Haroon Rashid

Reg#: 16549

Semester:6th

Sessional Assignment: Software Design & Architecture

Submitted To: Madam Aasma Khan

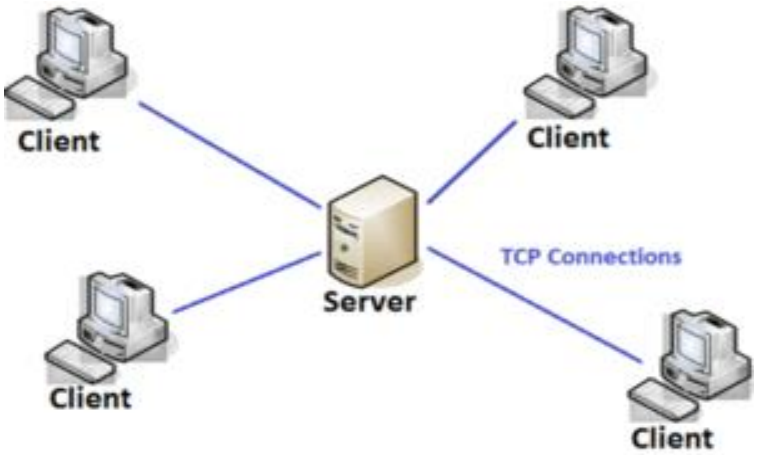
Question No: 01

Choose an software architectural style of your choice and give its explanation that must cover the below given components of an architecture style:

- **Elements/components**
  - **that perform functions required by a system**
- **Connectors**
  - **that enable communication, coordination, and cooperation among elements**
- **Constraints**
  - **that define how elements can be integrated to form the system**
- **Attributes**
  - **that describe the advantages and disadvantages of the chosen structure.**

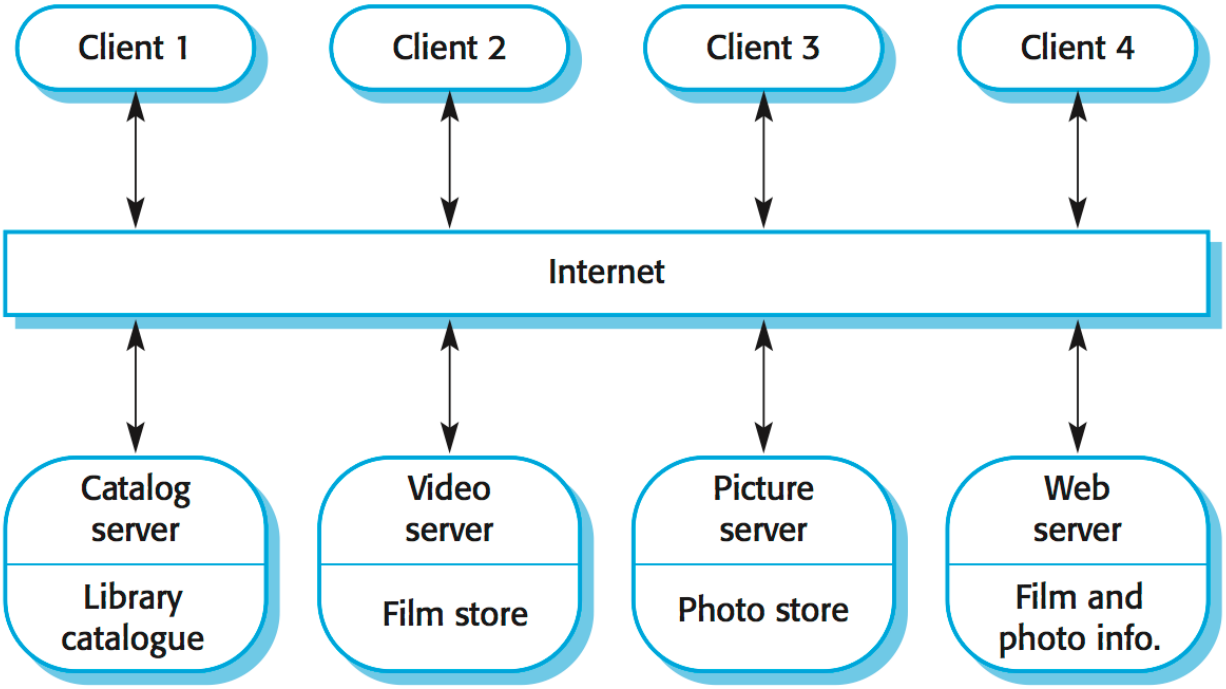
Answer:

Client Server Architecture: Client Server Architecture is a computing model in which the server hosts, delivers and manages most of the resources and services to be consumed by the client. This type of architecture has one or more client computers connected to a central server over a network or internet connection. This system shares computing resources. Client/server architecture is also known as a networking computing model or client/server network because all the requests and services are delivered over a network.



Client-server architecture is an architecture of a computer network in which many clients (remote processors) request and receive service from a centralized server (host computer). Client computers provide an interface to allow a computer user to request services of the server and to display the results the server returns. Servers wait for requests to arrive from clients and then respond to them. Ideally, a server provides a standardized transparent interface to clients so that clients need not be aware of the specifics of the system (i.e., the hardware and software) that is providing the service. Clients are often situated at workstations or on personal computers, while servers are located elsewhere on the network, usually on more powerful machines. This computing model is especially effective when clients and the server each have distinct tasks that they routinely perform. In hospital data processing,

for example, a client computer can be running an application program for entering patient information while the server computer is running another program that manages the database in which the information is permanently stored. Many clients can access the server's information simultaneously, and, at the same time, a client computer can perform other tasks, such as sending e-mail. Because both client and server computers are considered intelligent devices, the client-server model is completely different from the old "mainframe" model, in which a centralized mainframe computer performed all the tasks for its associated "dumb" terminals.



1. **Elements/components:**

Components of a Client Server

A client/server network has three main components: workstations, servers and the network devices that connect them.

Workstations

Workstations, or client computers, initially differentiate themselves by the operating systems running them. In a client/server network, Windows 2000, Windows XP, Windows Vista and Windows 7 are examples of workstation operating systems. Aside from being relatively cheaper than server operating systems, their functions and processes are essentially intended for client computers. Centralized databases, shared programs, management and security policies are not part of their operating systems. What they have are localized versions of databases, programs and policies that can be applied individually to them. Workstations also have lower technical specifications than servers in the areas

memory, hard drive space and processor speed, because they are not required to process requests or record data from multiple computers.

Servers

Servers are distinguished by different sets of operating systems like Windows 2000 Server, Windows 2003 or Windows 2008. They also have higher memory and hard drive space and faster processors because they store and service multiple (and often simultaneous) requests from workstations. A server can assume many roles in a client/server network. It can be a file server, a mail server, a database server and domain controller all at the same time. A well-set-up network, however, delineates these roles to different servers to optimize performance. A server, regardless of what role it has, essentially acts as a centralized repository of network files, programs, databases and policies. It makes for easier management and backup because it is not dependent to individual user configurations, but can be universally and uniformly implemented across the network.

Network Devices

Network devices connect workstations and servers. They ensure that requests to and from workstations are routed properly to the correct server. Several network devices each provide different types of network connectivity. In a simple client/server network, a hub can connect a server to multiple workstations. It acts as a repeater, passing on data from one device to another. Bridges separate network segments. This is useful for offices with several departments to distinguish which department a particular workstation belongs to. Another network device, a switch, is similar to a bridge, but can detect conflicts between network segments like same IP addresses or computer names across departments. Wide-area networks use routers to connect network segments in different locations. Routers are also used to connect networks, or route information to the Internet.

Other Components:

Client/server networks usually have network printers or scanners, which are shared and can be used by all computers in the network. Instead of installing them individually to each computer, they can be placed in one location that everyone can access. This saves both space and money.

2. Connectors

Connectors of Client Server Protocols

1.Protocols
2.Remote procedure calls (RPC

Protocols

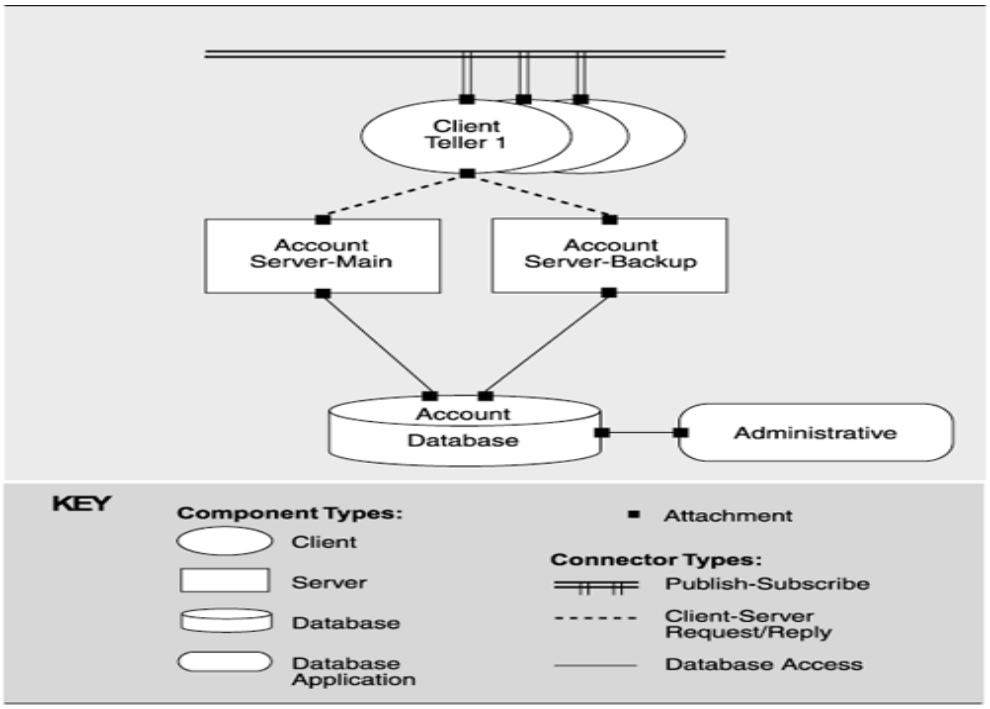
A communications protocol that provides a structure for requests between client and server in a network. For example, the Web browser in the user's computer (the client) employs the HTTP protocol to request information from a website on a server.

.

Remote procedure calls (RPC)

A remote procedure call is an interprocess communication technique that is used for client-server-based applications. It is also known as a subroutine call or a function call. A client has a request message 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. The client is blocked while the server is processing the call and only resumed execution after the server is finished.

Example:



What is this diagram, backed up by its supporting documentation, attempting to convey? We are being shown a bird's-eyeview of the system as it might appear during runtime. The system contains a shared repository of customer accounts (Account Database) accessed by two servers and an administrative component. A set of client tellers can interact with the account repository servers, embodying a client-server style. These client components communicate among themselves by publishing and subscribing to events. We learn from the supporting documentation that the purpose of the two servers is to enhance reliability: If the main server goes down, the backup can take over. Finally, a component allows an administrator to access, and presumably maintain, the shared-data store.

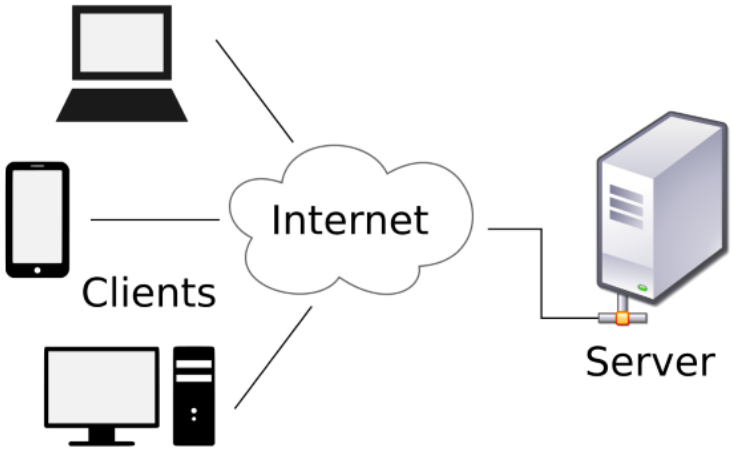
3.Constraints:

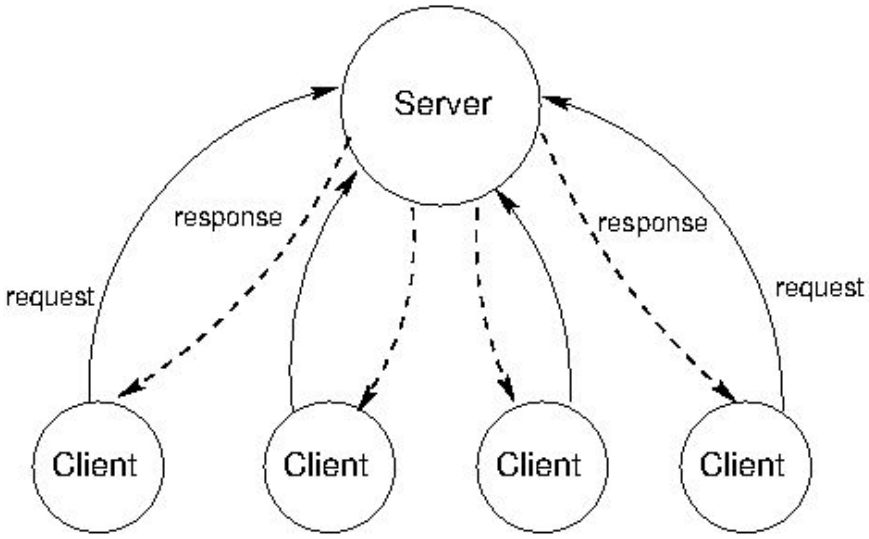
- Two levels, typically many clients with one server.
- From client to server and server to client.
- Constraints:

○ clients cannot communicate directly with each other. If needed, the server acts as a message relay for the clients to communicate.

○ Only clients can initiates communication

- All workloads are done at the server side.





Most applicable to specific kinds of problems (Constraints)?

- Where data can be centralized and easy to do for collaboration
- Where all clients are requesting the same type of data.

● Where clients can give specific information to request different data dynamically

● Less computational burden on the client side, which make client more lightweight.

● When clients are unable to do the heavy computation and the computation are done on the server side.

● Provide better data integrity and backup system, thus higher reliability.

● In general, people can access the data at any time as long as they have network and authorization.

● When mobility is needed, applicated and data can be easily moved and replicated.

4.Attributes

Advantages:

Advantages: Organizations often seek opportunities to maintain service and quality competition to sustain its market position with the help of technology where the client/server model makes an effective impact. Deployment of client/server computing in an organization will positively increase productivity through the usage of cost-effective user interfaces, enhanced data storage, vast connectivity and reliable application services. If properly implemented its capable of improving organizational behavior with the help of the knowledgeable worker-who can manipulate data and respond to the errors appropriately.

- **Improved Data Sharing:** Data is retained by usual business processes and manipulated on a server is available for designated users (clients) over an authorized access. The use of Structured Query Language (SQL) supports open access from all client aspects and also transparency in network services depict that similar data is being shared among users.

- **Integration of Services**: Every client is given the opportunity to access corporate information via the desktop interface eliminating the necessity to log into a terminal mode or another processor. Desktop tools like spreadsheet, power point presentations etc can be used to deal with corporate data with the help of database and application servers' resident on the network to produce meaningful information.

- **Shared Resources amongst Different Platforms:** Applications used for client/server model is built regardless of the hardware platform or technical background of the entitled software (Operating System S/W) providing an open computing environment, enforcing users to obtain the services of clients and servers (database, application, communication servers).

- **Inter-Operation of Data:** All development tools used for client/server applications access the back-end database server through SQL, an industry-standard data definition and access language, helpful for consistent management of corporate data. Advanced database products enable user/application to gain a merged view of corporate data dispersed over several platforms. Rather than a single target platform this ensures database integrity with the ability to perform updates on multiple locations enforcing quality recital and recovery.

- **Data Processing capability despite the location:** We are in an era which undergoes a transformation of machine-centered systems to user-centered systems. Machine-centered systems like mainframe, mini-micro applications had unique access platforms and functionality keys, navigation options, performance and security were all visible. Through client/server users can directly log into a system despite of the location or technology of the processors.

- **Easy maintenance:** Since client/server architecture is a distributed model representing dispersed responsibilities among independent computers integrated across a network, it's an advantage in terms of maintenance. It's easy to replace, repair, upgrade and relocate a server while clients remain unaffected. This unawareness of change is called as encapsulation.

- **Security:** Servers have better control access and resources to ensure that only authorized clients can access or manipulate data and server-updates are administered effectively.

Disadvantages:

● **Single point of failure:** Since there's a reliance on the central server, if it fails, client requests cannot be done.

● **Traffic congestion:** Happens when a large number of simultaneous clients send requests to the same server. This might cause the server to slow down or even shut down.

● **Cost:** The cost of server hardware and software is much greater than the cost of buying desktop hardware and software licences. Thus, it is expensive to scale or even hard to scale

Disadvantages (compared to peer-peer networks)

- **Overloaded servers:** When there are frequent simultaneous client requests, servers severely get overloaded, forming traffic congestion. But in a P2P network adding more nodes will increase its bandwidth since it's calculated as the sum of bandwidths of each node in the network.

- **Impact of centralized architecture:** Since its centralized if a critical server fails, client requests are not accomplished. Therefore client/server lacks robustness of a good P2P network (resources are distributed among many nodes)