Chapter 10: Distributed Databases

1. Distributed Database Management Systems (DDBMS)

- What are the key components of a Distributed Database Management System (DDBMS)?

- Answer : The key components of a DDBMS include:

- Distributed Database : A collection of multiple databases distributed across different locations but logically connected.

- Data Fragmentation : Dividing the database into smaller, manageable pieces.

- Data Replication : Storing copies of the data at multiple sites.

- Data Allocation : Determining where each piece of data will be stored.

- Communication Network : The infrastructure that connects the distributed databases.

- Distributed DBMS Software : Manages the distribution, ensuring data consistency, integrity, and transparency to users.

- How does a DDBMS provide transparency to users regarding data distribution?

- Answer : A DDBMS provides transparency through:

- Location Transparency : Users do not need to know the physical location of data.

- Replication Transparency : Users are unaware of data replication.

- Fragmentation Transparency : Users interact with the database as a whole without needing to know about fragmentation.

2. Levels of Data and Process Distribution

- Describe the different levels of data distribution and provide examples of each.

- Answer :

- Fragmentation : Dividing a database into smaller pieces.

- Example : A customer table split into East Coast and West Coast fragments.

- Replication : Duplicating data across multiple sites.

- Example : A product catalog replicated at all regional offices.

- Allocation : Assigning specific fragments to different sites.

- Example : Sales data for the North region stored in a Chicago server, while South region data is stored in an Atlanta server.

- How do horizontal and vertical fragmentation differ?
  - Answer :
    - Horizontal Fragmentation : Divides a table into subsets of rows.
      - Example : A table of employees split into two tables: one for employees in the US and another for employees in Europe.
    - Vertical Fragmentation : Divides a table into subsets of columns.
      - Example : Splitting an employee table into two tables, one with personal details and another with job-related information.

- What are the benefits and challenges of data replication in distributed databases?
  - Answer :
    - Benefits :
      - Improved data availability and reliability.
      - Faster query response times due to data being closer to users.
    - Challenges :
      - Maintaining data consistency across replicas.
      - Increased storage and maintenance costs.
      - Complexity in handling updates.

- Explain the difference between centralized, decentralized, and hybrid process distribution.
  - Answer :
    - Centralized : All data and processing are done at a single location.
      - Example : A central server handling all database requests.
    - Decentralized : Data and processing are spread across multiple locations with local autonomy.
      - Example : Independent databases in different branches of a multinational company.
    - Hybrid : Combines aspects of both centralized and decentralized systems.
      - Example : Centralized control with distributed data storage and processing.

3. Transaction Management in Distributed Databases

- What are the ACID properties, and why are they important in transaction management?
  - Answer :
    - Atomicity : Ensures that all parts of a transaction are completed successfully or none at all.

- Consistency : Ensures that a transaction brings the database from one valid state to another.

- Isolation : Ensures that transactions do not interfere with each other.

- Durability : Ensures that once a transaction is committed, it remains so even in the event of a system failure.

- Importance : These properties ensure the reliability, integrity, and consistency of data, which are crucial for maintaining trust and correctness in database operations.

- Describe the Two-Phase Commit Protocol and its role in maintaining transaction integrity in distributed databases.

- Answer :

- Two-Phase Commit Protocol :

- Prepare Phase : The coordinator sends a prepare request to all participating nodes, asking if they can commit the transaction.

- Commit Phase : If all nodes agree, the coordinator sends a commit request; otherwise, it sends a rollback request.

- Role : Ensures that a distributed transaction is either fully committed or fully rolled back across all participating nodes, maintaining data integrity and consistency.

4. Trade-offs of Implementing a Distributed Data System

- Discuss the main advantages of implementing a distributed data system.

- Answer :

- Improved reliability and availability: Data is available even if one site fails.

- Enhanced performance: Queries can be processed closer to where the data is used.

- Scalability: Easier to add new nodes to the system.

- Local autonomy: Sites can operate independently.

- What are the primary challenges and disadvantages associated with distributed databases?

- Answer :

- Increased complexity: More difficult to manage and maintain.

- Security challenges: Protecting data across multiple sites is harder.

- Higher costs: Infrastructure and maintenance costs are higher.

- Difficult maintenance and administration: Ensuring consistency and synchronization across distributed sites is complex.

Chapter 12: Database Administration and Security

1. Data as a Valuable Business Asset

- Why is data considered a valuable business asset?

- Answer : Data is crucial for decision-making, strategic planning, and daily operations. It helps businesses understand customer behavior, market trends, and internal processes, leading to better-informed decisions and competitive advantage.

- How can effective data management enhance business performance?

- Answer : Effective data management ensures accurate, timely, and relevant information is available for decision-making. It improves efficiency, reduces costs, enhances customer satisfaction, and supports regulatory compliance.

2. Database's Critical Role in an Organization

- What roles do databases play in supporting organizational operations and decision-making?

- Answer : Databases store and organize data, making it easily accessible for various applications. They support transaction processing, reporting, data analysis, and business intelligence, enabling informed decision-making and efficient operations.

- How do databases ensure data integrity and security?

- Answer : Databases enforce data integrity through constraints, rules, and validation checks. Security is ensured through access controls, encryption, authentication, and regular audits to protect against unauthorized access and breaches.

3. Database Administrator (DBA) Roles

- Differentiate between the managerial and technical roles of a Database Administrator (DBA).

- Answer :

  - Managerial Role :

    - Policy formulation: Setting data management policies.

    - Planning and strategy: Long-term planning for data needs.

    - Data governance: Ensuring data quality and compliance.

  - Technical Role :

    - Database design: Structuring databases for efficiency and performance.

- Performance tuning: Optimizing database performance.

- Backup and recovery: Ensuring data is regularly backed up and can be recovered in case of failure.

- What are the key responsibilities of a DBA in ensuring database performance and security?

- Answer :

- Regularly monitoring database performance and tuning for optimal speed.

- Implementing and maintaining robust security measures.

- Performing regular backups and ensuring data recovery procedures are in place.

- Managing user access and permissions to protect data integrity.

4. Data Security

- Explain the concepts of confidentiality, integrity, and availability in the context of data security.

- Answer :

- Confidentiality : Ensuring only authorized individuals have access to data.

- Integrity : Ensuring data is accurate, consistent, and protected from unauthorized changes.

- Availability : Ensuring data is accessible to authorized users when needed.

- What security measures can be implemented to protect databases from unauthorized access and breaches?

- Answer :

- Authentication : Verifying the identity of users.

- Authorization : Granting appropriate access rights to users.

- Encryption : Protecting data by converting it into a secure format.

- Auditing and Monitoring : Regularly reviewing access logs and monitoring database activity for suspicious behavior.

5. Database Administration Tools and Strategies

- Describe the tools used for database administration in Oracle and SQL Server Management Studio (SSMS).

- Answer :

- Oracle : Offers tools like Oracle Enterprise Manager, which provides comprehensive database management capabilities, including performance monitoring, backup, recovery, and security management.

- SQL Server Management Studio (SSMS) : An integrated environment for managing SQL Server, offering tools for database design, query execution, performance tuning, and security configuration.

- What strategies should DBAs employ to maintain database performance and security?

- Answer :

- Regular performance monitoring and tuning.

- Implementing robust security policies and regular security audits.

- Ensuring regular backups and testing recovery procedures.

- Keeping software and patches up to date to protect against vulnerabilities.

Chapter 13: Managing Transactions and Concurrency

1. Database Transactions

- What defines a database transaction, and what are its key properties?

- Answer : A database transaction is a sequence of operations executed as a single logical unit of work. Its key properties are ACID:

- Atomicity : Ensures all operations within the transaction are completed successfully or none are.

- Consistency : Ensures the transaction brings the database from one valid state to another.

- Isolation : Ensures that intermediate states of the transaction are not visible to other transactions.

- Durability : Ensures that once a transaction is committed, it remains so, even in the event of a system failure.

- Why is it important for transactions to exhibit the ACID properties?

- Answer : The ACID properties ensure the reliability, consistency, and integrity of data, which are crucial for maintaining trust in database operations and ensuring correct application behavior.

2. Concurrency Control

- What is concurrency control, and why is it necessary in database management?

- Answer : Concurrency control ensures that multiple transactions can execute simultaneously without interfering with each other, maintaining data consistency and isolation. It is necessary to prevent issues like lost updates, dirty reads, and uncommitted data being accessed.

- Compare and contrast optimistic and pessimistic concurrency control methods.

- Answer :

- Optimistic Concurrency Control : Assumes conflicts are rare and checks for conflicts only at commit time. If a conflict is detected, the transaction is rolled back.

- Example : A booking system allowing multiple users to book seats, checking for overbooking only when finalizing the transaction.

- Pessimistic Concurrency Control : Locks resources before accessing them to prevent conflicts. It ensures that once a transaction locks a resource, other transactions cannot access it until the lock is released.

- Example : A bank ensuring that an account cannot be accessed by two transactions simultaneously by locking it during a withdrawal or deposit.

3. Locking Methods

- Describe the different types of locks (shared and exclusive) used in concurrency control.

- Answer :

- Shared Lock (S) : Allows multiple transactions to read a resource but not modify it.

- Example : Multiple users can read a product catalog at the same time.

- Exclusive Lock (X) : Prevents other transactions from reading or modifying a resource.

- Example : Updating an inventory count in a warehouse management system.

- What are the advantages and disadvantages of row-level, table-level, and page-level locking?

- Answer :

- Row-level Locking :

- Advantages : Provides the highest level of concurrency by allowing different transactions to access different rows simultaneously.

- Disadvantages : Requires more overhead to manage many locks.

- Table-level Locking :

- Advantages : Simple to implement with low overhead.

- Disadvantages : Reduces concurrency as it locks the entire table, preventing other transactions from accessing any part of it.

- Page-level Locking :

- Advantages : Balances between row-level and table-level locking, locking only a page of rows.

- Disadvantages : Can still lead to contention if multiple transactions need access to rows on the same page.

4. Database Recovery Management

- Why is database recovery management crucial for maintaining database integrity?

- Answer : Database recovery management is essential to ensure that data can be restored to a consistent state after a failure, such as a system crash, power outage, or software error. It helps prevent data loss and corruption, maintaining the reliability and integrity of the database.

- Explain the techniques of deferred update and immediate update in the context of database recovery.

- Answer :

- Deferred Update : Changes are not applied to the database until the transaction commits. If the transaction fails, no changes are made, simplifying rollback.

- Example : An online shopping cart where items are only reserved when the purchase is confirmed.

- Immediate Update : Changes are applied to the database as they occur but logged so they can be undone if necessary. It allows for more immediate application of changes but requires complex logging and rollback mechanisms.

- Example : A bank transaction updating account balances immediately but keeping a log to revert in case of failure.

- What is the purpose of checkpoints in database recovery management?

- Answer : Checkpoints save the state of the database at a specific point in time. They help speed up recovery by reducing the amount of log data that must be processed after a failure. By restoring to the checkpoint state and then applying subsequent log records, recovery can be completed more efficiently.