INTRODUCTION

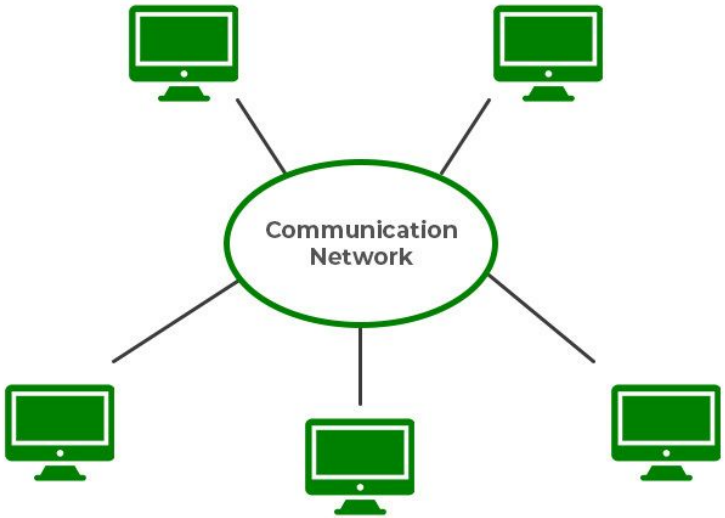
CHAPTER 1

BY ANKU JAISWAL, ASST.PROF ,IOE, PULCHOWK

CONTENT

1.1. INTRODUCTION TO DISTRIBUTED SYSTEM

A distributed system is a collection of **independent computers** at networked locations such that they communicate and interact only through message passing that is **viewed as a single system** by its users.



- All the components of the distributed system interact so as to obtain a common goal.

- A computer program running in distributed system is called distributed program.

APPLICATION OF DISTRIBUTED SYSTEM

Domain	Use of Distributed System
Healthcare	Storing and accessing medicine and patient information
E-Commerce	For storing and accessing payment information
Tracking Systems	Global Positioning System (GPS) for tracking one's location.
Gaming	In Multiplayer Games, allowing different players to play a single game.

Real World Example of Distributed System-**GOOGLE**

Google Search Engine (Distributed Information Retrieval System)

What happens: When you type a query into Google, it looks simple, but behind the scenes your request is sent to **thousands of servers** worldwide.

Distributed aspect:

1. **Data distribution**: The web index is too large for one machine, so it is split (sharded) across many servers. Each server stores part of the data.

2. **Parallel processing**: Multiple servers process your query simultaneously on different parts of the index.

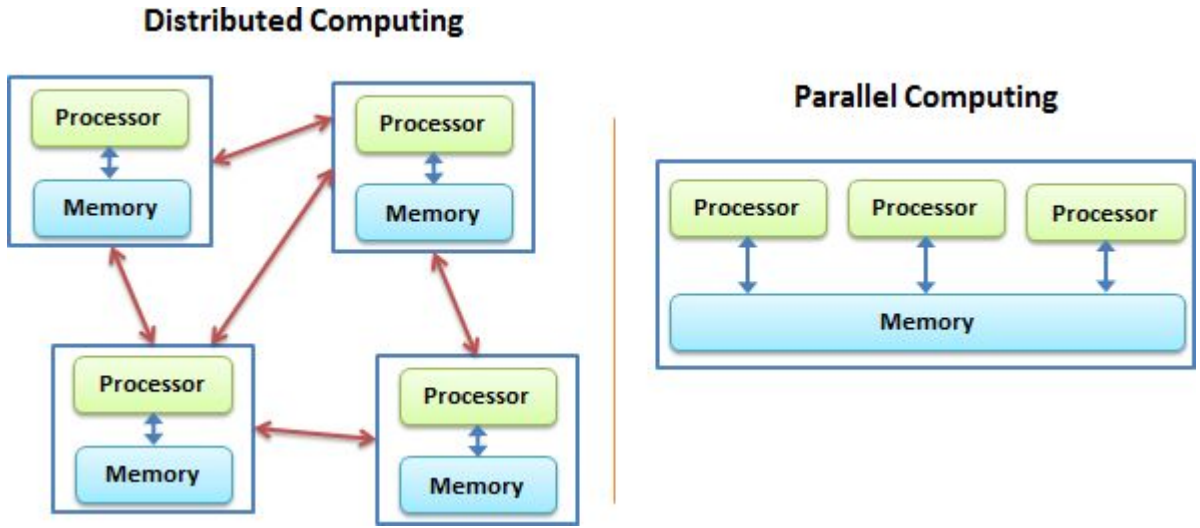
3. **Load balancing**: A load balancer decides which server cluster should handle your request based on availability and location.

4. **Fault tolerance**: If one server fails, another takes over—so you never notice a breakdown.

User experience: You see results in less than a second, even though billions of web pages were checked.

PARALLEL SYSTEM VS DISTRIBUTED SYSTEM

Each computer in a distributed system has its own memory i.e. distributed memory; but in parallel system, all processors have access to shared memory for information exchange.



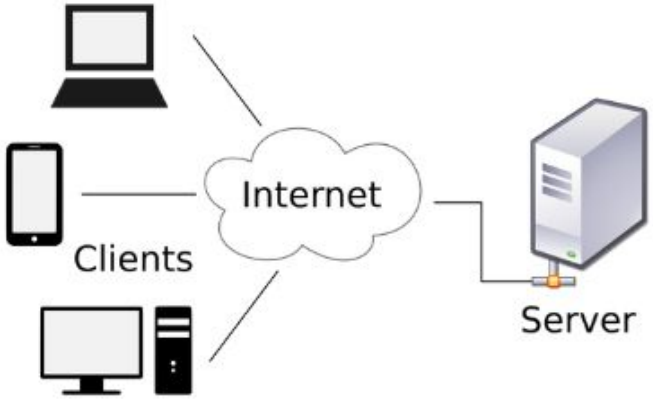
EXAMPLE OF DISTRIBUTED SYSTEM

1. Internet- largest collection of interconnected computer networks.

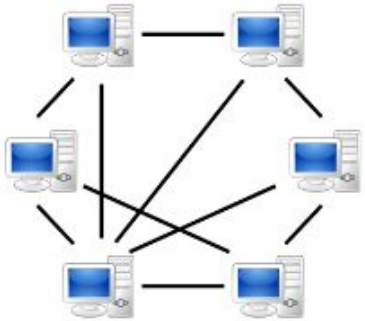
1. Mobile Networks- network is distributed over land areas called cells.

1. Distributed Database System- data stored in multiple devices

TYPES OF DS



Client-Server



Peer-to-Peer

CHARACTERISTICS OF DISTRIBUTED SYSTEM

- Resource Sharing
- Heterogeneity
- Independent Failure
- Fault Tolerance
- Scalability

ADVANTAGES OF DS

1. COST EFFECTIVE

1. LESSER DELAY

1. EASY SCALING

1. FLEXIBLE FUNCTIONALITY

1. FAULT TOLERANCE

DISADVANTAGES OF DS

- It is very difficult to implement the distributed system making it more costlier than other systems.
- Security Risk
- Complex Strategy
- Overloading
- Network Error

DESIGN CHALLENGES OF DISTRIBUTED SYSTEM

Heterogeneity: Heterogeneity is applied to the network, computer hardware, operating system, and implementation of different developers.

Openness: The openness of the distributed system is determined primarily by the degree to which new resource-sharing services can be made available to the users.

Scalability: The scalability of the system should remain efficient even with a significant increase in the number of users and resources connected.

Security: The security of an information system has three components Confidentially, integrity, and availability. Encryption protects shared resources and keeps sensitive information secrets when transmitted.

Failure Handling: When some faults occur in hardware and the software program, it may produce incorrect results or they may stop before they have completed the intended computation so corrective measures should to implemented to handle this case.

Concurrency: There is a possibility that several clients will attempt to access a shared resource at the same time. Multiple users make requests on the same resources, i.e. read, write, and update. Each resource must be safe in a concurrent environment.

Transparency: The user should be unaware of where the services are located and the transfer from a local machine to a remote one should be transparent.

DESIGN GOALS

-Resource Sharing

-Openness

-Transparency

-Scalability

TYPES OF DISTRIBUTED SYSTEM

- Distributed Computing Systems
  - Clusters
  - Grids
  - Clouds
- Distributed Information Systems
  - Transaction Processing Systems
  - Enterprise Application Integration
- Distributed Embedded Systems
  - Home systems
  - Health care systems
  - Sensor networks

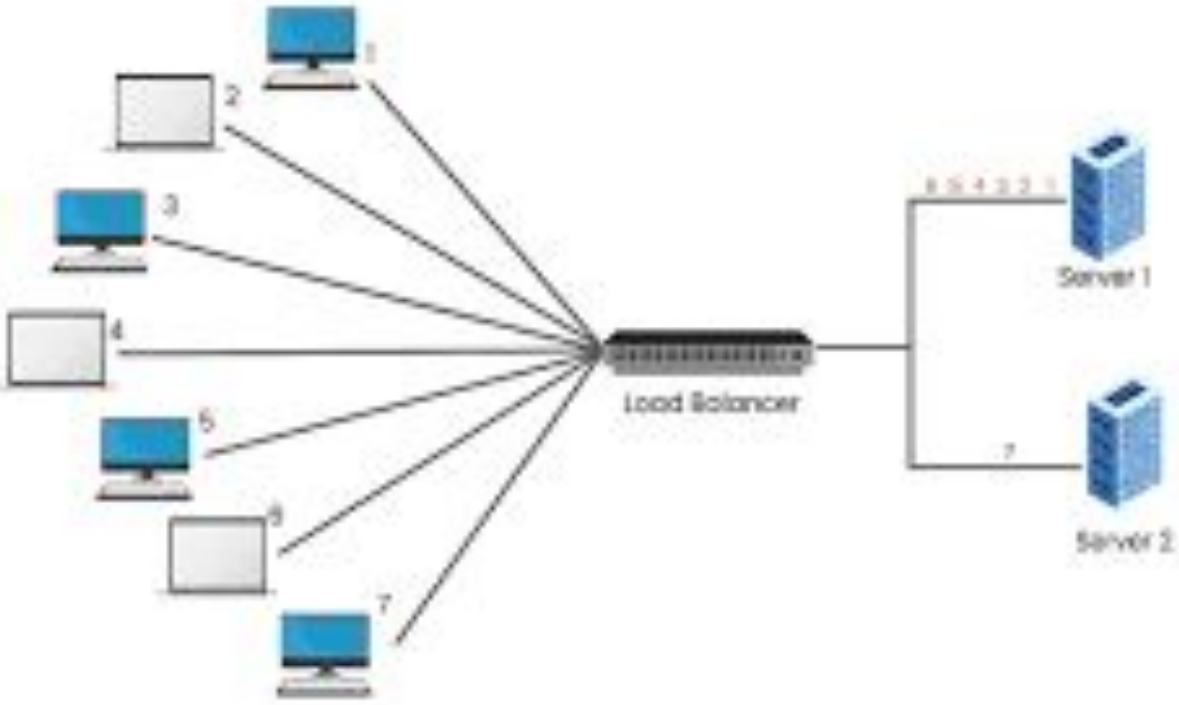
DISTRIBUTED COMPUTING SYSTEM

A distributed computing system is a network of multiple computers or nodes working together to achieve a common goal.

This distributed system is used in performance computation which requires high computing.

Cluster Computing: A collection of connected computers that work together as a unit to perform operations together, functioning in a single system.

Clusters are generally connected quickly via local area networks & each node is running the same operating system.


1
2
3
4
5
6
7
Load Balancer
Server 1
Server 2

FEATURES

Collection of similar workstations/PCs, closely connected by means of a high-speed LAN:

- Each node runs the same OS.
- Homogeneous environment(*computers using similar configurations and protocols*
- Can serve as a supercomputer
- Excellent for parallel programming

Examples: Linux-based Beowulf clusters, MOSIX (from Hebrew University).

ADVANTAGES

- High Performance
- Easy to manage
- Scalable
- Expandability
- Availability
- Flexibility
- Cost effectiveness

DISADVANTAGES

High cost

The problem in finding the fault

More space is needed

Increased infrastructure needed

APPLICATION

- In many web applications functionalities such as Security, Search Engines, Database servers, web servers, proxy, and email.
- Assist and help to solve complex computational problems
- Cluster computing can be used in weather modeling
- Earthquake, Nuclear, Simulation, and tornado forecast

GRID COMPUTING

Grid Computing is a computing infrastructure that combines computer resources spread over different geographical locations to achieve a common goal

-use of widely **distributed computer** resources to reach a common goal.

-each system can belong to a different administrative domain and can differ greatly in terms of hardware, software, and implementation network technology.

Characteristics

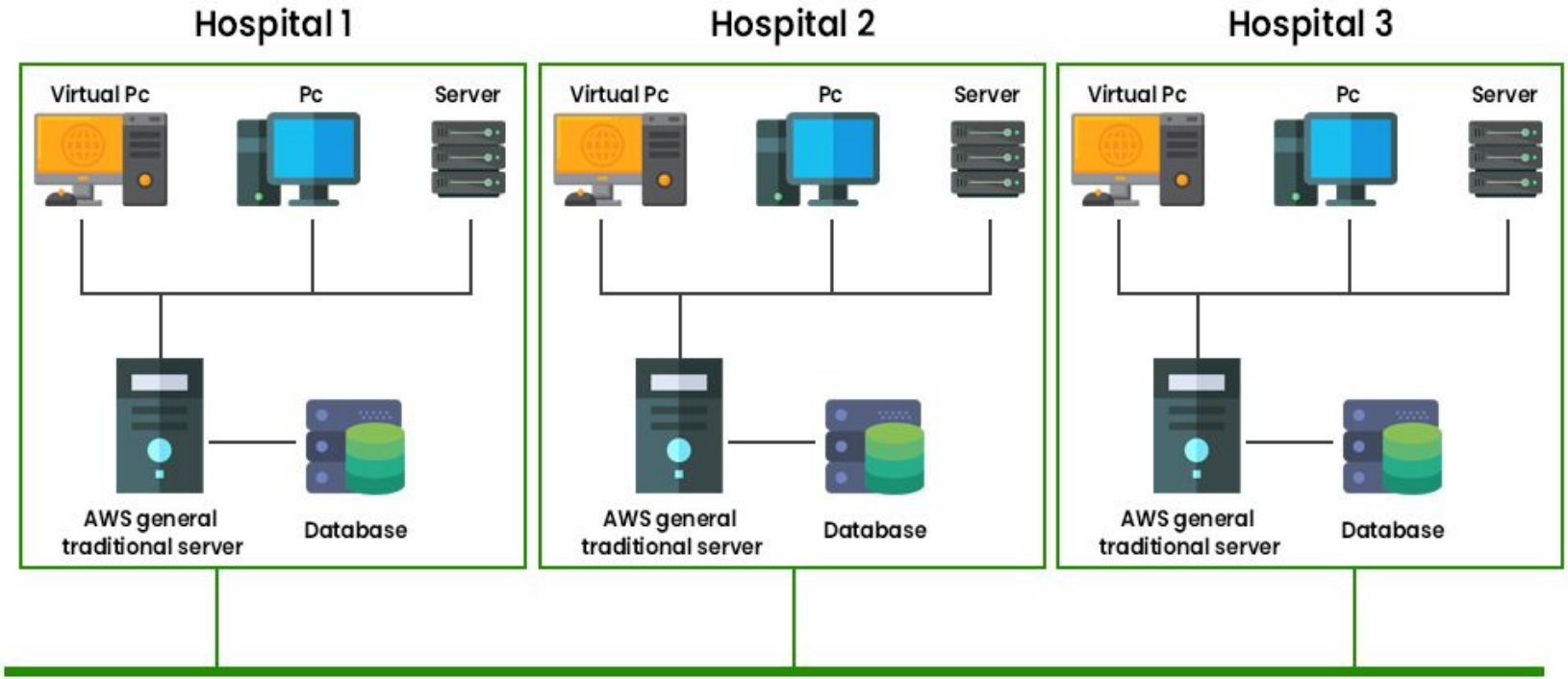
1. **Resource Sharing** –
   Multiple heterogeneous resources (computers, storage, applications, data) are shared across different organizations and locations.

2. **Heterogeneity** –
   Grid computing integrates different types of resources (PCs, clusters, supercomputers, storage devices) running on different operating systems and architectures.

3. **Geographical Distribution** –
   Resources are located in different geographical locations but are connected through a network (usually the Internet).

4. **High Performance & Parallel Processing** –
   Tasks are divided into sub-tasks and executed simultaneously across multiple systems, improving speed and efficiency.

5. **Collaboration** –
   Promotes collaboration among organizations, research centers, and individuals by pooling resources.


Hospital 1
Virtual Pc
Pc
Server
AWS general traditional server
Database
Hospital 2
Virtual Pc
Pc
Server
AWS general traditional server
Database
Hospital 3
Virtual Pc
Pc
Server
AWS general traditional server
Database

ADVANTAGES

- Can solve bigger and more complex problems in a shorter time frame.
- Easier collaboration with other organizations and better use of existing equipment
- Existing hardware is used to the fullest.
- Collaboration with organizations made easier

CLOUD COMPUTING

Cloud computing is the **on-demand delivery of computing resources** (like servers, storage, databases, networking, software, and AI) over the internet (“the cloud”) with **pay-as-you-go pricing**.

Instead of owning and maintaining physical data centers or servers, users can rent computing power and services from cloud providers like **Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP)**.

Characteristics of Cloud Computing

1. **On-demand self-service** – Users can get computing resources instantly.
2. **Broad network access** – Services are available over the internet from anywhere.
3. **Resource pooling** – Multiple users share resources (multi-tenancy).
4. **Rapid elasticity** – Resources can scale up or down automatically.
5. **Measured service** – Pay only for what you use.

Types of Cloud Service Models

1. **IaaS (Infrastructure as a Service):**
    - Virtual machines, storage, networks.
    - Example: AWS EC2, Google Compute Engine.
2. **PaaS (Platform as a Service):**
    - Developers build apps without managing servers.
    - Example: Google App Engine, Heroku.
3. **SaaS (Software as a Service):**
    - End users use software over the internet.
    - Example: Gmail, Microsoft 365, Dropbox.

Types of Cloud Deployment Models

1. **Public Cloud** – Services offered over the public internet (AWS, Azure).
2. **Private Cloud** – Exclusive to a single organization (banks, govt.).
3. **Hybrid Cloud** – Mix of public + private.
4. **Community Cloud** – Shared by multiple organizations with a common purpose.

DISTRIBUTED INFORMATION SYSTEM

A set of information systems physically distributed over multiple sites, which are connected with some kind of communication network

Examples of distributed computing and information systems are :

systems that automate the operations of commercial enterprises such as banking and financial transaction processing systems

DISTRIBUTED TRANSACTION PROCESSING

A transaction process system (TPS) is an information processing system for business transactions involving the collection, modification and retrieval of all transaction data.

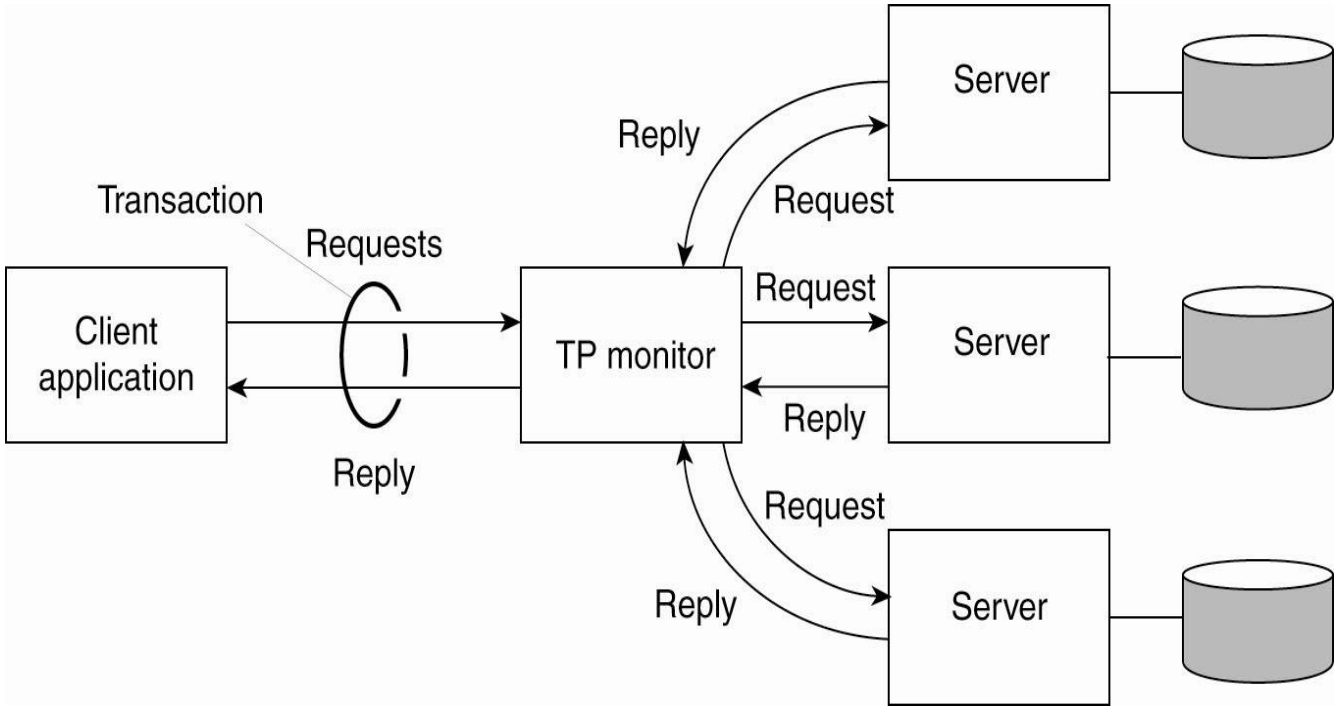
It works across different servers using multiple communication models. The four characteristics that transactions have:

Atomic

Consistent

Isolated

Durable


Server
Reply
Request
Transaction
Requests
Client application
TP monitor
Request
Server
Reply
Reply
Request
Reply
Server

ENTERPRISE APPLICATION INTEGRATION

- **Enterprise Application Integration (EAI)** is the process of linking different enterprise applications (ERP, CRM, SCM, databases, legacy systems, etc.) within an organization so that they can work together and share information seamlessly.

Characteristics of EAI

1. **Integration of Heterogeneous Systems** – connects different applications (legacy systems, cloud apps, databases).
2. **Data Sharing** – ensures consistent and real-time data flow between applications.
3. **Process Automation** – enables end-to-end business process automation.
4. **Middleware Usage** – often uses middleware (message brokers, APIs, ESB) to connect systems.
5. **Scalability & Flexibility** – supports growing business needs without major system changes.
6. **Transparency** – users get a unified view without worrying about underlying complexity.

ADVANTAGE

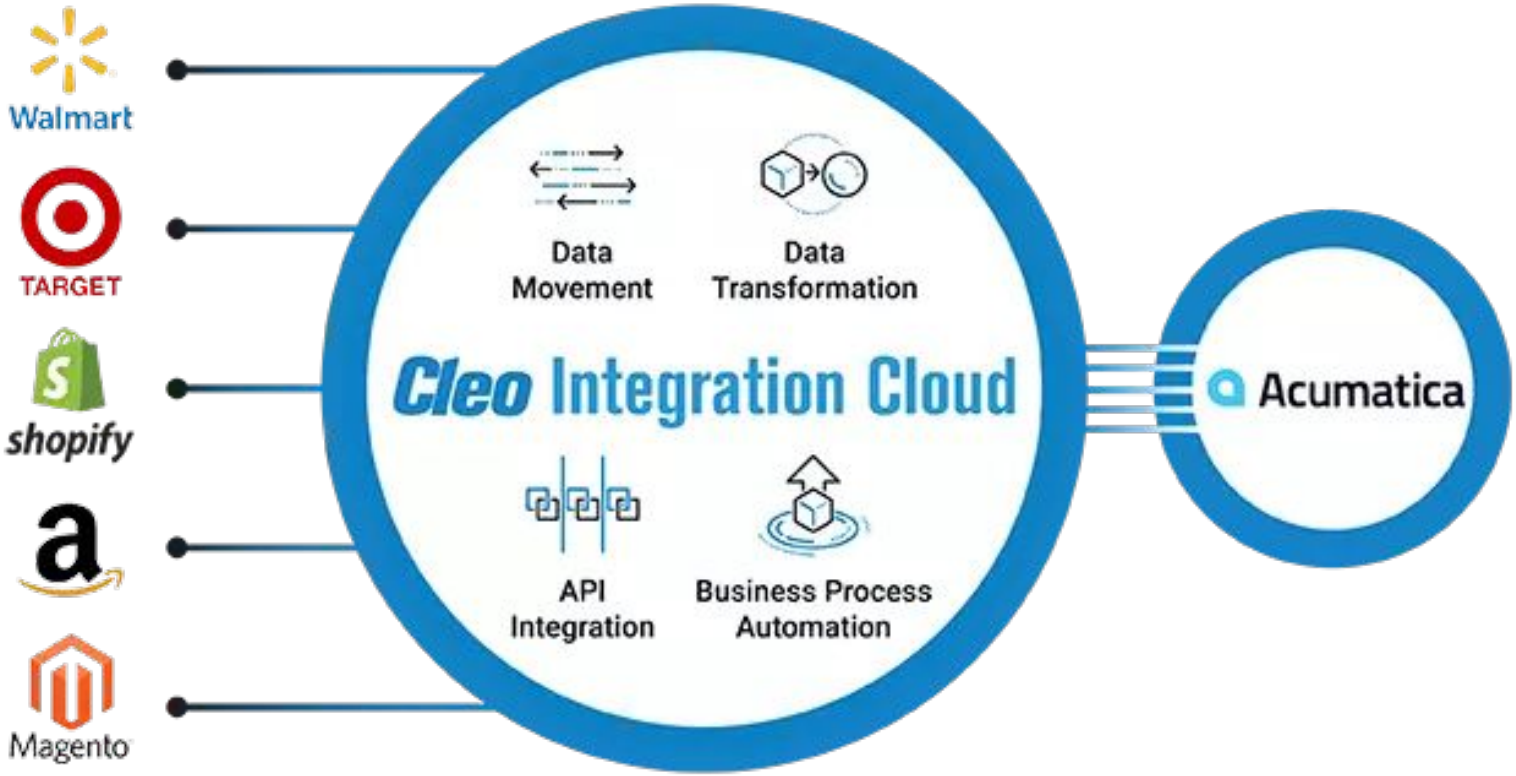
- Promotes information flow.
- Improves **efficiency** and reduces manual work.
- Enhances **decision-making** with real-time data access.
- Increases **agility** to adapt to changing business needs.
- Enables **reuse of existing IT investments** (legacy systems).

Applications of EAI

- Linking ERP with CRM (e.g., SAP + Salesforce).
- Synchronizing financial, HR, and supply chain systems.
- Real-time data exchange between online sales and warehouse systems.
- Integrating cloud services with on-premise applications.

EXAMPLE

In the example below, you can see how Cleo's EAI system connects external-facing applications like Amazon Vendor Portal, Shopify, and Magento, along with EDI trading partners like Walmart and Target, to the back-end ERP system, which is Acumatica.


Walmart
TARGET
shopify
Magento
Data Movement
Data Transformation
Cleo Integration Cloud
API Integration
Business Process Automation
Acumatica

DISTRIBUTED PERVASIVE SYSTEM

Pervasive Computing is a step towards integrating everyday objects with microprocessors so that this information can communicate.

It’s a system of **interconnected devices** (sensors, mobiles, IoT devices, wearables, smart appliances) that work together, often without direct human intervention, to provide intelligent services.

Characteristics

1. **Distribution** – Devices/resources are spread across multiple locations.
2. **Pervasive** – Computing devices are integrated into daily environments (homes, offices, hospitals, cities).
3. **Heterogeneity** – Mix of different devices (sensors, smartphones, IoT nodes, servers).
4. **Context-awareness** – System adapts based on user's situation (location, time, activity).
5. **Autonomy** – Devices operate independently but collaborate when needed.
6. **Scalability** – Can support thousands or millions of devices.
7. **Transparency** – Users don't notice the complexity; interaction feels natural.
8. **Mobility support** – Works with moving users/devices (wearables, vehicles).
9. **Fault tolerance** – System continues functioning even if some devices fail.

Home system: Nowadays many devices used in the home are digital so we can control them from anywhere and effectively.

Electronic health system: Nowadays smart medical wearable devices are also present through which we can monitor our health regularly.

Sensor network (IoT devices): Internet devices only send data to the client to act according to the data send to the device.

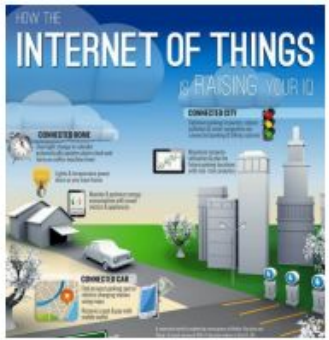
Examples

- **Smart Homes** (lights, AC, security cameras working together).
- **Smart Cities** (traffic sensors, surveillance, waste management).
- **Healthcare Systems** (wearables + hospital monitoring).
- **IoT-based Agriculture** (sensors for soil, water, weather).

Challenges

1. **Security & Privacy** – Sensitive user data can be exposed.
2. **Resource Constraints** – Many devices have limited battery/storage.
3. **Interoperability** – Different devices and platforms must work together.
4. **Scalability** – Managing huge numbers of devices.
5. **Reliability** – Ensuring continuous service despite failures.

- Internet of things
  - ability to network devices and have them communicate
- Sensor networks
  - Large networks of sensors
- Driven by miniaturization of computing
  - Tiny sensors with computing and communication capability



- Smart home


hue
PHILIPS
hue


nest
72


we mo


Introducing your new garage door opener.
FREE APP
NO FEES
LiftMaster

Personal Health Monitoring

- Sensors to monitor fitness, diabetes, blood pressure, detect falls

Google tests prototype of diabetes-tracking 'smart' contact lens

Advantages of Distributed Systems over Centralized Systems

- **Economics**: a collection of microprocessors offer a better price/performance than mainframes. Low price/performance ratio: cost effective way to increase computing power.
- **Speed**: a distributed system may have more total computing power than a mainframe. Ex. 10,000 CPU chips, each running at 50 MIPS. Not possible to build 500,000 MIPS single processor since it would require 0.002 nsec instruction cycle. Enhanced performance through load distributing.
- **Inherent distribution**: Some applications are inherently distributed. Ex. a supermarket chain.
- **Reliability**: If one machine crashes, the system as a whole can still survive. Higher availability and improved reliability.
- **Incremental growth**: Computing power can be added in small increments. Modular expandability

Students Work

Can you explain the working of Pervasive Distributed System ?

TRANSPARENCY

A transparency is some aspect of the distributed system that is hidden from the user (programmer, system developer, user or application program).

WHY TRANSPARENCY IS REQUIRED ?

1. Location transparency enables resources to be accessed without knowledge of their physical or network location (for example, which building or IP address).
2. Concurrency transparency enables several processes to operate concurrently using shared resources without interference between them.

1. Replication Transparency – In distributed systems to achieve fault tolerance, replicas of resources are maintained. The Replication transparency ensures that users cannot tell how many copies exist.