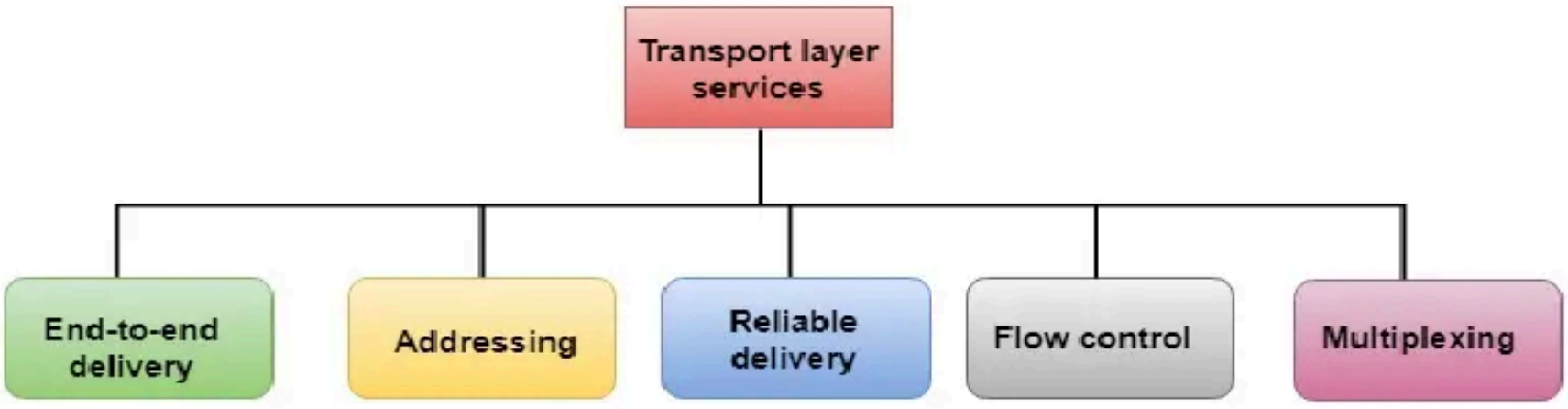


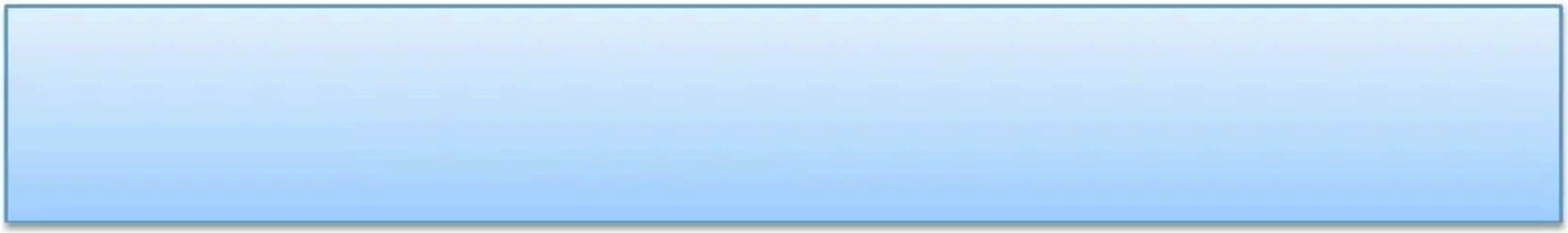


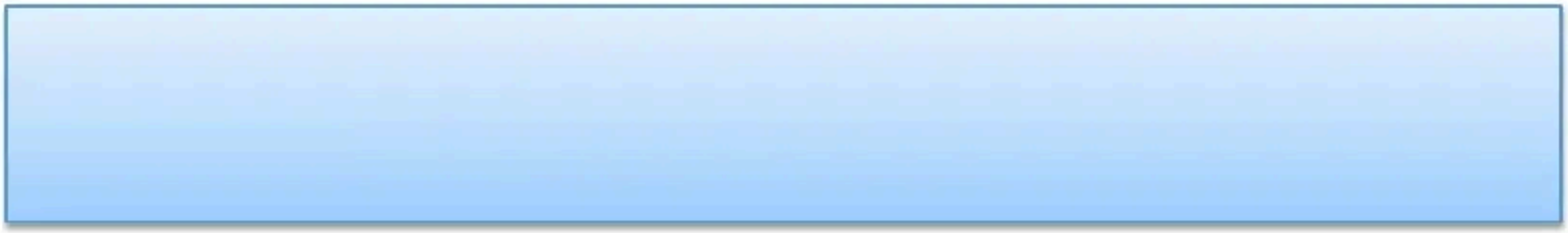


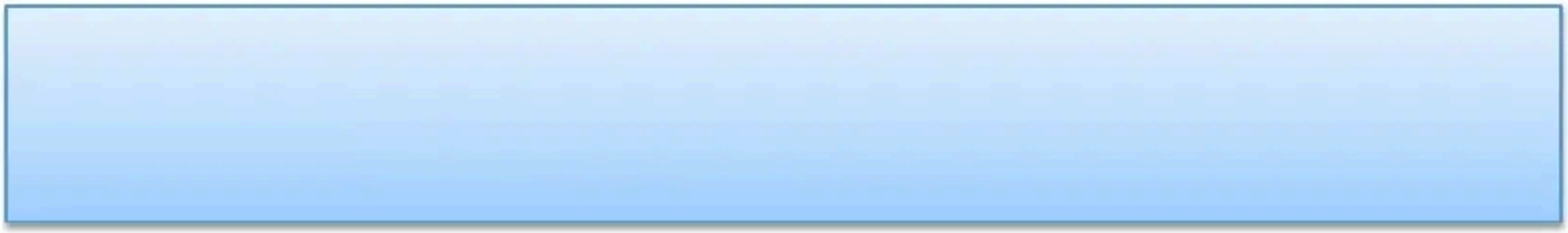


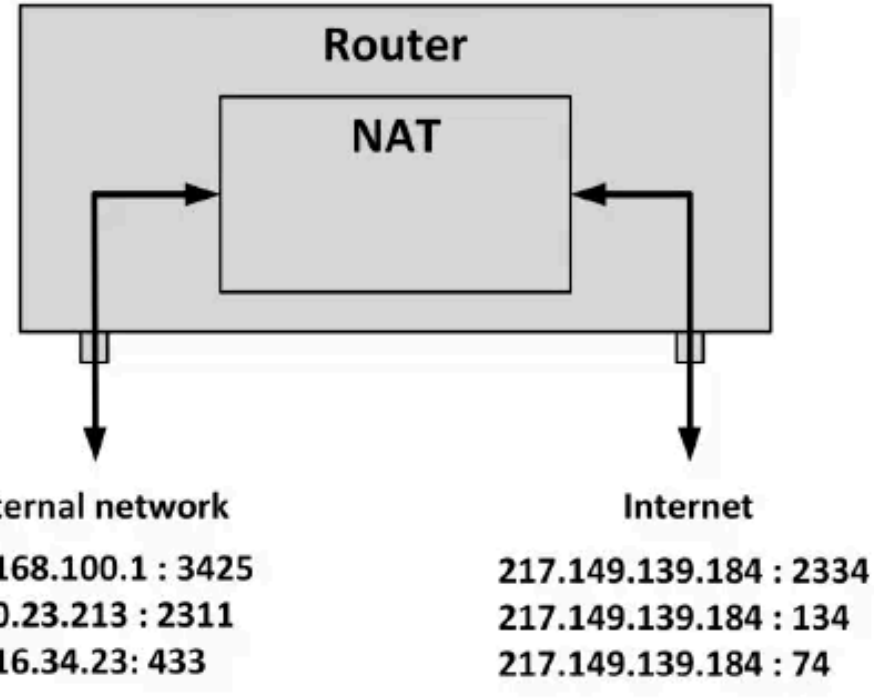


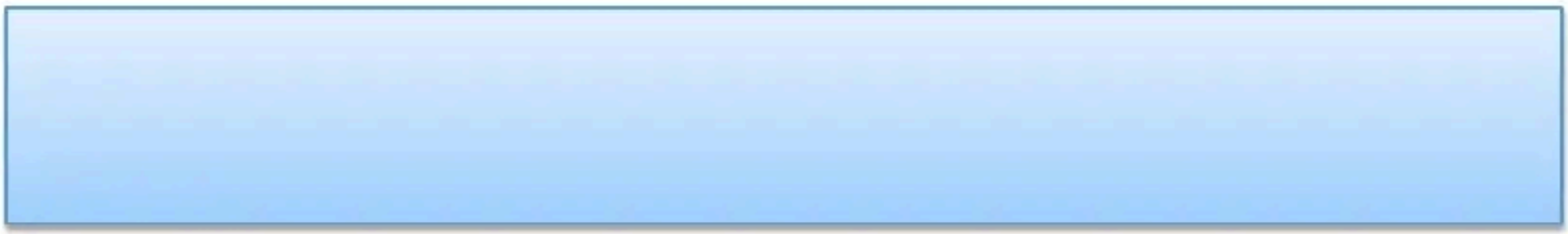

Transport layer services
End-to-end delivery
Addressing
Reliable delivery
Flow control
Multiplexing

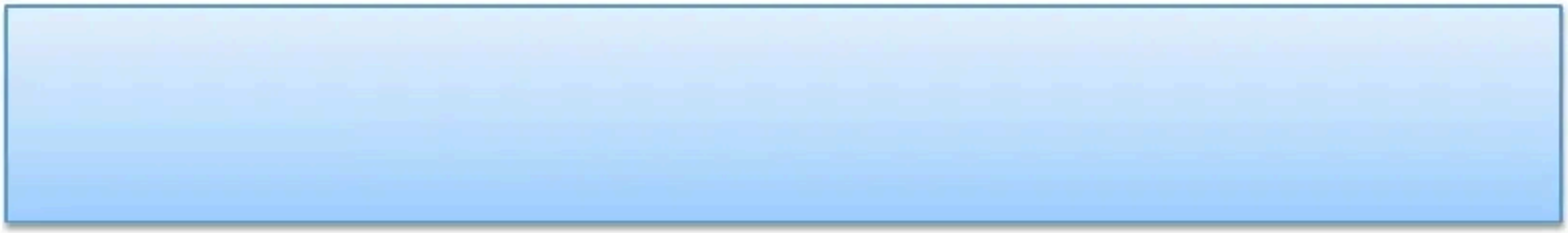


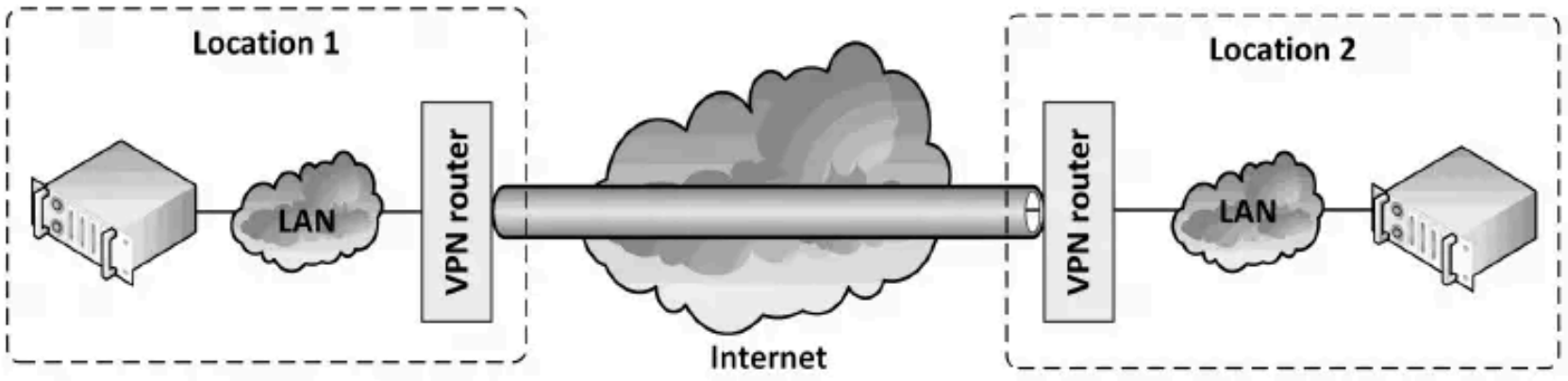


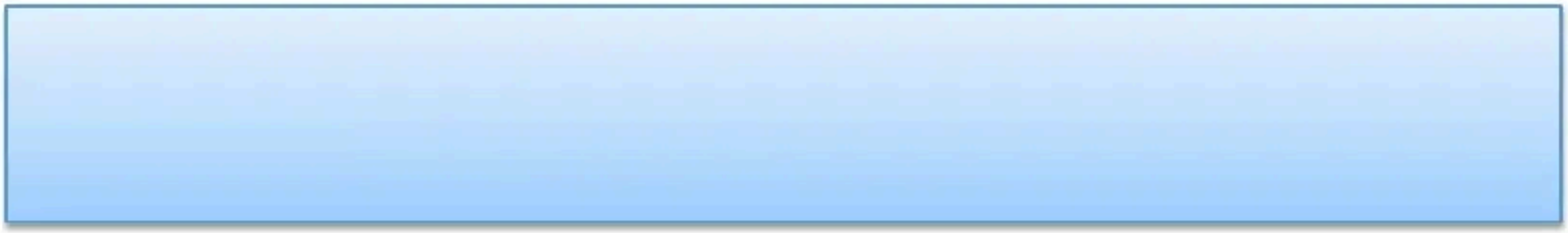


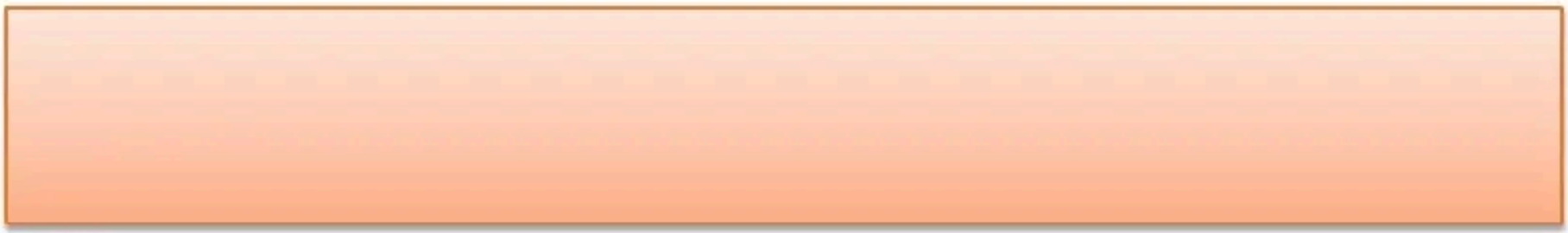

Router
NAT
Internal network
192.168.100.1 : 3425
10.10.23.213 : 2311
172.16.34.23: 433
Internet
217.149.139.184 : 2334
217.149.139.184 : 134
217.149.139.184 : 74

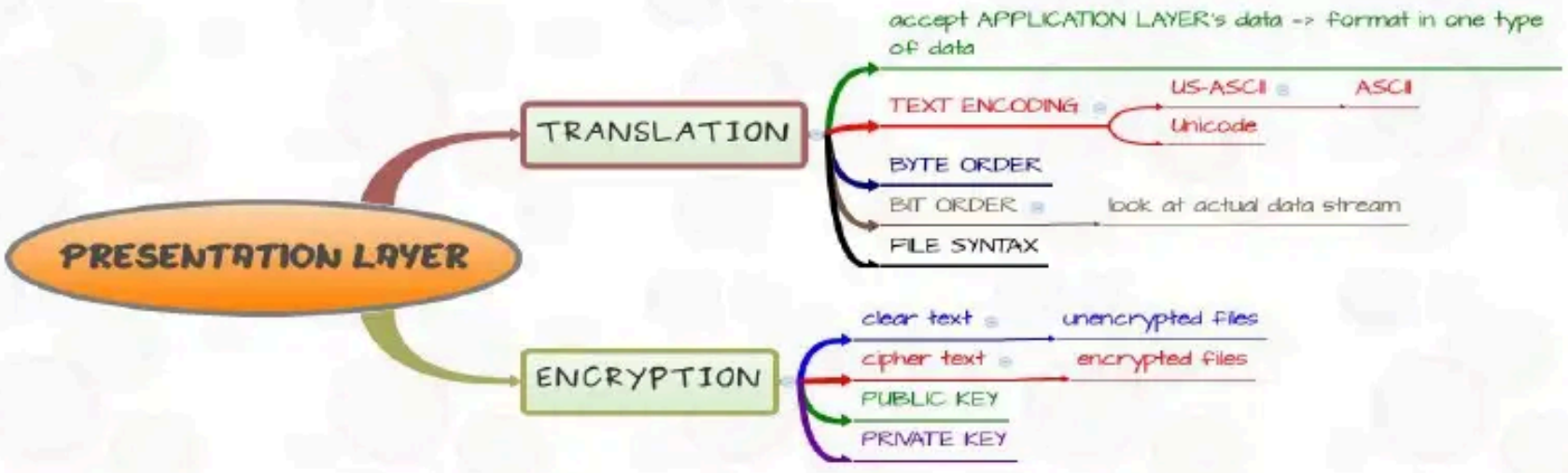


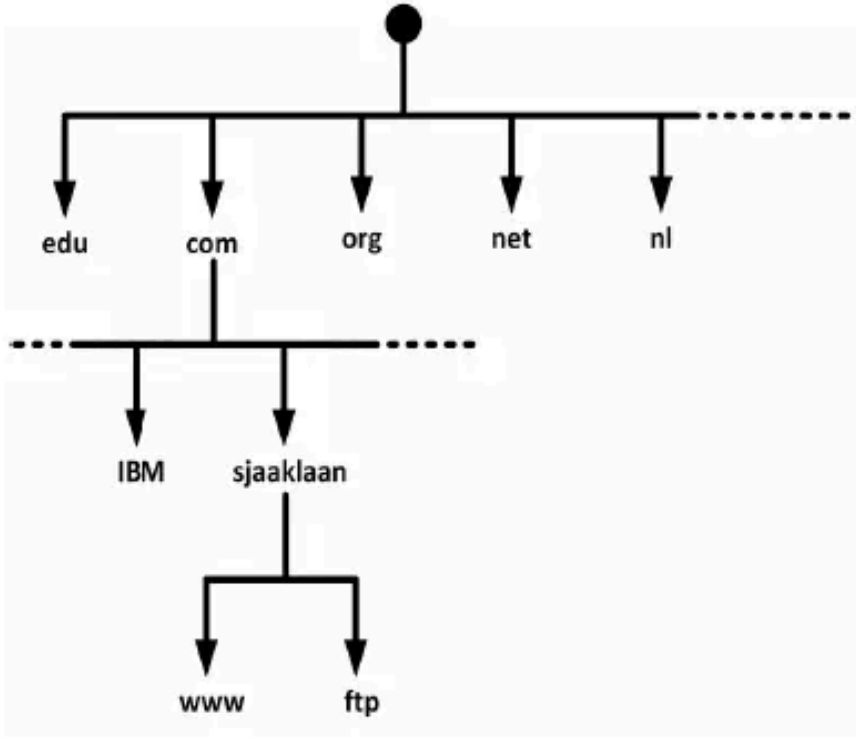


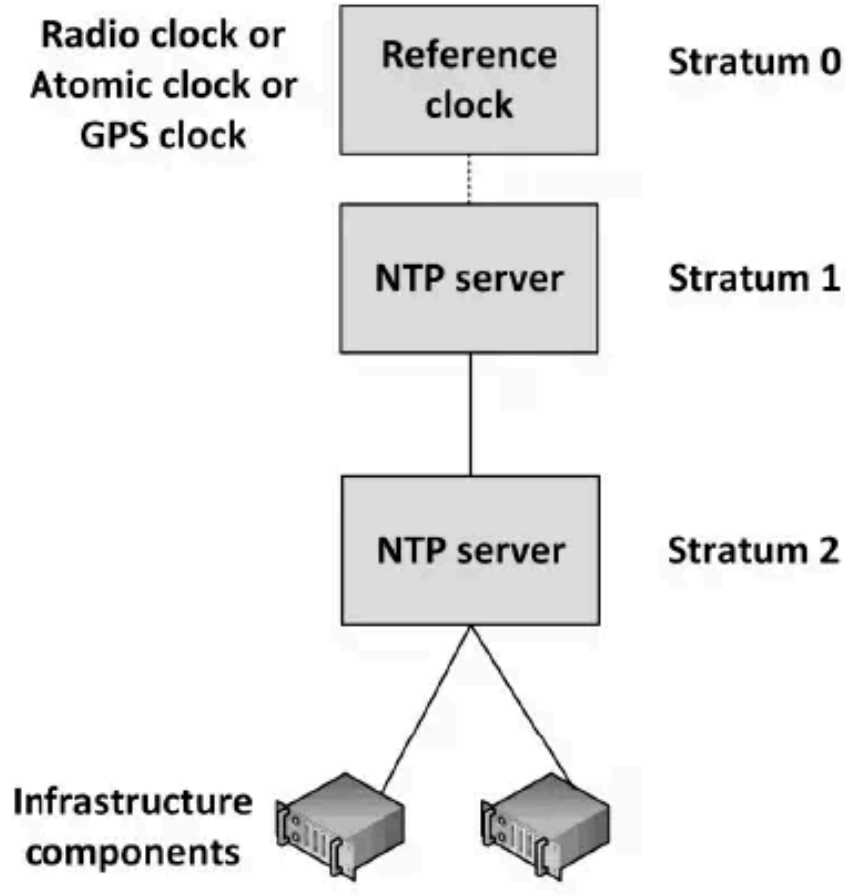

Location 1
LAN
VPN router
Internet
Location 2
VPN router
LAN






PRESENTATION LAYER
TRANSLATION
accept APPLICATION LAYER's data -> format in one type of data
TEXT ENCODING
US-ASCII
ASCII
Unicode
BYTE ORDER
BIT ORDER
look at actual data stream
FILE SYNTAX
ENCRYPTION
clear text
unencrypted files
cipher text
encrypted files
PUBLIC KEY
PRIVATE KEY


edu
com
org
net
nl
IBM
sjaaklaan
www
ftp


Radio clock or
Atomic clock or
GPS clock
Reference clock
Stratum 0
NTP server
Stratum 1
NTP server
Stratum 2
Infrastructure components