NAME: YUGIESHWARI D REG.NO: 212720080110S

SUBJECT: IT18305 DATABASE SYSTEMS

DATE: 11-02-2022 SEM 03 CAT-03

PART-C

7 (a) The technology behind this functionality is Geographical Data.

Geographical Data:

- They are spatial in nature
- Maps and satellite images are typical examples
- Provides much more detailed information associated with location such as elevation, soil type, annual rainfall.

Application of Geographic data:

They have variety of uses including online-map services; vehicle navigation systems; distribution-network information for public-service utilities such as telephone, electric power and water supply systems; and land-usage information for ecologists and planners.

Web based road map services form a very widely used application of map services data. An important benefit is that it is easy to scale maps to desired size.

Road maps services also provides information like, layout of road, speed limit, restrictions. With these additional information, maps can be used for getting directions from one place to another and automatic trip planning. Users can query online information about services to locate like hotels, gas, spa with desired offerings and price ranges.

In recent years web based map services have defined API's that allow customers programmers to create customised maps.

Vehicle Navigation systems are systems that are mounted in automobiles and provide road maps and trip-planning services. They include Global Positioning System (GPS) unit, which uses information broadcast from GPS satellites to find current location with an accuracy of tens of meters. It finds location in terms of latitude, longitude and elevation. The navigation system can query geographic database to find where and on which road vehicle is located. Geographic database for public utility information become very important. They are categorised into two types:

1) Raster data

2) Vector data.

1) Raster data: consists of bitmaps or pixel maps in two or more dimensions. A typical example of 2-d raster image is satellite image of an area. It also includes location of image with latitude, longitude, resolution. Raster data is often represented as tiles, each covering a fixed size area. It can also be three-dimensional, with help of satellite.

2) Vector data: constructed from basic geometric objects. In context of geographic data, points are usually represented by latitude and longitude. Map data are often represented in vector format. The vector representation is more compact than the raster representation in some applications.

Topological information, that is information about elevation is represented in raster form. The information can even be converted to raster form before it is sent to user interface post scripting languages. Map services provide API's that follow users to create specialised map displays. Map API's for specific language are built on top of a web service that provides underlying map data.

⑧ b Deadlock Handling:

Consider following two transactions.

T_1: write(x) write(y)	T_2: write(x) write(y)
x - lock on x	
write (x)	
	x lock on y
	write (y)
	wait for x - lock on x
wait for x - lock on y	

Result: Deadlock which cannot be detected locally at either side.

Deadlock Detection:

* In centralized deadlock - detection approach, a global wait for graph is constructed and maintained in a single site. The dead lock coordinator

→ Real graph: Real, but unknown state of system

→ constructed graph: Approximation generated by the controller during algorithm execution.

* The global wait for graph can be constructed when,

→ a new graph inserted . removed

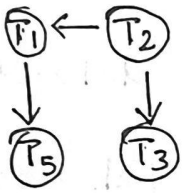
→ A number of changes have occurred in a local wait for graph

→ Co-ordinator needs to invoke cycle detection.

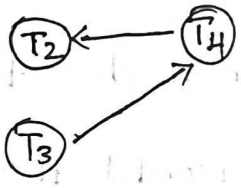
* If coordinator finds a cycle, it selects a victim and notifies all sites. The sites rollback transaction.

Local and Global wait for Graphs:

Local:

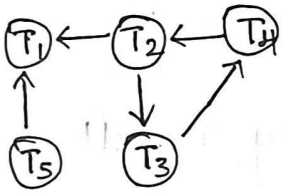


site1



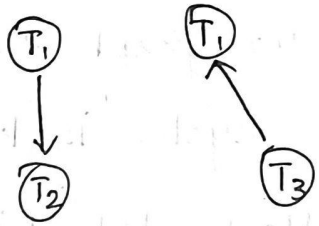
site 2

Global:



Example: wait for graph for false cycles

Initial state:



* suppose starting from site

1. T_2 releases resources at $S1$

Resulting in message remove $T_1 \rightarrow T_2$ message from transaction manger at S_1.

2. The T_2 requests resource by T_3 at S_2.

Resulting in insert $T_2 \rightarrow T_3$ from S_2.

suppose further insert message reaches before the delete message

This can happen due to network delays.

The coordinator would then find a false cycle.

$$T_1 \rightarrow T_2 \rightarrow T_3 \rightarrow T_1$$

Distributed deadlocks:

* Unnecessary rollbacks may result

→ When deadlock has indeed occurred and victim has been picked and meanwhile one of transactions have aborted

→ Due to global wait for graph false cycles. However likelihood for false cycles is low.

→ In distribution deadlock-detection, sites exchange wait-for information and check for deadlocks.

b) Challenges in maintaining Data Consistency:

* Data discrepancy occurs when data in target database deviates from source database. The extent to which data deviates depends on various factors.

Some Potential Causes:

* Migration Errors:

Different kinds of migration tools are employed to facilitate the initial load of target database before replication can begin.

* Lift and shift workload to cloud:

since world is moving towards cloud, the lift and shift workload is the need of Today's IT world.

* Difference in source and Target:

Difference in data can cause subtle discrepencies to happen during migration & replication.

* Instantiation Errors:

Before migration or replication can begin, the target databases will need to be instantiated.

* Configuration

Improper and unintended configuration of replication products can cause discrepencies.

* Replication Errors:

With asynchoronous replication, there will be a short lag between base and delivery database.

* User Errors:

Often taryet databases are treated to offload query processing from source database.

REQUIREMENTS FOR MANAGING DATA CONSISTENCY:

* High speed, low impact data compressions
* Support for Heterogeneous database
* Support for handling large data volumes
* Minimally intrusive
* Data security
* Comparision of huge table through automated and manual partioning.
* Early to use, understand, configure, deploy and diagnise.

PART - A

① In 2 disk mirrored case, we assume disk A and disk B. In order to loss data A and B have to be failed at same time. If A is already failed, then within 100,000 hours B will fail, then data will be lost and vice-versa.

In first case, A failed for 100 hours every 100,000 hrs.

So, in order to make B fail, it will need $100,000^2/100$ hrs.

In other case time is reduced to $100,000^2/(2 \times 100)$.

② Database indexing: Hashing Table

Justification: Hash tables are commonly used to distribute lows amongst nodes, reducing network traffic for hash joints.

∴ Type of database is hashing table, Database indexing

③ Advantage:

1) Computer based system provides enhanced data retrieval techniques to retrieve data.

2) It is early to edit information in computers to form files.

Disadvantages:

1) Data becomes Redundant in files

2) Because of Redundancy data inconsistency occurs.

(4) MySQL enables restrictions to be placed on reuse of previous passwords. To establish password-reuse policy globally, we use password history and password reuse interval system variables.

(5) SQL supports storing and querying of geospatial data, that is, location data referenced to earth. Common models of these data are planar and geodetic co-ordinate systems. Main distinction between these two systems is latter takes curvature of Earth. SQL supports geometry and geography.