@BUNDICHRISTOPHER

LESSON 1
CLIENT SERVER COMPUTING NOTES

In client server computing, the clients requests a resource and the server provides that resource. A server may serve multiple clients at the same time while a client is in contact with only one server. Both the client and server usually communicate via a **computer network** but sometimes they may reside in the same system.

An illustration of the client server system is given as follows −
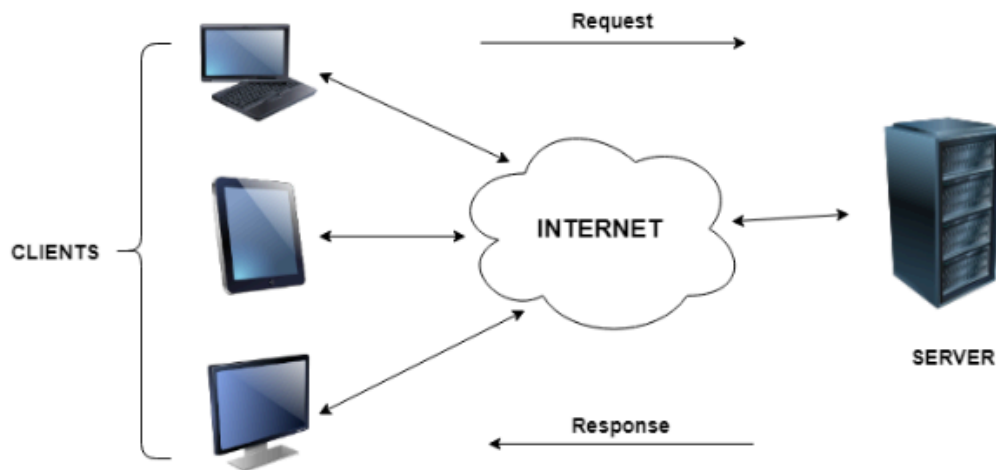
Characteristics of Client Server Computing

The salient points for client server computing are as follows:

The client server computing works with a system of request and response. The client sends a request to the server and the server responds with the desired information.

The client and server should follow a common communication protocol so they can easily interact with each other. All the communication protocols are available at the application layer.

A server can only accommodate a limited number of client requests at a time. So it uses a system based to priority to respond to the requests.

Denial of Service attacks hindera servers ability to respond to authentic client requests by inundating it with false requests.

An example of a client server computing system is a web server. It returns the web pages to the clients that requested them.

Difference between Client Server Computing and Peer to Peer Computing

The major differences between client server computing and peer to peer computing are as follows:

In client server computing, a server is a central node that services many client nodes. On the other hand, in a peer to peer system, the nodes collectively use their resources and communicate with each other.

In client server computing the server is the one that communicates with the other nodes. In peer to peer to computing, all the nodes are equal and share data with each other directly.

Client Server computing is believed to be a subcategory of the peer to peer computing.

Advantages of Client Server Computing

The different advantages of client server computing are −

All the required data is concentrated in a single place i.e. the server. So it is easy to protect the data and provide authorisation and authentication.

The server need not be located physically close to the clients. Yet the data can be accessed efficiently.

It is easy to replace, upgrade or relocate the nodes in the client server model because all the nodes are independent and request data only from the server.

All the nodes i.e clients and server may not be build on similar platforms yet they can easily facilitate the transfer of data.

Disadvantages of Client Server Computing

The different disadvantages of client server computing are −

If all the clients simultaneously request data from the server, it may get overloaded. This may lead to congestion in the network.

If the server fails for any reason, then none of the requests of the clients can be fulfilled. This leads of failure of the client server network.

The cost of setting and maintaining a client server model are quite high.

LESSON 2:

Operating Systems Client/Server Communication

Client/Server communication involves two components, namely a client and a server. They are usually multiple clients in communication with a single server. The clients send requests to the server and the server responds to the client requests.

There are three main methods to client/server communication. These are given as follows −

Sockets

Sockets facilitate communication between two processes on the same machine or different machines. They are used in a client/server framework and consist of the IP address and port number. Many application protocols use sockets for data connection and data transfer between a client and a server.

Socket communication is quite low-level as sockets only transfer an unstructured byte stream across processes. The structure on the byte stream is imposed by the client and server applications.

A diagram that illustrates sockets is as follows −
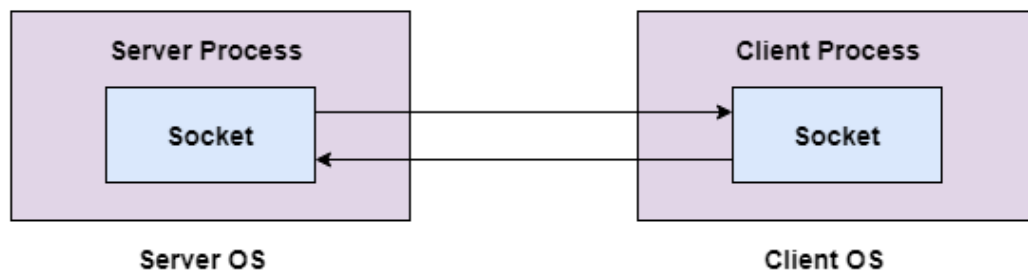
Remote Procedure Calls

These are interprocess communication techniques that are used for client-server based applications. A remote procedure call is also known as a subroutine call or a function call.

A client has a request that the RPC translates and sends to the server. This request may be a procedure or a function call to a remote server. When the server receives the request, it sends the required response back to the client.

A diagram that illustrates remote procedure calls is given as follows −
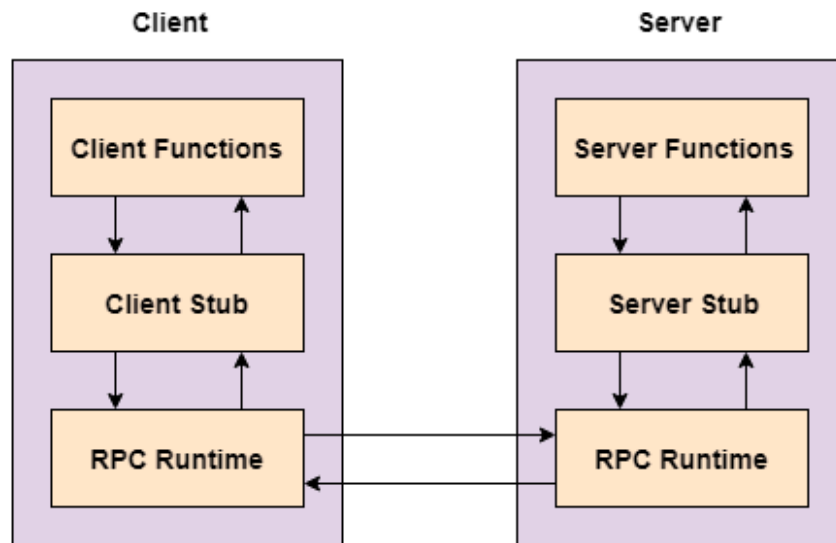
Pipes

These are interprocess communication methods that contain two end points. Data is entered from one end of the pipe by a process and consumed from the other end by the other process.

The two different types of pipes are ordinary pipes and named pipes. Ordinary pipes only allow one way communication. For two way communication, two pipes are required. Ordinary pipes have a parent child relationship between the processes as the pipes can only be accessed by processes that created or inherited them.

Named pipes are more powerful than ordinary pipes and allow two way communication. These pipes exist even after the processes using them have terminated. They need to be explicitly deleted when not required anymore.

A diagram that demonstrates pipes are given as follows −

Parent Process

Child Process

LESSON 3

Server Operating System

Server Operating System is a type of Operating system with advanced features that can be used only on servers. It is designed to use in-server computers to handle various users and requests at a time. When a person or company requires data it cannot be served using a human instead the solution is given by a machine.

Server Operating System

With one server and it can be connected to many receivers or clients concurrently using the server Operating system. It is also called Client Server based architecture for smooth communication and exchange of information. The primary use of it is that the users get benefitted from more usage and memory.

Features of Server OS

Server OS is designed to be used on the server

It has greater Processing power.

It is more stable and highly efficient

In this system we can have one server connected to many clients and provide service to them.

It is a complex operating system.

It provides more security.

Examples of Server OS

Some of the Server Operating System is,

File Server

A File Server is deployed in the computer which involves the process of storing and managing files on the network. From this server, multiple users can access the files and also share them.

Web Server

A Web server is a system that shares HTML-programmed web pages with the clients. It processes the system based on the request from the browsers and in return gives the data such as pictures and HTML pages.

Database Server

A Database server is a computer system used to store data and provides services to other computers or programs. Some of the functionalities in the database server are storing, organizing, and managing the data in the system.

Traditional Server

Traditional Server is used by small or medium organization with limited users and the servers are physically present in the organizations

Directory Server

This type of server uses a directory which is particularly used in UNIX and Linux Operating systems and it contains configuration files of the system and applications.

Rack-Monitored Server

Usually, rack-monitored servers will have a single monitor and an interface like a mouse. The number of servers in a rack depends on the height of the rack and server.

Blade Server

It is more economical or slim in space as compared to rack-mounted servers and can handle hundreds of blade servers

Red Hat Enterprise Linux Server

The Operating system provides more features than any other servers and is currently used worldwide. Technology like automation is supported and some security features are secured profiles and also provide certification for the secured connections in the server.

Windows Server

Windows OS has got a remarkable place for both professional and personal users. The storage provided by Windows OS is huge and it can be managed efficiently. Some of the threats faced by this OS are users should have a license for using it and are prone to virus or malware attacks easily.

Server Operating system architecture

The Architecture used in a server operating system primarily comprises of client and server. The server provides services and resources to the clients, allowing them to share the same resources simultaneously. A server operating system is designed to facilitate this type of communication and resource sharing. Users can access the server's resources from anywhere on the network, as long as they have the necessary permissions and authentication.

Advantages of Server Operating System

The users accessing the server Operating system have the advantage of choosing between the two interfaces namely GUI and command-level interfaces.

This allows the server to provide a wide range of services to users

The designs of the server OS are based on the commands received and apart from the client OS, the Server OS has administrators to perform and manage the tasks like server hardware, software, and network settings using tools.

It provides a central interface for managing users, implementing security measures, and performing other administrative tasks. This allows administrators to easily manage the server and ensure its security

Server operating systems can manage and monitor client computers and operating systems. This allows administrators to keep track of the status of client devices and ensure that they are functioning properly.

Client Operating System (OS)

Client operating system is a system which works within computer desktops and other portable devices such as laptops and smartphones. Operating systems that are installed on the client's computer device like PC, Laptop are called client OS.

This operating system helps computers to run.

Client OS manages different hardware components that are connected to printers, monitors and cameras.

It supports a single user at a time.

It is also capable of obtaining services from a server operating system.

When compared to a server operating system it helps in providing multiprocessing power for a minimum price.

The examples of client operating systems are − Windows, Mac.

Server Operating System (OS)

A server operating system is an operating system that operates on the server. Generally, we can say server OS is an advanced version of an operating system which has more features and capabilities to provide various services to other devices or client machines connected to it.

In server operating systems, most processes execute from the OS commands. It helps to install and deploy business and web applications and is also capable of managing and monitoring the connected client devices. The server operating systems execute other administrative processes.

Given below is the diagram of Server OS −
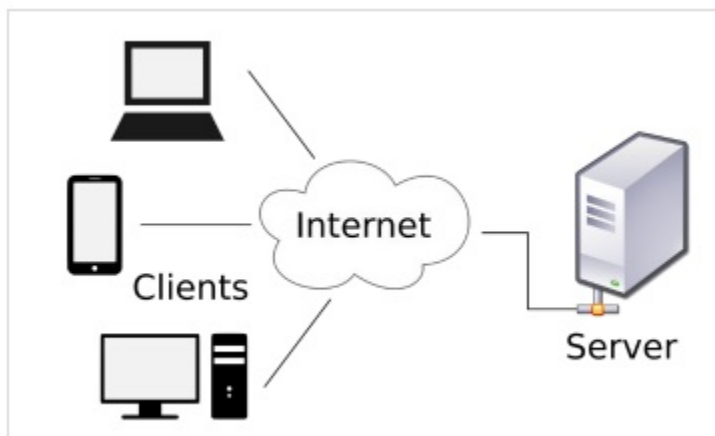
Types of Server OS

The different types of server OS are as follows −

> Web server operating systems that provide web resources like web pages based on client requests.
>
> Database server operating systems which runs Database Management Systems
>
> File server operating systems that provide files to the connected devices.

So, server operating systems provide services to client machines.

Differences

The major differences between client OS and Server OS are as follows −

A client machine is simple and inexpensive on the other hand server OS is expensive and more powerful.

The main difference between both is the performance. There are multiple clients working on a single server. Clients send requests for network access to the server and a client is an end user.

A server operating system is an OS to run on a server, they are special OS that work on client/server architecture and serve for requests from the client side.

If a client uses a browser to search information then a server sends the source to the client to search directly.

Difference on the base of security (SAM vs DC)

SAM is a database in OS that contains the user name and password. it is a part of which can be found on a hard disk.

In SAM each end user has assigned a LAN password and window password and when a user wants to login and enter a name and password then it matches into the SAM.

If a match occurs it opens the account otherwise it gives an error. DC OS is used to secure and protect the internal cluster communication. It protects all the authorised interfaces. It protects clusters from breaches and attacks.

Storage (Centralized vs Decentralized)

Client OS and Server OS are different on the basis of storage.

Server OS is based on centralized storage.

In this server OS is centralized and a powerful machine is there which stores all data and on the other hand clients are decentralized, because clients are end users and store their personnel data on the decentralized place.

In this all the decentralized clients are connected with the centralized server, and can access all the data from the server.

Server has a big amount of storage and the client has limited storage related to the end user.

Network Model (Work-group vs Domain)

In a network model we have clients which are connected on the same network to make a work group. A project is divided into different clients. Server is also known as domain, because on a single domain many groups work and take information from the domain they can share information. Work groups take help from the domain. So, we can say that the work group is a small part of a big domain. Work groups access the domain always.

Difference between Client and Server

In a client-server architecture, clients request services from a server. Servers provide the requested service to their clients. However, clients and servers are separate programs running on different machines.

A **client** is a piece of software that runs on a device and gives the user access to a server. The server provides the requested services upon request from the customer. A web browser, like Google Chrome or Mozilla Firefox, is a customer that asks a web server for webpages.

A **server** is a piece of software that runs on a device and makes services available to other devices which is connected to a network. It receives customer requests, and handles them, also provides the requested data or issues. A web server is a piece of software which responds to the customer requests by storing and delivering web pages and other sites via the internet.

Clients act as the front end, and servers remain at the back end. Clients are in charge of delivering the user with data, whereas servers are in charge of storing and managing the data.

Read this article to find out more about Clients and Servers and how they are different from each other.

What is a Client?

A computer device that asks a server for services is known as a customer. The customer is the front end element in client and server architecture which communicates with the user and sends requests to the server. The back end which satisfies or gives the customer's requests and offers the needed services is knowns as the server.

Initiating and managing requests for resources or services from a server is the primary responsibility of a client. The client software is in charge of informing the user, structuring requests, and processing server responses. The client-server design is popular because it offers scalability and allows for the separation of duties because multiple clients can send requests to the same server What is a Server?

A server is a computer software or device that offers services to other devices connected to a network. A server's primary liabilities include accepting and completing customer requests as well as storing and managing data.

The crucial benefit of using servers is that they give you a central position to manage and store your data, which makes it simpler for you to maintain. also, it offers an advanced position of trust ability because they can keep running indeed if one of the customer does not. These are suited for large- scale operations because they can handle massive volumes of data and serve a large number of guests.

Servers play a pivotal role in the customer-server architecture because they provide other networked gadgets access to services. customer requests are entered by servers, which also reuse them and store and manage data.

Difference between Client and Server

Characteristics	Client	Server
Definition	A computer device which asks a server for services is known as a customer. The customer is the front end element in client and server architecture which communicates with the user and sends requests to the server.	A server offers services to other devices connected to a network. A server's primary functions include accepting and completing customer requests as well as storing and managing data.
Support of Login	Clients support the login of single users.	Servers support multiple user login.
Example	Some of the examples of client are laptops, desktop,etc.	Some of the examples of server are web servers,database servers, and file servers.
Performance	The client performance is	The server performance is

	less when we compare it with the server.	high when we compare it with the client.
Tasks	The task mainly includes the requesting services.	The tasks are more complex in the server when we compare to a client. Because it fulfills the client request, and it stores and processes the large datasets, and the data analysis.
Configuration	It contains the simple hardware configuration, because it performs the simple tasks like the generating the request which are raised by the client.	It is quite opposite to the client when it comes to the configuration. It uses the most advanced hardware configuration because it performs the most complex tasks which are requested by the client.
Functionality	It completely depends on the services of server because the server generates the services requested by the client.	It sends the requested services to the client based on their requirements.

Advantages and Disadvantages of Three-tier Architecture

Advantages of Three−Tier Architecture

Following are the main advantages of three−tier architecture

Data integrity is enhanced. Here, the second tier is where all the updated information passes. It can make sure that only crucial data is permitted to

be altered in the database, eliminating the possibility that unreliable client apps will corrupt data.

As the client does not have direct access to the database, security is increased because it is more challenging for a client to collect unauthorized data. Because business logic is kept on a secure central server, it is more secure.

Applications can use easily scaled components to take advantage of the modular architecture of enabling systems, increasing availability.

Due to the separation of tasks, web designers handle presentations, software engineers handle logic, and database administrators handle data models.

Each tier is independent of the others, allowing for updates or changes to be made without having an impact on the programme as a whole.

Any code defined once in the business layer can be shared by as many components in the presentation layer as there are business layer components.

Only essential business layer methods can be displayed in the presentation layer.

Before updating the dat[1]abase, presented data can be checked or confirmed at the application layer.

Requesters frequently don't see the database's real structure, making any database changes transparent. As a result, a middle−tier process that trades data with other applications can maintain its current user interface even as the underlying database structure is changed.

Disadvantages of three−tier Architecture

Following are the disadvantages of three−tier architecture

---

[1] @bundichristopher639@gmail.com

The three−tier structure makes it difficult for developers to make updates to an application with the agility and flexibility necessary to keep up with the needs of mobile users as well as for operations teams to scale the service up or down to meet demand.

An application with a three−tier design is less agile at different stages of the application development process. Although an application's functionality may be broken down into modules, altering any one of them means rebuilding and testing the entire thing. This may require a lot of work.

A strong understanding of the object−oriented concept is required.

Because it takes more work to create a 3−tier application than a 2−tier application, it is more complicated than the 2−tier client−server computing model. There are two additional sources of communication.

There may be a need for a distinct proxy server.

Supporting distributed or remote access to data tools is not acceptable.

Sometimes the monolithic nature of the code increases the maintenance part.

When building 3−tier applications, more effort should typically be implemented due to the increased number of contact points.

Increased performance necessitates more work, whereas the 2−tier model can handle the specific job with an automated tool.

Since the majority of application logic is controlled by the client, the issues occur while managing software versions and distributing updated versions.

@bundichristopher6