

Study Guides on Computer Networks and Distributed Systems

Distributed Systems Models

All rights reserved © 2014-2021 by José María Foces Morán and José María Foces Vivancos

Questionnaire (In this questionnaire, questions marked with a leading (*) are not to be solved now and are not a part covered by the Term Exam)

1. Which Communication Paradigm do you think that **Socket programming** belongs to? (Consult slides 9 through 13 in the presentation about Architectural Models: <http://paloalto.unileon.es/ds/lec/DS-2019-Modeling.pdf>)

- Interprocess communication
- Remote invocation
- Indirect communication

2. Does Java offer some form of RPC or remote invocation? Tell its name and provide a brief description of your own.

Java incorporates a distributed computing technology known as Remote Method Invocation (RMI) which allows calling methods exported by remote objects.

3. What is the most used architectural style in the field of Distributed Systems?

The Client/Server architecture is the most used in DS

4. Can a server in a C/S DS act as a client to another server? Describe a concrete, real situation where these chained servers are present.

In real-world distributed systems, it's usual that a server connects with another server for consuming some of its services. In this case, the server becomes the client of the other server. This is the case of an Apache web server that becomes the client of a DNS server when resolving some URL's.

5. Briefly comment about the applicability of TCP and UDP transports in the construction of Distributed Systems that follow the Client/Server and the Peer-to-peer architectures?

The Client/Service model can be implemented over TCP and over UDP. If the developer chooses UDP for implementing their C/S system, then, he or she will have to confer reliability to the messages sent over UDP, since this protocol is not reliable. The messages of the C/S protocol can be equally encapsulated into UDP or into TCP.

In peer-to-peer architectures, where users share some resources with the user of the peer-to-peer network, the messages interchanged by any two peers when the distributed service is functioning constitute a C/S

protocol which, like in the preceding case, can be implemented over TCP or over UDP. In summary, either transport can be used, however, we must take into account that UDP is not reliable, consequently, the developer will have to provide reliability in the C/S protocol.

6. Does UDP lend itself to be used for implementing C/S systems

Yes, it does, however, the developer must compensate the lack of reliability in this transport protocol. In many cases, we will opt for TCP.

7. Explain the most important placement strategies used in the DS of today.

The processes that support a distributed system must be managed and run at Internet server systems. The placement strategies represent different ways of assigning those processes to real physical hosts. The four main placement strategies in use at present are the following:

- Caching (Web caching)
- Mobile code (Java Applets)
- Mobile agents (The application of artificial intelligence to programs that can migrate between servers)
- Having multiple servers offer a single service to a variety of users (A high performance compute cluster)

8. Why is crucial that placement be implemented properly in a DS?

Because placement affects the performance and the security of the distributed system

9. How many blocks is an application decomposed into in a three-tier architecture?

A distributed application that follows the 3-tier architecture is broken down into three responsibilities:

- Presentation
- Business logic
- Persistence

10. Regarding the preceding question, tell how many servers are needed for implementing a 3-tier architecture

Each of the tiers comprising a 3-tier application must be placed (Recall placement strategies, above) in one server, but, that server can run other applications or other tiers of this application. The whole 3-tier application can be run in a single server or, any of the constituent tiers can be run in a separate (physical) server. Furthermore, if the server dedicated to the presentation tier needed to provide a very high performance, it could be implemented by using a web load balancer.

11. In a three-tier architecture, what is the database responsibility?

The database (Database Management System, DBMS) is responsible for saving the data in a persistent storage and making sure that access to it offers a high performance, a high flexibility, high reliability and high availability.

12. (*) Today's web software offers the possibility that a client requests the update of even a small part of a web page? Read a little about the core technology enabling this functionality and which name is XMLHttpRequest.
13. (*) What is the main drawback of the thin-client architecture?
14. (*) What is the *reflection* pattern in distributed systems?
15. Distributed systems are composed of multiple *interacting processes*, in this context, what is a distributed algorithm?

A distributed algorithm, instead of running in a single thread, runs in several threads, normally in several physical servers. The various threads, each running some sections of the distributed algorithm, communicate with the other threads by sending and receiving messages, only.

16. The Network Time Protocol (NTP) can be used to synchronize computer clocks. Explain why, even with this service, no guaranteed lowest bound is given for the difference between two clocks attained after synchronization.

The reason stems from the fact that clock synchronization is based on the requester fetching the current time of the responder in an internet message. This message is delayed and the delayed (The return path delay) is not known, since the requester can only measure the Rtt, not the one-way delay (The return). The offset the requester has to apply to its clock is only an approximation to the actual return delay and when the Rtt grows, the error grows without bound.

Flaviu Cristian invented an algorithm that offsets this limitation of internet clock synchronization by sending repetitive time synchronization requests and by averaging them.

17. What failures does TCP protect from?

TCP offers protection against omission failures (A data packet is lost or an ACK is lost) and against errors induced in packets as they travel over the internet. In all these cases TCP will retransmit the relevant packets, thereby rendering the failure transparent to the application receiving the data, which will have to do nothing to successfully receive a clean, error-free copy of the data.

18. What is a Byzantine failure in a DS?

In Distributed Systems, failures that offer no guarantee regarding their extent are termed Byzantine failures. For example, a component of a Distributed System that fails some of the time and that the rest of time functions more or less correctly, corresponds to the profile of Byzantine Failure.

19. What is a secure channel in DS security? Tell an example protocol that sets up a secure channel before proceeding to data exchange in web protocols.

TLS (Transport Layer Security) and SSL (Secure Sockets Layer) protocols are used by web browsers when

connecting with web servers for creating secure channels before data exchange begins so the data sent and received remain private.

20. Tell an example of an element of a Distributed System that belongs to the Physical Model.

The computer systems where host processes are run (Internet hosts)

21. What DS transparencies apply to Mobile Computing?

Depending on the specific mobile technology considered, we might find the mobility, access, location transparencies.

22. What are the entities communicating in a Distributed System?

From the systems standpoint, the entities communicating in a distributed systems are the threads running at the internet hosts that comprise the distributed system

23. What aspects of the computers and networks most limit the performance of a C/S system?

Regarding networks, the aspect of networks that is more difficult to deal with is *delay*.

24. Same as preceding question, but applied to the Peer-to-peer architecture.

Delay remains important in this case also, but it's bandwidth that has to be maximized since the peers will have to sustain a large number of simultaneous communication flows with the rest of peers, and this entails a high communication bandwidth.

25. What DS transparencies apply to a Web Load Balancer like that in the pdf slide titled "Placement/Replicating servers: Load Balancers"?

- Performance: As the load offered to the load balancer, the performance offered is kept within reasonable limits for even large loads
- Failure: The load balancer, depending on its concrete features, can isolate the client from some failures that might happen in the web servers
- Location: The fact that a request is served by a specific node behind the load balancer or by any other, is irrelevant
- Replication: The fact that the resources served by the load balancer are duplicated remains hidden to the client
- Concurrency: The fact that the requests can be resolved concurrently, remains hidden to the client
- communication

26. Explain the functions of an Http Proxy.

An http proxy offers a consistent interface to its clients that offers access to a variety of services which include the security, the privacy and flexibility in routing http requests.

27. The name of the protocol on which Java RMI is based is RMI-IIOP. Describe that protocol briefly (<https://www.oracle.com/java/technologies/java-rmi-over-iiop.html>) and build the full protocol stack to Java RMI.

Do this exercise on your own

28. Contrast RMI-IIOP with the simpler Java RMI JRMP protocol.

Do this exercise on your own

29. What is a Web Service? Contrast Web Services to Java RMI.

Do this exercise on your own

30. What are the most significant transparencies offered by Java RMI?

Do this exercise on your own