This assignment is based on lecture 12 (chapter 24 – Distributed Databases)

o Submit your *own work* on time. No credit will be given if the assignment is submitted after the due date.
o Note that the completed assignment should be submitted in .doc, .docx, .rtf or .pdf format only.
o In MCQs, if you think that your answer needs more explanation to get credit then please write it down.
o You are encouraged to discuss these questions in the Sakai forum.

(1) Compare and contrast a DDBMS with distributed processing. Under what circumstances would you choose a DDBMS over distributed processing?
ANS:

In Distributed processing, a database's logical processing is shared among two or more physically independent sites that are connected through a network. For example, the data input/output (I/O), data selection and data validation might be performed on one computer, and a report based on that data might create on another computer.

A distributed database on the other hand stores a logically related database over two or more physically independent sites. The sites are connected via a computer network. In contrast, the distributed processing system uses only a single-site database but shares the processing chores among several sites. In a distributed database system, a database is composed of several sites. In a distributed database system, a database is composed of several parts known as database fragments. The database fragments are located at different sites and can be replicated among various sites. Each database fragment is, in turn, managed by its local database process.

Distributed Database is a logically interrelated collection of shared data (and a description of this data), physically distributed over a computer network. Distributed DBMs is a software system that permits the management of the distributed database and makes the distributed database and makes the distribution transparent to users. The DDBMS has the following functionality:
- Extended communication services
- Extended Data Dictionary
- Distributed query processing
- Extended concurrency control
- Extended recovery services.

(2) Compare and contrast a DDBMS with a parallel DBMS. Under what circumstances would you choose a DDBMS over a parallel DBMS?

ANS:
Distributed database has data in two or more server computers at separate locations Where as a typical parallel database resides in one location with one set of files, thought server computers share the workload.

The different circumstance to choose DDBMS and parallel DBMS are: -

Speed: parallel database main advantage is speed.
Access: Distributed databases improve access, as each local office has its own database.
Reliability: In case of parallel database the database becomes unavailable when computer crashes but in case of distributed database it will be available with reduced performance.

(3) Discuss the advantages and disadvantages of a DDBMS.
ANS: The advantages of DDBMS are:
- Extended communication services
- Extended Data Dictionary
- Distributed query processing
- Extended concurrency control
- Extended recovery services.
- Faster response
- Increase reliability and availability

The disadvantages of DDBMS
- Complex software
- Increased processing overhead
- Data integrity
- Different data formats might be used
- Deadlock is difficult to handle compared to a centralized system
- May cause much more network traffic in case of write operation in database.
- Handling failures is a difficult task.

(4) What is the difference between a homogeneous and heterogeneous DDBMS? Under what circumstances would such systems generally arise?
ANS: The difference between a homogeneous and heterogeneous DDBMS are
Homogeneous DDBMS:
- All sites use same DBMS product.
- Much easier to design and manage
- Approach provides incremental growth and allows increased performance

Heterogeneous DDBMS
- Sites may run different DBMS products, with possibly different underlying data models.
- Occurs when sites have implemented their own databases and integration is considered later:

(5) What functionality do you expect in a distributed DBMS?

ANS: The different functionality of DDBMS are:
- Extended communication services
- Extended Data Dictionary
- Distributed query processing
- Extended concurrency control
- Extended recovery services.

(6) One problem area with DDBMSs is that of distributed database design. Discuss the issues that have to be addressed with distributed database design. Discuss how these issues apply to the global system catalog.

ANS: The issues that have to be addressed with distributed database design are:
- Fragmentation: Relation may be divided into a number of sub-relations, which are then distributed.
- Allocation: Each fragment is stored at site with "optimal" distribution.
- Replication: Copy of fragment may be maintained at several sites.

(7) What are the strategic objectives for the definition and allocation of fragments?

ANS: Fragment is use in which applications works with views rather than entire relations. It increases the efficiency in which data is stored close to where it is most frequently used and data that is not needed by local application is not stored.

The strategic objectives for the definition and allocation of fragments are to achieve
- Locality of Reference
- Improved reliability and availability
- Improved performance
- Balanced storage capacities and costs
- Minimal communication costs

(8) Describe alternative schemes for fragmenting a global relation. State how you would check for correctness to ensure that the database does not undergo semantic change during fragmentation.

ANS: The alternative schemes for fragmenting a global relation are allocation and replication. In Allocation each fragment is stored at site with optimal distribution. In order to check for correctness to ensure that the database does not undergo semantic change during fragmentation. There are three rules that applies:
- Completeness: If relation R is decomposed into fragments R1, R2, … ,Rn, each data item that can be found in R must appear in at least one fragment

- Reconstruction: Must be possible to define a relational operation that will reconstruct R from the fragments.
- Disjointness: If data item di appears in fragment Ri, then it should not appear in any other fragment.

(9) What layers of transparency should be provided with a DDBMS? Give examples to illustrate your answer. Justify your answer.
ANS: The layer of transparency provided by DDBMS are
- Distribution transparency: It allows user to perceive database as single, logical entity.
- Naming transparency: It ensure no two sites create a database object with same name
- Transaction transparency: It ensures that all distributed transactions maintain distributed database's integrity and consistency.
- Concurrency Transparency: All transactions must execute independently and be logically consistent with results obtained if transactions executed one at a tie, in some arbitrary serial order.
- Failure transparency: it ensures atomicity and durability of global transaction.

(10) A DDBMS must ensure that no two sites create a database object with the same name. One solution to this problem is to create a central name server. What are the disadvantages with this approach? Propose an alternative approach that overcomes these disadvantages.
ANS: The disadvantages to create central name server are
- Loss of some local autonomy.
- Central site may become a bottleneck;
- Low availability; if the central site fails remaining sites cannot create any new objects.
  The alternative approach could be transaction transparency and Concurrency transparency.