

FINAL EXAM SOLUTION 2014

1- Define distributed Systems, state examples?

Distributed system is one in which hardware or software components located at networked computers communicate and coordinate their actions only by passing messages .

Examples : Web search , Multiplayer online games , financial trading system

2- List characteristics of distributed system . what are distributed system advantages ?

1- Concurrent system : I can do my work on my computer while you do your work on yours computer , the capacity of system is increased by adding more resources

2- No global clock : every software or hardware is component is autonomous process synchronization , coordinate by message passing

3- Independent failures : each component of the system can fail independently , leaving the others still running .

- Advantages :

1- Resource sharing

2- Communication

3- Speed

4- Economy

5- Reliability

6- Incrementally growth

7- Inherent Distribution

3- Design Challenges of Distributed Systems?

1- Heterogeneity: Heterogeneous components must be able to communicate.

2- Openness: interfaces should allow component to be added or removed.

3- Security: the system should only be used in the way intended.

4- Scalability: System performance should increase with inclusion of additional components

5- Failure handling: Failure in component shouldn't result in system failure.

6- Transparency: Distribution should be hidden from the users as much as possible.

Question 2 : T/F (مش مطلوب التصحيح في الامتحان بس ممكن تغير الجملة)

1- X >> independent

2- √

3- X >> each component of the system can fail independently, leaving the others still running

4- √

5- Hidden

6-Security

7- √

8-MiddleWare

9- √

10- √

11-Client-to-Server

12- √

13-Distributed algorithms

Question 3 :

A) Advantages and Disadvantages ::--

1) Client-to-Server:

Advantages:

- 1- Data management is much easier because the files are in one location.
- 2- Allows fast backups and efficient error management
- 3- There are multiple levels of permissions, which can prevent users from doing damage to files.
- 4- The server hardware is designed to serve requests from clients quickly, and this reduces the amount of network traffic between the server and the client machine, improving network performance.

Disadvantages:

- 1- Client-Server-Systems are very expensive and need a lot of maintenance.
- 2- The server constitutes a single point of failure. If failures on the server occur, it is possible that the system suffers heavy delay or completely breaks down, which can potentially block hundreds of clients from working with their data or their applications. Within companies high costs could accumulate due to server downtime.

2) Peer-to-Peer:

Advantages:

- 1- In a pure Peer-to-Peer architecture there is no single point of failure and that means if one peer breaks down, the rest of the peers are still able to communicate.
- 2- There is better scalability because most peers interact with each other.

3- Peer-to-Peer allows to prevent bottleneck such as traffic overload using a central server Architecture , because Peer-to-Peer can distribute data and balance request across the net without using a central server.

4- Peer-to-Peer provides the opportunity to take advantage of unused resources

Disadvantages:

1- Today many applications need a high security standard, which is not satisfied by current Peer-to-Peer solutions.

2- The connections between the peers are normally not designed for high throughput rates .

3- A centralized system or a Client-Server system will work as long as the service provider keeps it up and running. If peers start to abandon a Peer-to-Peer system, services will not be available to anyone.

4- Most search engines work best when they can search a central database rather than launch a meta search of peers .

3) Thin Client:

Advantages:

a-Lower Administrative Costs

b. Improved Security

c. Lower Hardware Costs

d. Lower Energy Consumption Costs

e. Lower Chance of Theft

f. Fewer Moving Parts Means Decreased Internal Dust and Increased Safety

g. Small Footprint and Lower Bandwidth Requirements

h. Easy Upgrades with Little or No Network Service

i. Easy Distribution : ability to make changes to the application without having to push software to every desktop that uses it.

Disadvantages:

a. Increased Server Requirements

b. Poor Multimedia Application Performance

c. Lack of Compatibility With Some Applications and Need for Improvements in Thin-Client Network Management

Software

d. Resistance to Change

e. Single Point-of-Failure Requires a Full Back-Up Server

B) What are the security mechanisms used in distributed computing ?

- 1- Encryption
- 2- Authentication
- 3- Authorization

Give two examples explained that the security is an important parameter in exchanging data on DS..

- 1- A doctor might request access to hospital patient data or send additions to that data
- 2- In electronic commerce and banking , users send their credit card numbers across the internet .

C) what is meant by internet protocol ?

internet protocol is the principal communications protocol and it has the task of delivering packets from the source host to the destination host , it defines packet structures that encapsulate the data to be delivered .

D) What is the purpose of fundamental model? Explain briefly .

- 1- To make explicit all the relevant assumptions about the systems we are modeling
- 2- To make generalizations concerning what is possible or impossible , given those assumptions .

E) What is the difference between RMI and RPC? What is meant by group communication?

(RMI) Remote Method Invocation : Is similar to RPC but for distributed objects, with added benefits in terms of using object-oriented programming concepts in distributed systems and also extending the concepts of an object reference to the global distributed environments , and allowing the use of object references as parameters in remote invocations.

(RPC) Remote Procedure Call : it represents a major intellectual breakthrough in distributed computing , with the goal of making the programming of distributed systems look similar .