Distributed Systems Exam Questions and Answers

Mihai Budurean

February 2024

Question 10 and question 33 are the same.

1. Explain the difference between a peer group and a hierarchical group in a group communication.

   Answer:

   In a **peer group**, all processes are equal, decisions are made collectively. A peer group is symmetric and has no point of failure. If one of the processes crashes, the group is smaller but otherwise, it is not affected. Decision-making is more complicated and a vote has to be taken.

   In a **hierarchical group**, there is one coordinator process and worker processes. The coordinator makes decisions, delegating a worker for a specific task. Decision-making is easier and faster, compared to a peer group, but if the coordinator crashes, the entire group is brought to a grinding halt.

2. Which topic is related to Cloud service provisioning?

   a. Spot instances
   b. Monitoring of resource usage
   c. On-demand resources
   d. Advance reservation of resources
   e. Task scheduling

3. Give a concrete example of a distributed system functionality to explain following concept: Replication transparency.

   Answer:

   A distributed system often employs data replication to ensure a fast response from databases and to enable the system to be resilient to hardware errors. Replication transparency is the term used to describe the fact that the user should be unaware that data is replicated.

   Take for example, Bitcoin. Bitcoin is a distributed network of computers which stores currency transaction and minting actions across a network of untrusted peers.

4. Comment the following key benefit of using Web services: Interoperability in a heterogeneous environment.

   Answer:

   A key benefit of the **web services** is that **interoperability** has the highest priority. Web service interoperability objectives are to allow programming applications to communicate with each other and exchange data or services among themselves.

   Web services can achieve interoperability in a heterogeneous system due to: platform independence, scalability, and flexibility. Web services are based on open standards, such as SOAP, REST, or XML which enable communication between systems regardless of the underlying operating systems, hardware platforms, or programming languages used. Web services can be scaled horizontally and vertically to accommodate changing workloads and user demands. This scalability ensures that systems can adapt to evolving requirements and handle increased traffic without sacrificing interoperability.

5. Consider an array of 5 string elements with given values. Write it in SOAP encoding style.

   Answer:

```
<soap:Envelope xmlns:soap="http://www.w3.org
   ↪ /2003/05/soap-envelope">
    <soap:Body>
        <m:StringArray xmlns:m="http://www.
           ↪ example.com/strings">
            <m:item>valoare1</m:item>
            <m:item>valoare2</m:item>
            <m:item>valoare3</m:item>
            <m:item>valoare4</m:item>
            <m:item>valoare5</m:item>
        </m:StringArray>
    </soap:Body>
</soap:Envelope>
```

6. Explain the following advantages of using CORBA instead client-server model: (1) dynamic invocation interface; (2) location transparency.

   Answer:

   CORBA facilitates dynamic invocation interfaces, enabling clients to invoke remote methods on objects at runtime without requiring compile-time knowledge of the object's interface or implementation details. This capability relies on CORBA's Interface Repository (IR), which stores interface definitions for all accessible objects. Clients can dynamically query the IR to discover available interfaces and methods, allowing for flexible

and extensible interaction with distributed objects across heterogeneous systems and languages.

CORBA offers location transparency, shielding clients from the physical location and distribution details of remote objects. Clients interact with remote objects through object references managed by the Object Request Broker (ORB). Object references encapsulate the necessary information for locating and communicating with remote objects, such as network address and communication protocol. This abstraction enables clients to interact with remote objects uniformly, regardless of their actual location or distribution in the distributed system, simplifying development and deployment processes.

7. Give two examples that show the difference between a transient fault and an intermittent fault.

   Answer:

   A **transient fault** occurs once and then disappears.

   An **intermittent fault** occurs, then vanishes on its own accord, then reappears, and so on.

   One example that can show the difference between the two types of faults is **the root cause**. Transient faults are often caused by temporary conditions like network congestion or resource shortages. Intermittent faults are caused by more persistent issues, such as hardware degradation or software bugs.

   Another example is **troubleshooting complexity**. Intermittent faults tend to be more challenging to diagnose and troubleshoot compared to transient faults due to their longer duration and unpredictable nature.

8. Explain the relationship between Grids and P2P networks.

   Answer:

   Grids and peer-to-peer (P2P) networks share similarities in their decentralized nature and resource-sharing principles. Both architectures involve the distribution of computing resources among participants, with Grids focusing on collaborative computing across administrative domains and P2P networks emphasizing direct sharing among peers without centralized control. They exhibit dynamic behavior, allowing nodes to join or leave the network seamlessly. While Grids may have some level of centralization for coordination, P2P networks operate in a fully decentralized manner. Despite their differences, both Grids and P2P networks can utilize overlay networks and hybrid architectures to enhance scalability, resilience, and resource utilization in distributed computing environments.

9. Explain the differences between application virtualization and storage virtualization.

   Answer:

Application virtualization describes the process of compiling applications into machine-independent byte code.

Storage virtualization is the logical abstraction of physical storage.

The first difference between them is the purpose. Application virtualization decouples software applications from the underlying operating system and hardware. Storage virtualization abstracts the physical storage resources from the applications and provides a logical layer of storage management.

The second difference is the scope. As it is specified in the names, one has application scope and the other one has storage as the scope.

The last difference is the method of virtualization. Application virtualization focuses on encapsulating different applications, while storage virtualization only rearranges the storage.

10. Consider the case of a distributed application with only two processes that are need to update some data by entering a critical section. Compare the centralized algorithm with the token ring algorithm for mutual exclusion: how they are different, how many messages are sent, what happens in at the crash of one process. Which you suggest to be the better in this case?

11. What is a skeleton?

    Answer:

    The **skeleton** is the server-side object participating in distributed object communication. It acts as a gateway for server-side objects and all incoming client requests are routed through it.

12. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: device sharing.

    Answer:

    One practical example of device sharing in a distributed system is seen in collaborative document editing platforms like Google Docs or Microsoft Office Online. In these systems, multiple users can simultaneously edit a document hosted in the cloud, with changes being synchronized in real-time across all connected devices. This distributed approach enables seamless collaboration among users regardless of their physical location or the devices they use. Users can edit documents collaboratively from different devices such as desktop computers, laptops, tablets, or smartphones, with changes instantly propagated to all connected users. This shared access to documents enhances productivity and facilitates teamwork, allowing users to collaborate efficiently without the constraints of location or device compatibility.

13. Enumerate the types of communication models used in SOAP. Explain the differences between them.

Answer:

SOAP supports the following two types of communication models: **SOAP RPC** and **SOAP Messaging**.

SOAP RPC defines a remote procedural call-based synchronous communication where the nodes send and receive messages using requests and responses.

SOAP Messaging defines a document-driven communication where nodes send and receive XML-based documents using synchronous and asynchronous messaging.

14. Give two examples that show the difference between a permanent fault and an intermittent fault.

    A **permanent fault** continues to exist until the faulty component is repaired.

    An **intermittent fault** occurs, then vanishes on its own accord, then reappears, and so on.

    One example that can show the difference between the two types of faults is **the duration and persistence**. Permanent faults are persistent and remain until addressed. Intermittent faults may occur sporadically and disappear on their own, only to reappear later.

    Another example is **predictability and complexity**. Permanent faults are easier to detect, caused of their consistent occurrence. Intermittent faults are more challenging in terms of detection because their occurrence is hard to predict and is not as consistent as permanent ones.

15. Explain the concept of in-network data aggregation.

    Answer:

    In-network data aggregation is an effective technique to reduce communications costs by eliminating the inherent redundancy in raw data collected from the sensors. In-network aggregation routing protocols are divided into the following approaches: tree-based, cluster-based, multipath, and hybrid.

16. Resources can be provisioned in Public Clouds

    a. based on contractual terms

    b. based on resource exchanges

    c. based on bids for free resources

    d. based on user location

17. Which statements are true?

    a. A special compiler reads the client specification and generates the client stub

b. A special compiler reads the server specification and generates the client stub

c. A special compiler reads the server specification and generates the server stub

d. Stubs are usually generated automatically

e. A special compiler reads the client specification and generates the server stub

18. What is a standard? What is an open standard? What is a middleware standard?

Answer:

A **standard** stipulates the specification of a product. It serves as a blueprint according to which different products can be produced. A specification identifies the verifiable characteristics of a system.

An **open standard** represents a standard that is not subject to any commercial interest. This kind of system is nonproprietary, freely available, technology-independent. An open standard has a democratic creation process and can appear only if there is product availability.

A **middleware standard** has the purpose of establishing the interfaces between different components to enable their interaction with one another. The interfaces can be horizontal or vertical. Horizontal interfaces exist between an application and the middleware, define how an application can access the functionality of the middleware and it is also referred to as an **Application Programming Interface (API)**. A vertical interface is between two instances of a middleware and it helps with the interoperability between applications.

19. Explain the difference between the selective broadcast and adaptive broadcast in P2P systems.

Answer:

In peer-to-peer (P2P) systems, selective broadcasting and adaptive broadcasting are two different methods of transmitting information between nodes.

Selective broadcasting involves manual selection of information destinations. Nodes carefully select their communication partners and transmit information only to them. This approach is based on a centralised structure, where one node is responsible for selecting destinations and transmitting information to them.

Adaptive broadcasting, on the other hand, is based on automatic adaptation of the information destination according to the state of the network. Nodes can change their destinations depending on network congestion or other problems. This type of broadcasting is based on a decentralised system, where each node is responsible for choosing the destination and

transmitting the information. In general, adaptive broadcasting can be more flexible and efficient in terms of network usage, but it can be more difficult to implement and may require more complexity in terms of information processing.

20. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: reliability.

    Answer:

    A concrete example of distributed system functionality that hides heterogeneity in reliability can be the implementation of an active replication solution. In this case, the middleware can manage multiple components of the distributed system running on different operating systems or different hardware.

    To manage system reliability, the middleware can use active replication mechanisms, such as Mirroring or Master-Slave Replication, so that it can redirect requests to alternative components if a component fails or is no longer available. The middleware can also implement mechanisms to check the status of components so that it can automatically detect problems and redirect requests to alternative components.

    Through these mechanisms, middleware can hide the heterogeneity of distributed systems and ensure a high degree of reliability without requiring specific knowledge about individual system components.

21. Explain the difference between the ostrich algorithm and the avoidance approach used in handling deadlocks.

    Answer:

    The **Ostrich algorithm** is based on the idea of preemptive resource allocation. This means that when a process requests a resource, the operating system checks to see if the resource is available.

    If the resources are available, the operating system grants the resources to the process. If it is not available, the operating system puts the process in a wait state until the resources become available.

    In addition, if a process requests a resource that is not available, the operating system simply ignores the deadlock and continues to allocate resources to other processes. This means that the system can continue to function even in the presence of deadlocks.

    While the Ostrich algorithm is a popular choice in distributed systems, **avoidance strategy** cannot be used in these systems. The main problem with this approach is that algorithms need to know in advance how much of each resource every process will eventually need.

22. Web services are supporting

a. a loosely coupled application components

b. machine to machine interaction

c. human to application interactions

d. tightly coupled application components

23. Explain the differences between MOM, CORBA, RMI and DCOM.

Answer:

MOM (Message Oriented Middleware), CORBA (Common Object Request Broker Architecture), RMI (Remote Method Invocation), and **DCOM (Distributed Component Object Model)** are all technologies used to interconnect applications in distributed systems.

MOM focuses on the transmission of messages between applications, allowing asynchronous and decoupled communication.

CORBA is an industry standard that provides a distributed object model and infrastructure for interconnecting applications in different programming environments.

RMI provides the possibility of invoking remote methods between Java objects, thus making it possible to interconnect Java applications.

DCOM, developed by Microsoft, is a technology that provides an infrastructure for networking software components based on Windows protocols.

24. Explain the difference between the detection approach and the prevention approach used in handling deadlocks.

Answer:

Detection approach lets deadlocks occur, finds them, and tries to recover to a safe state using recovery techniques. It is a popular approach, after the Ostrich algorithm, because prevention and avoidance are hard to execute properly.

Deadlocks Prevention describes how to avoid deadlocks by allocating resources carefully.

The difference between the two approaches is one focuses on what to do after the deadlock occurs, and the other one focuses on what to do before the deadlock occurs.

25. Consider the case of a distributed system of a bank in which messages can be lost. In the case of money redraw request what quality requirements are allowed in the services to be build?

a. Exactly once

b. Maybe

c. At-most-once

d. At-least-once

26. What problem of distributed systems is modelled with the Byzantine generals problem?

    Answer:

    The **Byzantine generals** problem is a well-known concept in distributed computing that describes the difficulty of coordinating the actions of several independent parties in a distributed system.

27. Resource discovery is a topic of

    a. Resource provisioning

    b. Resource allocation

    c. Resource monitoring

28. Comment the following key benefit of using Web services: Integration with existing systems.

    Answer:

    One key benefit of using web services is their seamless integration with existing systems. Web services provide standardized protocols and interfaces for communication, allowing disparate systems and applications to interact and exchange data efficiently. By leveraging web services, organizations can integrate new functionalities and services into their existing systems without the need for significant changes or redevelopment. This integration capability enables organizations to leverage their existing investments in technology while extending the functionality and interoperability of their systems to meet evolving business requirements.

29. Explain the advantages of using dynamic bindings by a concrete example of a distributed system functionality.

    Answer:

    Advantages of usage of **dynamic bindings** in distributed systems:

    - Can handle multiple servers that support the same interface.
    - The binder can spread the clients randomly over the servers to even load.
    - The binder can poll the servers periodically, automatically deregistering any server that fails to respond, to achieve a degree of fault tolerance.
    - The binder can also assist in authentication.
    - The binder can also verify that both client and server are using the same version of the interface.

    These advantages of dynamic binding in distributed systems are closely associated with the functionality of **service discovery** and **dynamic load balancing**.

Service discovery and dynamic load balancing allow clients to locate and connect to available servers in a distributed system efficiently. Dynamic binding facilitates this process by enabling clients to bind to servers dynamically at runtime.

30. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: economics.

    Answer:

    A practical example of **economics** in distributed systems can be the use of **cloud computing technology**. Instead of buying its hardware resources, an organization can lease or use hardware resources from the cloud, which can be more cost-effective as it no longer has to pay for the purchase and maintenance of equipment. Also, if more or less storage or processing power is needed, the organization can easily increase or decrease access to resources without investing in additional hardware.

31. Consider the case of the real-time distributed software system assisting an underground network. Explain the relationship between (1) the fact that the implicit color of the signals is controlled by the system is red and only when a train is near to start a green is allowed, and (2) the fail-safe condition of the real-time system.

    Answer:

    In a real-time distributed software system that assists an underground railway network, the default color of signals controlled by the system is red. This red signal indicates that the train should stop and not proceed. Only when a train is approaching and about to start, the system allows a green signal, which means that the train can proceed.

    The relationship between the red signal and the real-time emergency state of the system is critical. The real-time nature of the system means that the system must respond quickly to changing conditions and respond to unexpected events, such as a train approaching too quickly. If the system fails to respond in real time, it could result in a train collision or other hazardous situations.

    In this scenario, the red signal acts as a safety mechanism to ensure that the system is functioning properly and that trains are not proceeding unless it is safe to do so. The green signal, on the other hand, indicates that the system has determined that it is safe for the train to proceed, based on its real-time monitoring of the railway network.

32. Explain the difference between storage virtualization and operating system virtualization.

    Answer:

Storage virtualization is the logical abstraction of physical storage. This abstraction helps to combine physical disk drives into pools of available storage. It enables great amounts of physical storage to be available to many individual systems.

Operating system virtualization describes various implementations of running multiple, logically distinct system environments on a single instance of an operating system kernel. Operating within the confines of their root directories and associated filesystem.

33. Consider the case of a distributed application with only two processes that are need to update some data by entering a critical section. Compare the centralized algorithm with the token ring algorithm for mutual exclusion: how they are different, how many messages are sent, what happens in at the crash of one process. Which you suggest to be the better in this case?

    Answer:

    The centralized algorithm requires a coordinating process that controls access to the critical section. Processes request access to the coordinator, and the coordinator decides which process has access at any given time. This can be done by sending a message to the coordinator and waiting for a response. This can result in a single point of lock if the coordinator cannot grant access to the critical section or if the coordinator is locked out.

    The token ring algorithm involves the use of a virtual token circulating among processes. The process holding the token has access to the critical section. The process that has completed access to the critical section passes the token to the next process. This can be done by sending a message to the next process. The token ring algorithm requires more messages than the centralized algorithm, but there is no single locking point.

    In our case, where there are only two processes, the centralized algorithm would probably be sufficient and easier to implement. Where there are multiple processes or a large number of requests for access to the critical section, the token ring algorithm would probably be more appropriate since there is no single point of lock.

34. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: high performance.

    Answer:

    An example of a high-performance distributed system is the Hadoop distributed storage system. Hadoop is an open-source system that can process large amounts of data efficiently and quickly. It works by distributing data across a network of nodes and processing it in parallel on each node, which enables high performance and shortens the time it takes to process

data. For example, if you want to analyze millions of data records, with a centralized system this could take hours or even days, whereas with Hadoop, this can be done in minutes or even seconds.

An example of a high-performance distributed system is the Bitcoin mining network. It uses blockchain technology and allows users around the world to participate in validating Bitcoin transactions through a distributed infrastructure. Each node in the network is responsible for processing and validating part of the transactions, which allows the system to handle a large volume of transactions in a short time and with high performance.

35. Name and discuss a concrete example of P2P application for instant messaging.

    Answer:

    A good concrete example of P2P application for instant messaging is BitTorrent, which allows its users to share files without the existence of a central server.

    For example, when a user downloads a file, he receives pieces of the file from different users at the same time. This permits a faster download and a more efficient use of band length.

    BitTorrent has a feature of instant messaging that allows the users to start a communication about the shared files, and to offer technical support during the file transfer.

36. Explain how ubicomp devices can form an adhoc network.

    Answer:

    Ubicomp is called also **pervasive computing**. The vision of this concept is to create a world of ubiquitous computers that become invisible by being embedded into the physical environment to support people unobtrusively in fulfilling their tasks.

    Ubicomp devices can form an ad-hoc network by using highly specialized wireless computing devices embedded into our physical environment. These devices can perceive and control certain parameters of their physical environment and they can communicate with each other.

37. What is an orphaned invocation?

    Answer:

    Orphan invocation is a client process that breaks down during the execution of a RPC (remote procedure call).

38. Explain why the Time Division Multiple Access (TDMA) protocol is used in real-time systems.

    Answer:

Time Divison Multiple Access (TDMA) protocol is a channel access method for shared-medium networks. It allows several users to share the same frequency channel by dividing the signal into different time slots. The users transmit in rapid succession, one after the other, each using its time slot.

This technique is preferable in real-time systems because it provides tight control over access to communication resources. For example, in industrial control applications, where response times and stability are critical, TDMA allows efficient use of radio spectrum and even frequency access distribution. TDMA can also be used to limit interruptions and avoid collisions between multiple transmissions by separating transmissions into specific time slots.

39. Why is it difficult to keep a synchronized system of physical clocks in distributed systems?

    Answer:

    It is difficult to keep a synchronized system of physical clocks in distributed systems because the different nodes maintain their own time using local clocks and their time values may not be the same for different nodes

40. Explain what pervasive computing means.

    Answer:

    Pervasive computing, also called ubiquitous computing, is the growing trend of embedding computational capability (generally in the form of microprocessors) into everyday objects to make them effectively communicate and perform useful tasks in a way that minimizes the end user's need to interact with computers as computers. Pervasive computing devices are network-connected and constantly available.

41. Give a concrete example of a distributed system functionality to explain the following concept: Failure transparency.

    Answer:

    Failure transparency in distributed systems ensures that users and applications are shielded from the effects of component failures within the system. A concrete example is found in cloud computing platforms like Amazon Web Services (AWS) or Microsoft Azure, where virtual machines (VMs) are hosted across multiple physical servers. If a physical server hosting a VM fails due to hardware issues or maintenance, the cloud platform automatically migrates the VM to a healthy server without interrupting service. Users and applications accessing the VM experience no disruption or awareness of the underlying failure. This failure transparency ensures high availability and reliability in cloud-based services, as users can continue operations without being impacted by individual server failures.

42. Consider a distributed application with four processes with the identifiers 1, 2, 3 and 4. The coordinator, process 4 crashes and process 1 identify this situation and starts an election. In the case of a Bully algorithm how many messages are send in the system until the coordinator is know? And in case of Ring algorithm?

    Answer:

    In the case of the Bully algorithm:

    1. Process 1 detects the failure of the coordinator (Process 4) and starts an election.
    2. Process 1 sends election messages to processes with higher identifiers (2 and 3).
    3. Process 2 responds to Process 1 if it has a higher identifier.
    4. Process 3 responds to Process 1 if it has a higher identifier.
    5. Process 1 becomes the coordinator if no higher identifier processes respond.

    In this scenario, the total number of messages sent in the system until the coordinator is known depends on the number of processes with higher identifiers than the failed coordinator. In this case, two messages are sent (1 from Process 1 to Process 2, and 1 from Process 1 to Process 3).

    In the case of the Ring algorithm:

    1. Process 1 detects the failure of the coordinator (Process 4) and starts passing an election message clockwise.
    2. Each process receiving the election message checks its identifier and forwards the message if it has a higher identifier.
    3. The election message continues to circulate until it reaches the highest identifier process, which becomes the new coordinator.

    In this scenario, the total number of messages sent in the system until the coordinator is known depends on the number of processes in the ring. Each process forwards the message until it reaches the highest identifier process, which results in a total of three messages being sent (one per process in the ring).

43. Comment the differences between the SOA and REST types of a Web service architecture.

    Answer:

    SOA (Service-Oriented Architecture) and REST (Representational State Transfer) are two different types of web service architectures, each with specific advantages and disadvantages.

SOA is based on the client-server model and focuses on defining and using services formally so that it is possible to integrate them into multiple applications. In this way, services can be used by multiple applications or components, reducing redundancy and improving system flexibility. SOA uses protocols such as XML and SOAP for communication between components, as well as other techniques such as service contracts and service registries to manage and discover available services.

REST, on the other hand, focuses on using HTTP as a communication protocol and is based on principles such as recursion, hypermedia and statelessness. Instead of defining services formally, REST is based on resource usage, where an application can access and manipulate these resources via HTTP. REST does not use a specific protocol such as SOAP, thus allowing a wider set of technologies and protocols to be used for implementation.

In conclusion, both SOA and REST have specific advantages depending on the needs and characteristics of the system, so the choice between them should be made according to the specific context and requirements of the project.

44. What is a purpose of an XSD document?

    Answer:

    An XML schema definition **XSD** is a framework document that defines the rules and constraints for an XML document. An XSD describes the elements in an XML document and can be used to validate the contents of the XML document.

45. Explain the following advantages of using CORBA instead client-server model: (1) programming-language independence (2) network transparency.

    Answer

    CORBA enables programming language independence by providing a standardized interface definition language (IDL) that allows developers to describe interfaces and data types in a language-neutral manner. Clients and servers can be implemented in different programming languages as long as they adhere to the CORBA IDL specifications. This flexibility allows organizations to leverage existing expertise and legacy systems written in various programming languages without requiring extensive modifications or rewrites. Additionally, CORBA's language-independent nature promotes interoperability and code reusability across diverse environments, facilitating the integration of distributed components and systems.

    CORBA achieves network transparency by abstracting the underlying network communication details from application developers and users. The CORBA infrastructure, including the Object Request Broker (ORB), handles network communication transparently, shielding developers from low-level networking complexities such as socket programming and protocol

implementation. Clients and servers interact with remote objects using high-level method invocations, regardless of their physical location or network configuration. This transparency simplifies distributed system development and maintenance, allowing developers to focus on business logic and application functionality rather than intricate network intricacies.

46. Explain the role of the client stub in remote procedure call.

47. Explain the difference between the P2P model and the client-server model.

48. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: extensibility.

49. Middleware refers to ...:

    a. software

    b. neither software, neither hardware

    c. hardware

50. A cloudlet serves mobile or smart devices in

    a. Fog computing

    b. Cloud computing

    c. Edge computing

    d. Cluster computing

51. Give three concrete examples of applications of ubicomp.

52. Explain the differences between application virtualization and operating system virtualization.

53. Explain the difference between the server driven and client driven algorithm to locate idle workstations.

54. Explain the difference between the multicasting and broadcasting techniques.

55. Tanenbaum's definition of a distributed system is stating that it is a collection of independent computers that appear to the users of the system as a single computer. Please develop the single system image concept by two examples.

56. Explain the difference between a close group and an open group in a group communication.

57. Give a concrete example of a distributed system functionality that explain how the middleware hides the heterogeneity at the level of: computer architectures.

58. Explain the differences between application virtualization and desktop virtualization.

59. A remote procedure call requires

    a. an asynchronous communication
    b. a request-oriented communication
    c. a synchronous communication
    d. a message-oriented communication

60. Explain the notion of virtual organization used in Grids.

61. What is specific for Edge and Fog compared with Cloud?

    a. Use of mobile resources
    b. Network traffic
    c. Location awareness

62. Explain the difference between message-oriented communication and request-oriented communication.

63. Comment the following key benefit of using Web services: Freedom of choice.

64. Explain the differences between network virtualization and storage virtualization.

65. Give a concrete example of a distributed system functionality to explain the following concept: Access transparency.

66. Which device can be a node in Edge computing environment?

    a. Sensor
    b. Router
    c. Car
    d. Network switch

67. Discovering resources in P2P systems

    a. is associated with the notion of servent
    b. is done through broadcasting
    c. is done using resource indexing
    d. is associated with the notion of presence

68. Consider the case of a distributed application with only two processes that are need to update some data by entering a critical section. Compare the distributed algorithm with the token ring algorithm for mutual exclusion: how they are different, how many messages are sent, what happens in at the crash of one process. Which you suggest to be the better in this case?

69. Explain the difference between the simple broadcast and selective broadcast in P2P systems.

70. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: reuse.

71. Describe the difference between the notions of ports associated with Java sockets and WSDL.

72. Explain the differences between network virtualization and machine virtualization.

73. Explain the difference between the multicasting and unicasting techniques.

74. The computational grid

    a. is included in an electrical grid
    b. is equivalent with an electrical grid
    c. received the name from an analogy with the electrical grid
    d. is based on an electrical grid

75. Give a concrete example of a component fault in a distributed system and its effect.

76. Which Cloud resource management topic deals with the selection of an optimal set of physical machines to host virtual machines necessary for an application?

77. Explain the relationship between Grids and Internet computing.

78. The amount of network capacity consumed is a metric for:

    a. reliability
    b. scalability
    c. transparency
    d. flexibility
    e. performance

79. Consider a distributed system based on 5 sensors and each of those has associated a process that ensure active replication. Which is the degree of fault tolerance in the case when the sensors are expected to exhibit Byzantine failures?

80. Consider a vector of 4 integer elements with given values. Write it in SOAP encoding style.

81. Consider the case of a distributed application with only two processes that are need to update some data by entering a critical section. Compare the distributed algorithm with the centralized algorithm for mutual exclusion: how they are different, how many messages are sent, what happens in at the crash of one process. Which you suggest to be the better in this case?

82. Explain the role of rmiregistry.

83. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: inherent distribution.

84. Explain the difference between a blocking and a non-blocking receiving primitive.

85. Consider the case of a distributed system of a bank in which messages can be lost. In the case of account check what quality requirements are allowed in the services to be build?

    a. Exactly once
    b. Maybe
    c. At-most-once
    d. At-least-once

86. Explain the role of the server stub in remote procedure call.

87. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: scalability.

88. Explain the relationship between Grids and clusters.

89. Consider a distributed system based on 3 sensors and each of those has associated a process that ensure active replication. Which is the degree of fault tolerance in the case when the sensors are expected to exhibit Byzantine failures?

90. Consider the case of a distributed system of a bank in which server breakdown can be encountered. In the case of account check what quality requirements are allowed in the services to be build?

    a. Exactly once
    b. Maybe
    c. At-most-once
    d. At-least-once

91. Give an example proving the fact that SOA is not resumed only to Web services.

92. Explain how a UDDI register can be used.

93. Explain the role of the proxies to ensure transparency.

94. Comment the differences between the RPC and REST types of a Web service architecture.

95. Explain the difference between a data-centric communication used in ubicomp and a computer-centric (network address-based) co-mingling used in the client-server model.

96. Give a concrete example of a distributed system functionality to explain the following concept: Technology transparency.

97. An edge device is:

    a. A resource for local processing in user context
    b. Exclusively a network resource
    c. Exclusively a mobile resource
    d. A resource located at the end of the network

98. Explain the difference between a synchronous system and an asynchronous system.

99. Globus is a toolkit for:

    a. Edge computing
    b. Cloud computing
    c. Grid computing
    d. Global computing

100. What is clock synchronization?

101. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: Higher productivity and shorter development cycle.

102. Give a concrete example of a distributed system functionality that explain how the middleware hides the heterogeneity at the level of: network.

103. Comment the following key benefit of using Web services: Programming productivity.

104. Consider a distributed system based on 2 sensors and each of those has associated a process that ensure active replication. Which is the degree of fault tolerance in the case when the sensors are expected to have fail-silent failures?

105. Comment the following key benefit of using Web services: Supporting multiple types of clients.

106. Explain the differences between application virtualization and machine virtualization.

107. Explain the utility of a virtual namespace.

108. Give a concrete example of a distributed system functionality to explain the following concept: Location transparency.

109. Explain the concept of single connection used in Grid.

     Consider a distributed system based on 2 sensors and each of those has associated a process that ensure active replication. Which is the degree of fault tolerance in the case when the sensors are expected to exhibit Byzantine failures?

110. Give two examples that show the difference between a fail-silent fault and a Byzantine fault.

111. Explain the difference between SOAP and Inter-ORB protocol.

112. Comment the differences between the RPC and SOA types of a Web service architecture.

113. Explain the difference between network virtualization and operating system virtualization.

114. Give a concrete example of a distributed system functionality to explain the following concept: Operating systems.

115. Explain the difference between the centralized and decentralized indices in P2P systems.

116. Give an example when Berkley's algorithm for time synchronization is preferable to Cristian's algorithm.

117. Give two examples of middleware tools for wireless sensor networks.

118. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: Reduced cost.

119. Explain the role of the register in SOA architecture.

120. Explain the concept of leasing in ubicomp.

121. Give a concrete example of a distributed system functionality to explain the following concept: Programming language.

122. Explain the difference between a synchronous and an asynchronous forwarding primitive.

123. Why do synchronization techniques used for processes running one computer not apply to distributed systems?

124. Is SOA useful for loosely coupled systems or tightly coupled systems? Explain your answer.

125. Name two advantages and two disadvantages of distributed versus centralised systems.

126. Consider a vector of 3 floating point elements with given values. Write it in SOAP encoding style.

127. Give an example when Cristian's algorithm for time synchronization is preferable to the Berkley algorithm.

128. Give two examples to illustrate the difference between a soft real-time distributed system and a hard real-time distributed system.

129. Consider a distributed application with three processes with identifiers 1,2,3. The coordinator, process 3 crashes and process 1 identifies this situation and starts the election. In the case of a Bully algorithm how many messages are sent to the system until the new coordinator is known? And in the case of the Ring algorithm?

130. Explain the difference between a synchronous and an asynchronous communication.

131. Give a concrete example of the functionality of a distributed system to explain the following concept: Parallelism transparency.

132. What is the web enhancement offered by web services?

133. Give a practical example (a concrete distributed system other than in the textbook) for the following advantage of a distributed system over other kinds of systems: collaboration.

134. Name and discuss a concrete example of a P2P application that manages and shares information.