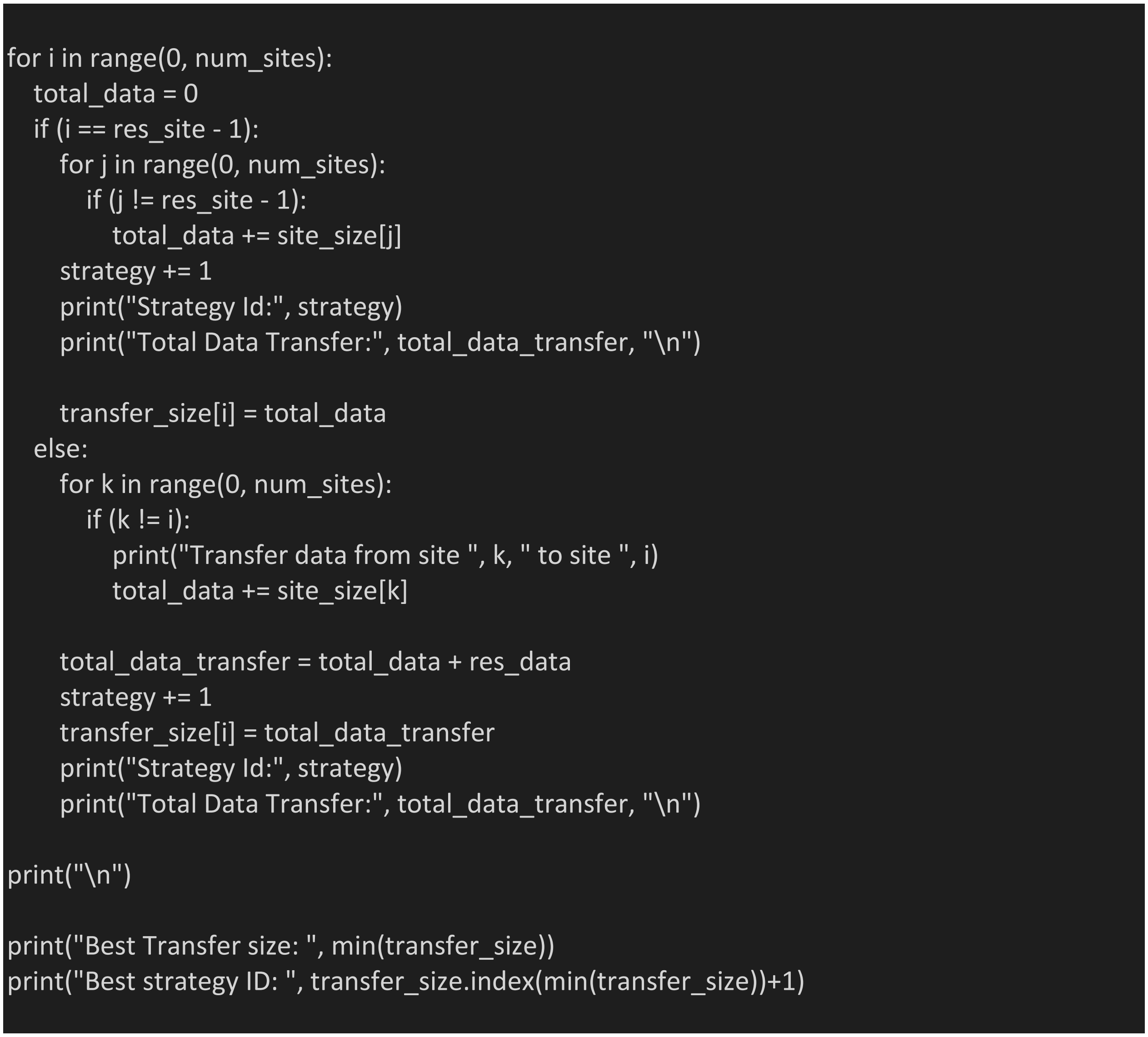
Name: Mareena Fernandes
Roll no.: 8669
Class: TE IT
Batch: B

Experiment No.: 7

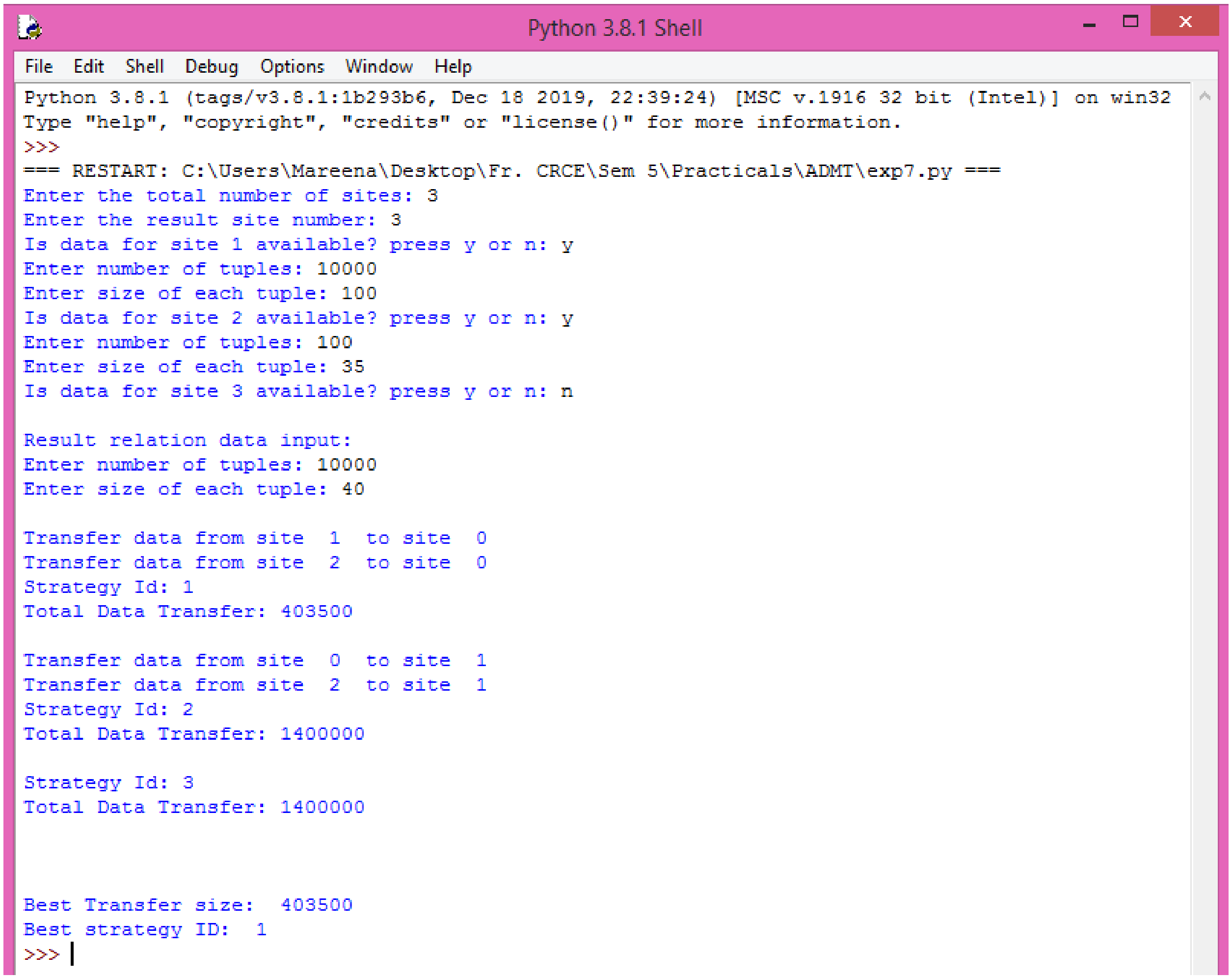
exp7.py:

```
num_sites = int(input("Enter the total number of sites: "))
res_site = int(input("Enter the result site number: "))
site_size = [None] * num_sites
transfer_size = [None] * num_sites

# Accept data information for each site(check for availability)

for i in range(0, num_sites):
    is_available = input(
        "Is data for site {} available? press y or n: ".format(i + 1))

    if (is_available == 'y'):
        num_tuples = int(input("Enter number of tuples: "))
        size = int(input("Enter size of each tuple: "))

        site_size[i] = num_tuples * size

    else:
        site_size[i] = 0

# Accept result relation data

print("\nResult relation data input: ")
num_tuples = int(input("Enter number of tuples: "))
size = int(input("Enter size of each tuple: "))
res_data = num_tuples * size

print()

(strategy, total_data, total_data_transfer) = (0, 0, 0)

# Find total strategies and calculate data transfer for each strategy
```



```
for i in range(0, num_sites):
    total_data = 0
    if (i == res_site - 1):
        for j in range(0, num_sites):
            if (j != res_site - 1):
                total_data += site_size[j]
        strategy += 1
        print("Strategy Id:", strategy)
        print("Total Data Transfer:", total_data_transfer, "\n")

        transfer_size[i] = total_data
    else:
        for k in range(0, num_sites):
            if (k != i):
                print("Transfer data from site ", k, " to site ", i)
                total_data += site_size[k]

        total_data_transfer = total_data + res_data
        strategy += 1
        transfer_size[i] = total_data_transfer
        print("Strategy Id:", strategy)
        print("Total Data Transfer:", total_data_transfer, "\n")

print("\n")

print("Best Transfer size: ", min(transfer_size))
print("Best strategy ID: ", transfer_size.index(min(transfer_size))+1)
```

Output:



```
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 22:39:24) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
=== RESTART: C:\Users\Mareena\Desktop\Fr. CRCE\Sem 5\Practicals\ADMT\exp7.py ===
Enter the total number of sites: 3
Enter the result site number: 3
Is data for site 1 available? press y or n: y
Enter number of tuples: 10000
Enter size of each tuple: 100
Is data for site 2 available? press y or n: y
Enter number of tuples: 100
Enter size of each tuple: 35
Is data for site 3 available? press y or n: n

Result relation data input:
Enter number of tuples: 10000
Enter size of each tuple: 40

Transfer data from site  1  to site  0
Transfer data from site  2  to site  0
Strategy Id: 1
Total Data Transfer: 403500

Transfer data from site  0  to site  1
Transfer data from site  2  to site  1
Strategy Id: 2
Total Data Transfer: 1400000

Strategy Id: 3
Total Data Transfer: 1400000



Best Transfer size:  403500
Best strategy ID:  1
>>> |
```

Post labs:

1. **Explain with example query processing in Distributed Databases using semi join operation.**

Ans:

1. A join where the result only contains the columns from one of the joined tables.
2. Useful in distributed databases, so we don't have to send as much data over a network.
3. Can dramatically speed up certain classes of queries.
4. Semi-join strategies are technique for query processing in distributed database systems.
5. Used for reducing communication cost.
6. Semi-join between two tables returns rows from the first table where one or more matches are found in the second table.
7. Different between a semi-join and a conventional join is that rows in the first table will be returned at most matches for a row in the first table, only one copy of the row will be returned.
8. Semi-joins are written using EXISTS or IN.

Query – Conventional Join

Select D.deptno, D.dname from dept D, emp E where E.deptno = D.deptno order by D.dept;

a) A department with N employees will appear in the list N times.
b) We could use a DISTANT keyword to get each department to appear only once.

Query – Semi Join

Select D.deptno, D.deptname from dept D, where exists (Select 1 from emp E where E.deptno = D.deptno) order by D.deptno

a) No department appears more than once.

Experiment No.: 8

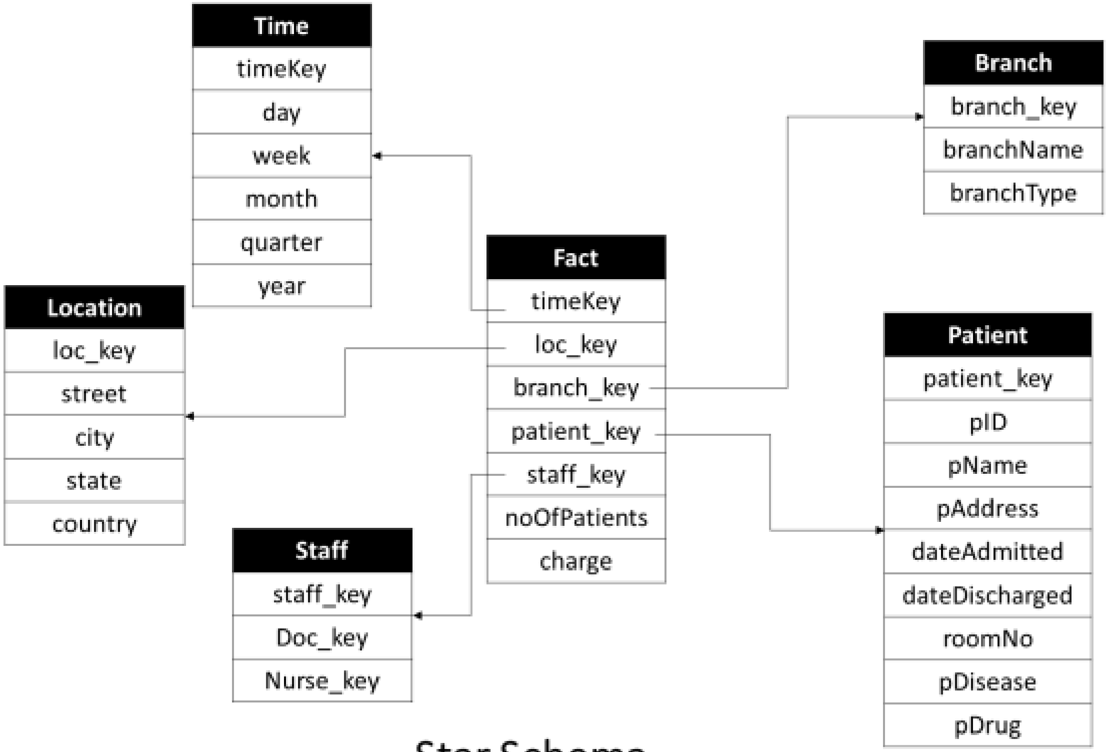
HOSPITAL MANAGEMENET SYSTEM

Information Package Diagram

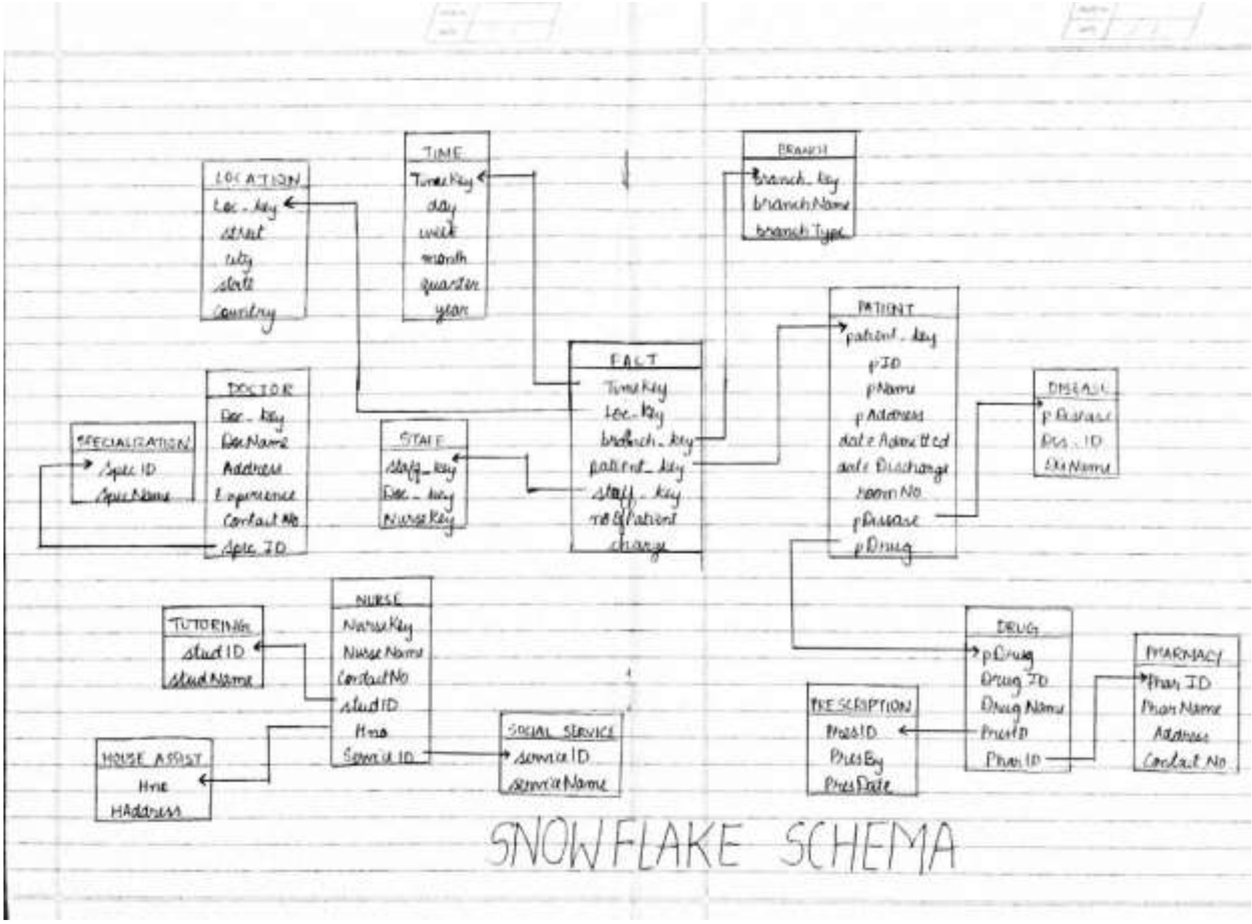
Business Process – Hospital Record

Time	Branch	Location	Patient	Staff
day	branchName	street	pID	Doctor
week	branchType	city	pName	Nurse
month		state	pAddress	
quarter		country	dateAdmitted	
year			dateDischarge	
			roomNo	
			pDisease	
			pDrug	

Facts: noOfPatients, charge



Star Schema

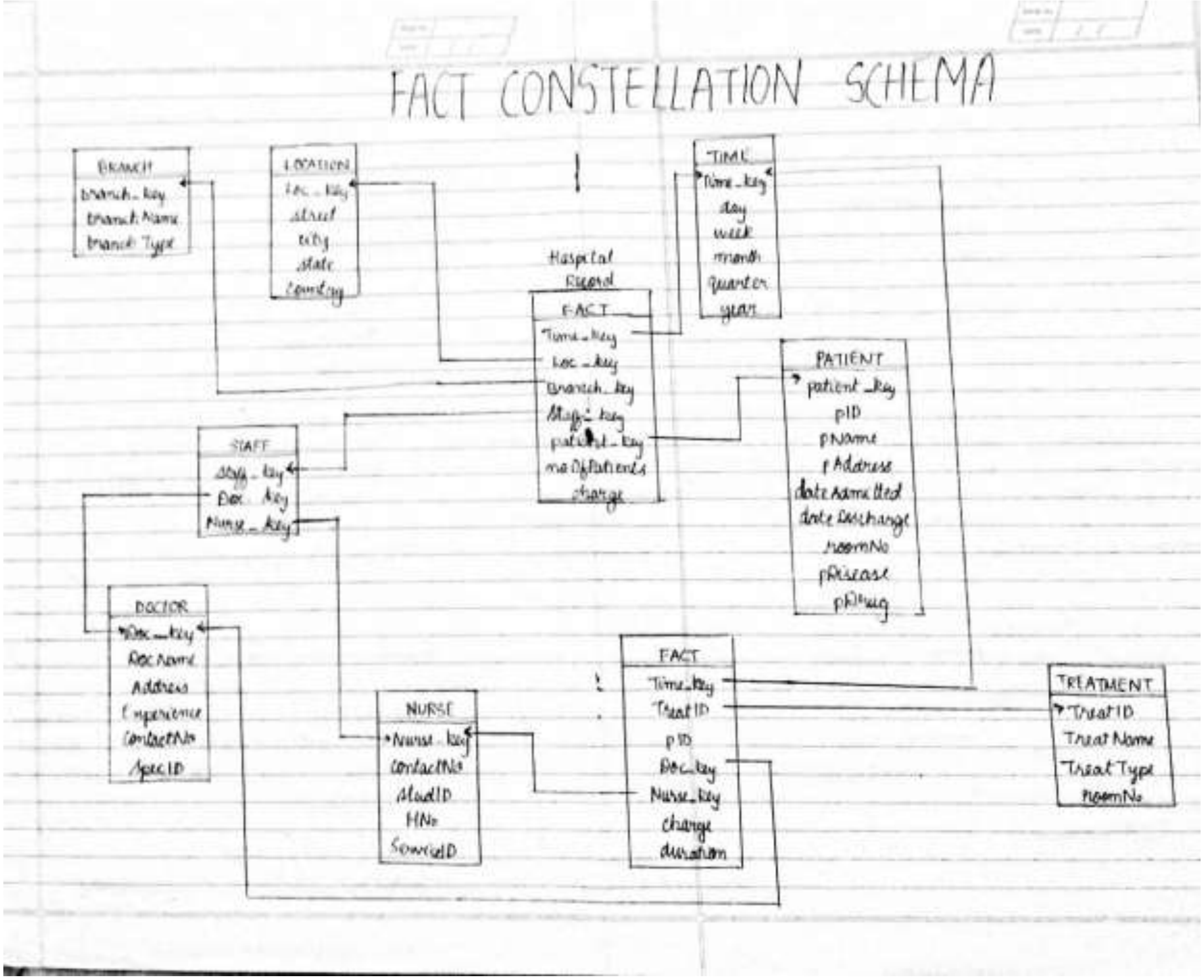


FACT CONSTELLATION SCHEMA

Business Process – Patient treatment analysis

Time	Treatment	Patient	Doctor	Nurse
day	TreatID	pID	Doc_key	Nurse_key
week	TreatName	pName	DocName	NurseName
month	TreatType	pAddress	Address	ContactNo
quarter	roomNo	dateAdmitted	Experience	StudID
year		dateDischarge	ContactNo	Hno
		roomNo	SpecID	Service
		pDisease		
		pDrug		

Facts: charge, duration


FACT CONSTELLATION SCHEMA
BRANCH
branch_key
branch Name
branch Type
LOCATION
Loc_key
street
city
state
Country
TIME
Time_key
day
week
month
Quarter
year
Hospital Record
FACT
Time_key
Loc_key
Branch_key
Staff_key
patient_key
no Of Patients
charge
PATIENT
patient_key
pID
pName
pAddress
date Admitted
date Discharge
roomNo
pDisease
pDrug
STAFF
staff_key
Doc_key
Nurse_key
DOCTOR
Doc_key
Doc Name
Address
Experience
ContactNo
SpecID
NURSE
Nurse_key
ContactNo
StudID
HNo
SourceID
FACT
Time_key
TreatID
pID
Doc_key
Nurse_key
charge
duration
TREATMENT
TreatID
Treat Name
Treat Type
RoomNo

Experiment No.: 9

1.
   ```
   CREATE TABLE sales1(state VARCHAR(20),city VARCHAR(20), area VARCHAR(20),profit
   NUMERIC(6),product_sold NUMERIC(4));
   ```

2.
   ```
   INSERT INTO sales1 VALUES('Maharastra','Mumbai','Dadar',1200,100);
   INSERT INTO sales1 VALUES('Gujrat','Ahamadabad','Jamalpur',12000,16);
   INSERT INTO sales1 VALUES('Maharastra','Pune','Lakshmi road',4500,10);
   INSERT INTO sales1 VALUES('Andhra-pradesh','Hydrabad','Mallapur',1500,18);
   INSERT INTO sales1 VALUES('Maharastra','Mumbai','Borivali',20000,25);
   INSERT INTO sales1 VALUES('Gujrat','Surat','Textile market',25000,116);
   INSERT INTO sales1 VALUES('Maharastra','Pune','Tilak road',8000,12);
   INSERT INTO sales1 VALUES('Andhra-pradesh','Vijayvada','M.G.Road',10000,2000);
   INSERT INTO sales1 VALUES('Gujrat','Surat','Ring road',27000,206);
   INSERT INTO sales1 VALUES('Andhra-pradesh','Hydrabad','Hitech City',1000,117);
   INSERT INTO sales1 VALUES('Rajasthan','Jaipur','Johari Bazar',11000,127);
   INSERT INTO sales1 VALUES('Madya-pradesh','Indore','Khajuri market',1000,20);
   ```

3.
   ```
   SELECT * FROM sales1;
   ```

4.
   ```
   SELECT * FROM sales1 ORDER BY state,city,area;
   ```

5.
   ```
   SELECT state, SUM(profit),SUM(product_sold) FROM sales1 GROUP BY state;
   ```

6.
   ```
   SELECT city, SUM(profit),SUM(product_sold) FROM sales1 GROUP BY city;
   ```

7.
   ```
   SELECT area, SUM(profit),SUM(product_sold) FROM sales1 GROUP BY area;
   ```

8.
   ```
   SELECT state, city, area, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY GROUPING
   SETS(state,city,area);
   ```

9.
   ```
   SELECT state, city, area, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY GROUPING
   SETS(state,city,area,());
   ```

10.
    i. ```
       SELECT state, city, area, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY
       ROLLUP(state,city,area);
       ```

    ii. ```
        SELECT state, city, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY
        ROLLUP(state,city);
        ```

    iii. ```
         SELECT state, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY ROLLUP(state);
         ```
         ```

11.

i. SELECT state, city, area, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY CUBE(state,city,area);

ii. SELECT state, city, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY CUBE(state, city);

iii. SELECT city, area, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY CUBE(city, area);

iv. SELECT state, area, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY CUBE(state, area);

v. SELECT state, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY CUBE(state);

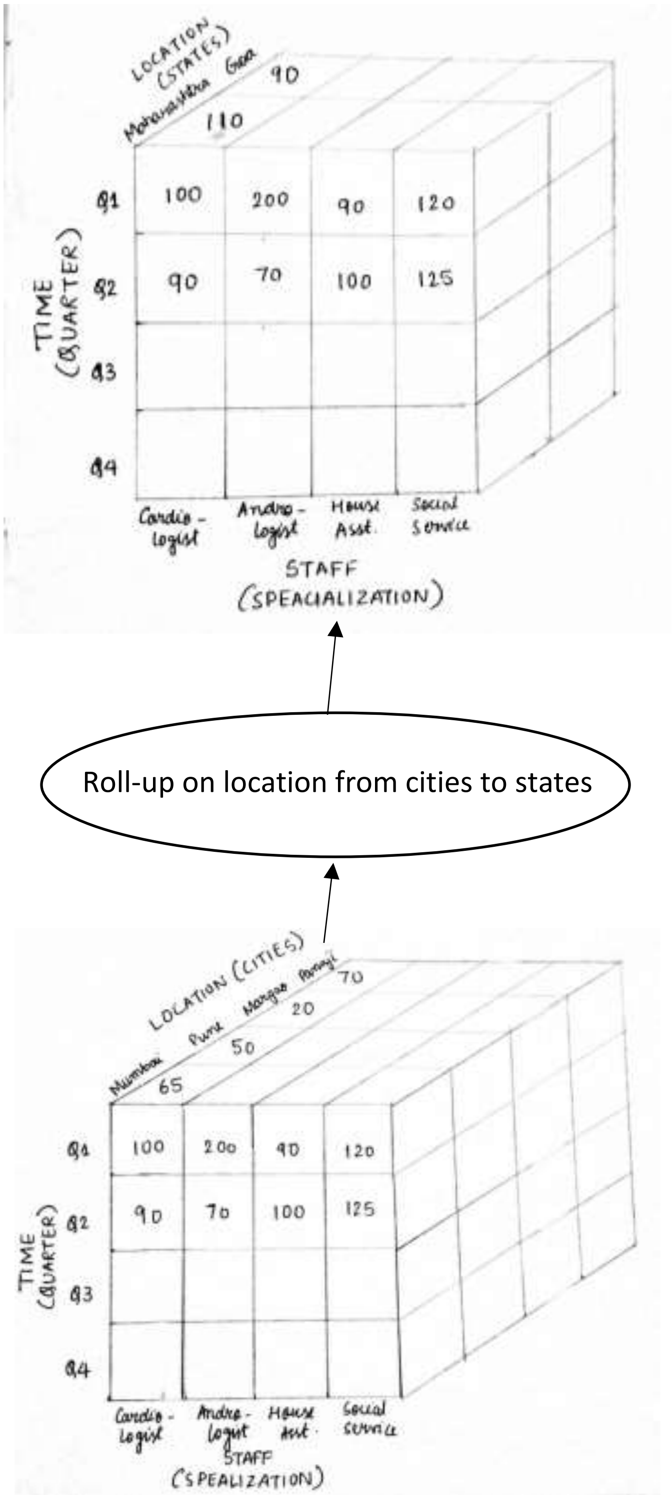
vi. SELECT city, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY CUBE(city);

vii. SELECT area, SUM(profit), SUM(product_sold) FROM sales1 GROUP BY CUBE(area);

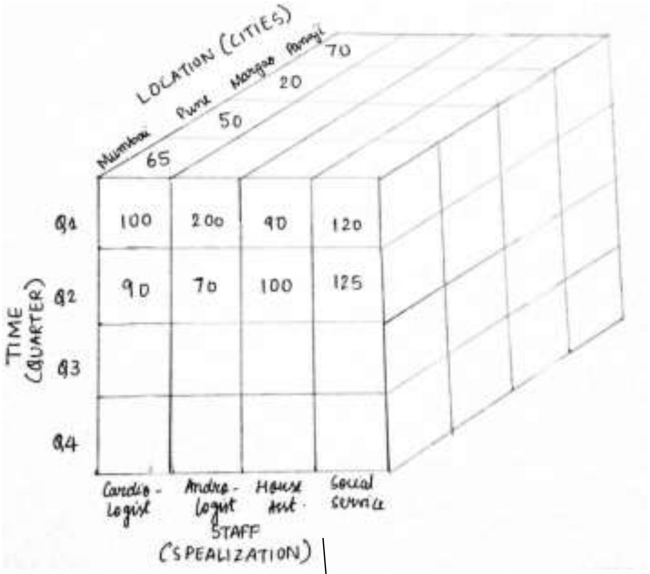
Post labs:

1. **Draw OLAP cube for the case study (for experiment no 8) and perform different OLAP operations (Roll up, Drilldown, Slice, Dice and Pivot).**

Ans:

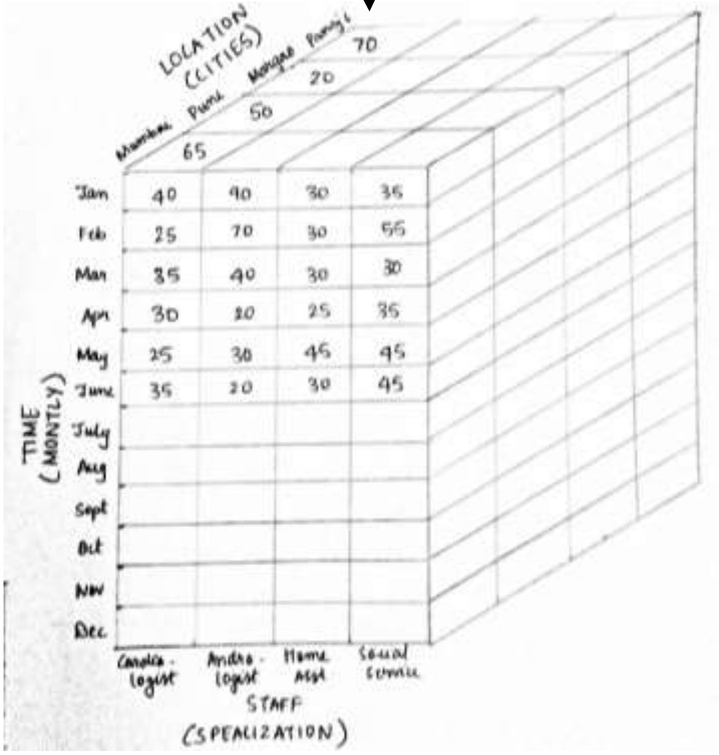


Roll-up on location from cities to states

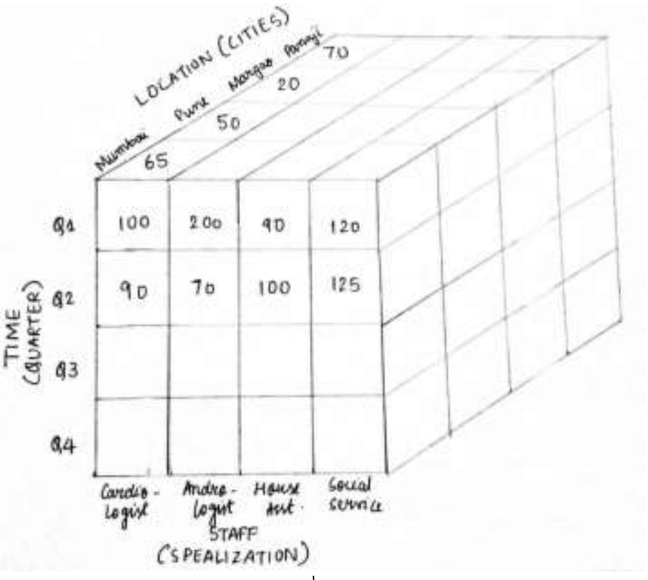


	Cardio-logist	Andro-logist	House Asst.	Social service
Q1	100	200	90	120
Q2	90	70	100	125
Q3				
Q4				

Drill down on time from quarters to months



	Cardio-logist	Andro-logist	Home Asst.	Social service
Jan	40	90	30	35
Feb	25	70	30	55
Mar	35	40	30	30
Apr	30	20	25	35
May	25	30	45	45
June	35	20	30	45
July				
Aug				
Sept				
Oct				
Nov				
Dec				



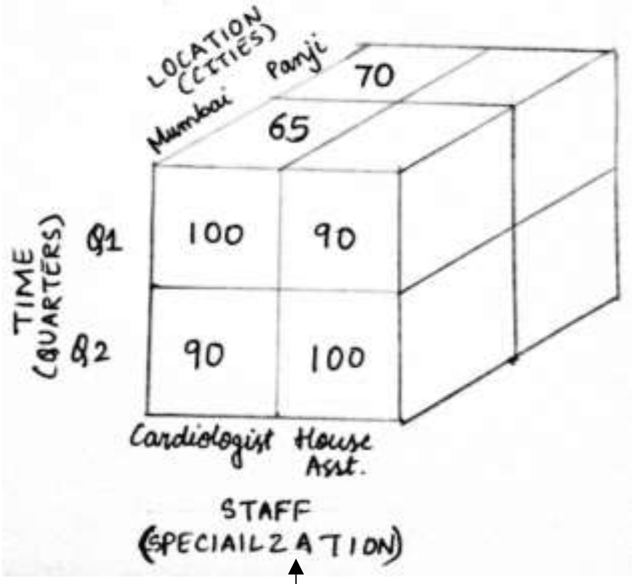
Slice for time Q_2

LOCATION (CITIES)

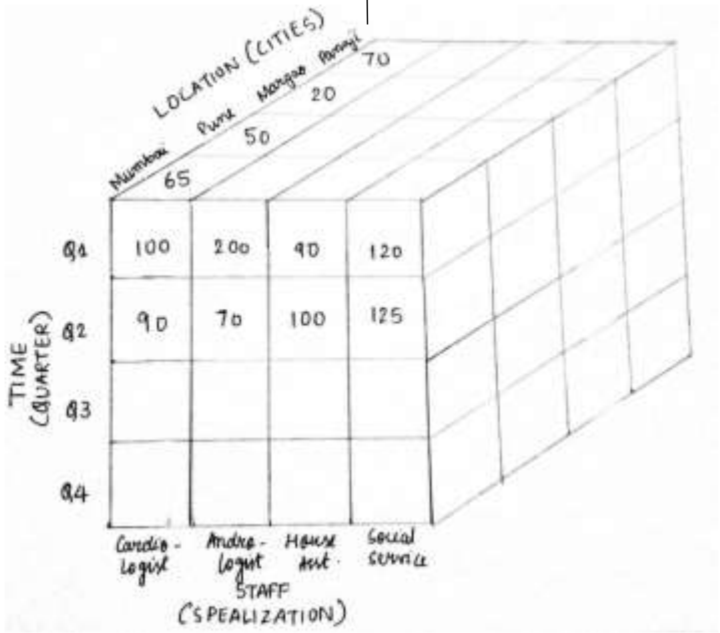
Panaji				
Margao				
Pune				
Mumbai	90	70	100	125
	Cardiologist	Andrologist	House Assistant	Social Service

STAFF

(SPEACIALIZATION)



Dice for (location = ‘Panaji’ or ‘Mumbai’)
and (time = ‘Q_1’ or ‘Q_2’)
and (staff = ‘Cardiologist’ or ‘House Assistant’)



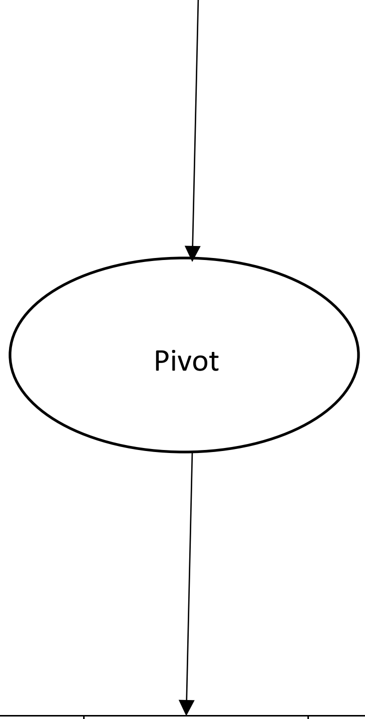
LOCATION

(CITIES)

Panaji				
Margao				
Pune				
Mumbai	100	200	90	120
	Cardiologist	Andrologist	House Assistant	Social Service

STAFF

(SPEACIALIZATION)



STAFF

(SPEACIALIZATION)

Cardiologist				100
Andrologist				200
House Assistant				90
Social Service				120
	Panaji	Margao	Pune	Mumbai

LOCATION

(CITIES)

EXPERIMENT NO.: 10

MINI PROJECT ON
MOBILE DATABASE SYSTEM

MOBILE DATABASE SYSTEM

Team Members

Shresht Acharya	8649
Mareena Fernandes	8669
Deepak Tiwari	8707
Ashally Tuscano	8710

“You can have data without information, but you cannot have information without data”

-- Daniel Keys Moran

Abstract

- The rapidly expanding technology of mobile communication will give mobile users capability of accessing information from anywhere and any time.
- The wireless technology has made it possible to achieve continuous connectivity in mobile environment.
- When the query is specified as continuous, the requesting mobile user can obtain continuously changing result.
- In order to provide accurate and timely outcome to requesting mobile user, the locations of moving object has to be closely monitored.

What is Mobile Database System?

- A mobile database is a database that can be connected to a mobile computing device over a wireless mobile network.
- As the use of laptops, mobiles and PDAs is increasing more and more applications increasing to reside in the mobile system.
- Mobile data-driven applications enable us to access any data from anywhere, anytime.

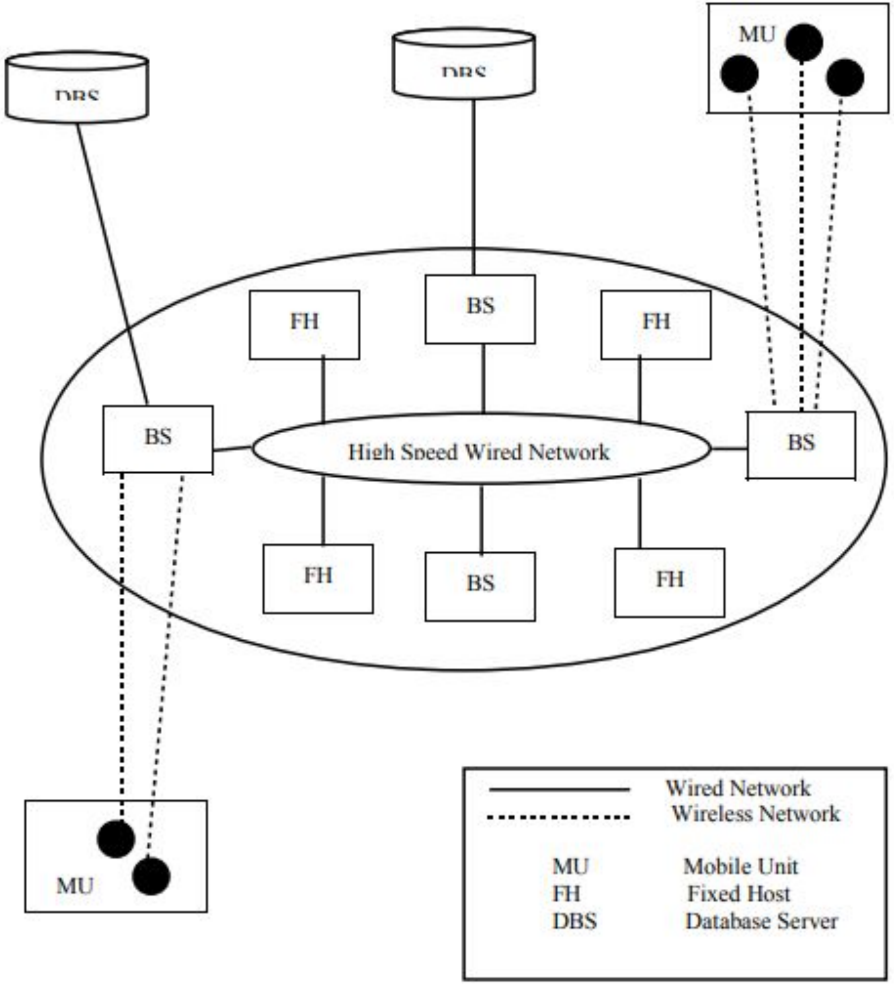


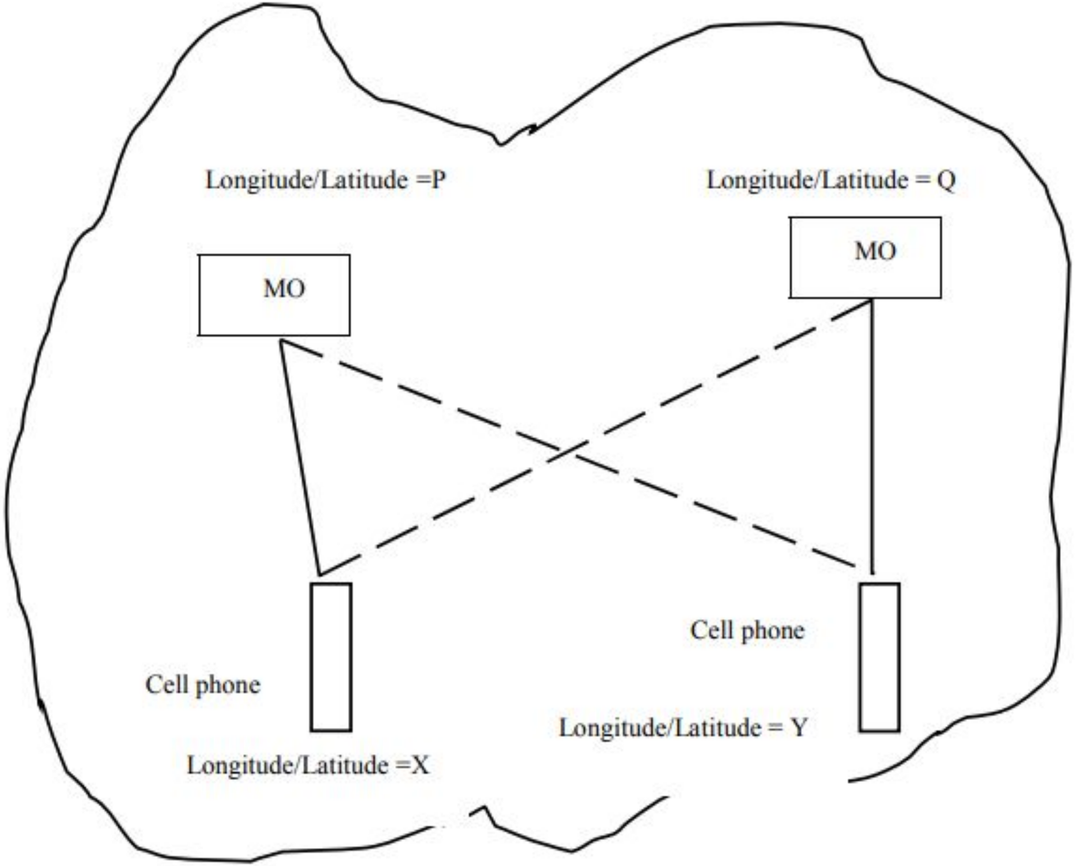
Figure Architecture of Mobile Environment

OBJECTIVES

The objective of this project is to discuss the problem related to the role of personal and terminal mobility and query processing in the mobile environment.In mobile environment, elements of the network are very dynamic and can be extremely volatile. Consider a database representing information about moving objects and their position in addition to information about stationary objects.

Terminal Mobility:

- In terminal mobility, the connection is established between two points and not between the two persons calling each other.
- This type of connection allows the use of communication devices to be shared among anybody.
- It allows a mobile unit to access the desired services from any location while in motion or stationary, irrespective of who is carrying the unit.

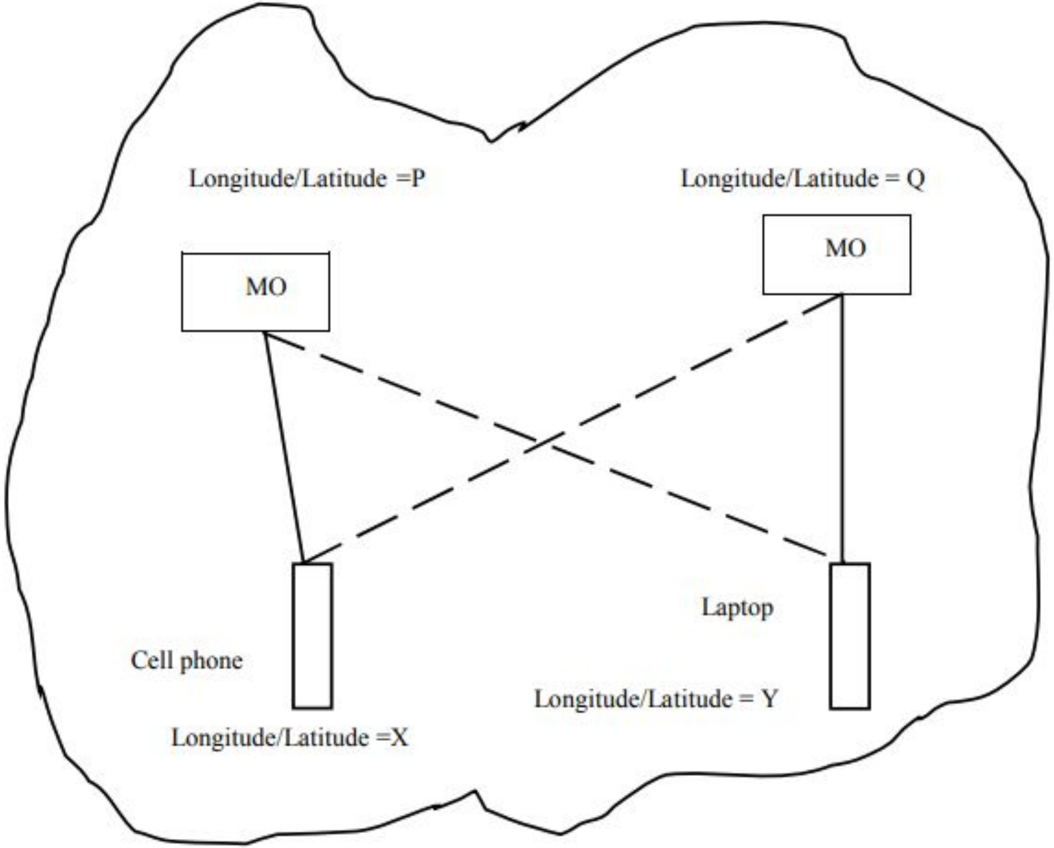


MO-Mobile Object (Car, Bike etc)

Figure Terminal Mobility

Personal Mobility:

- In personal mobility the mobility of a person is supported and thus a user does not have to carry any communication equipment with them.
- They can use any communication device for establishing contact with the other party from anywhere.
- In personal mobility each person has to be uniquely identified, and one way to do this is via a unique identification number



MO-Mobile Object (Car, Bike etc)

Figure Personal Mobility

EFFECT OF MOBILITY ON QUERY PROCESSING

The effects of mobility on query processing require that algorithms employed must be capable of managing frequent loss and appearance of mobile device in the network.

Different Location Models

- Current location models can be classified into symbolic or geometric models
- Symbolic models refer to a location by some abstract symbols.
- In geometric model locations are specified as an n-dimensional coordinate or a set of coordinates defining an area’s bounding geometric shape.

Different Query Types

Mobile queries can be classified into five categories, as follows:

- Non Location Related Query (NLRQ)
- Location Dependent Queries (LDQ)
- Location Aware Query (LAQ)
- Continuous Query (CQ)
- Ad Hoc Query

Query Optimization

- Query optimization methods try in general to obtain execution plans which minimize CPU, input/output and communication costs.
- In a mobile distributed environment, the communication costs are much more difficult to estimate because the mobile host may be situated in different locations.

The other effects of mobility on query processing are:

- Query Execution
- The Impact of Portable Devices Limitations
- The Impact of Wireless Communication

QUERY PROCESSING IN MOBILE DATABASES

Current Work on Query Processing

- Query processing deals with designing algorithms that analyze queries and convert them into a series of data manipulation operations.
- The main function of a query processor is to transform high-level query (typically, in relational calculus) into an equivalent lower-level query

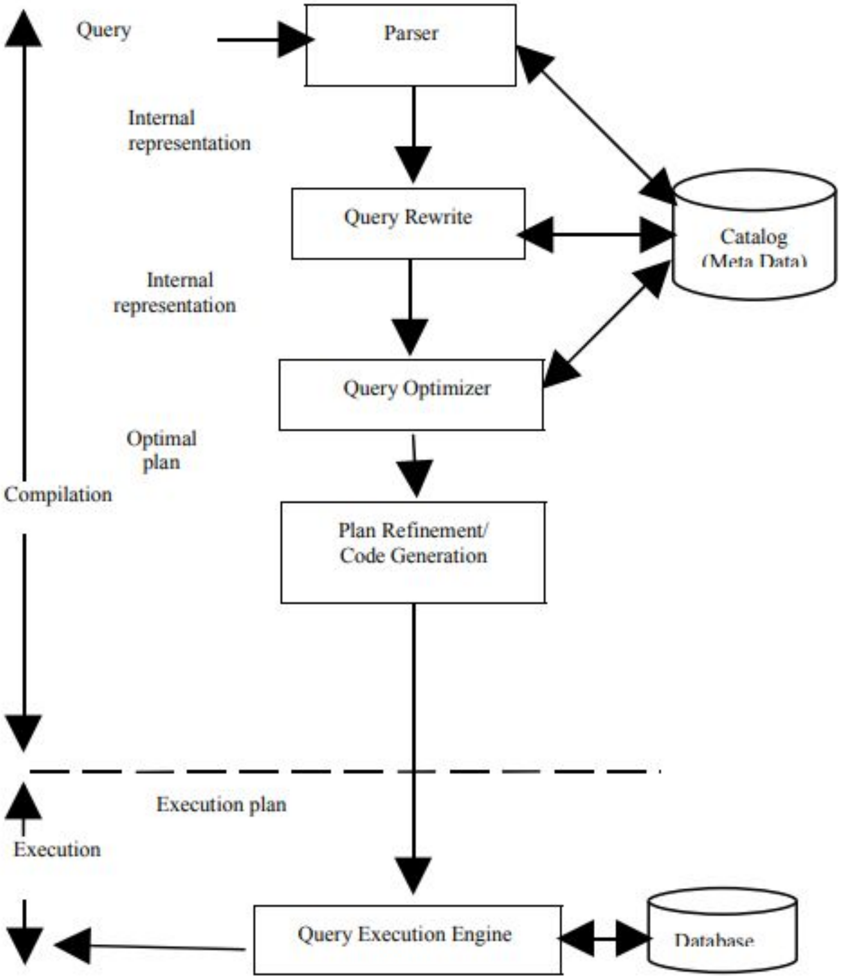


Figure Phase of Query Processing

Types of Query Processing Systems

We divide query-processing system for mobile databases into three parts, namely:

- Mobile Client System
- Wireless network System
- Server System

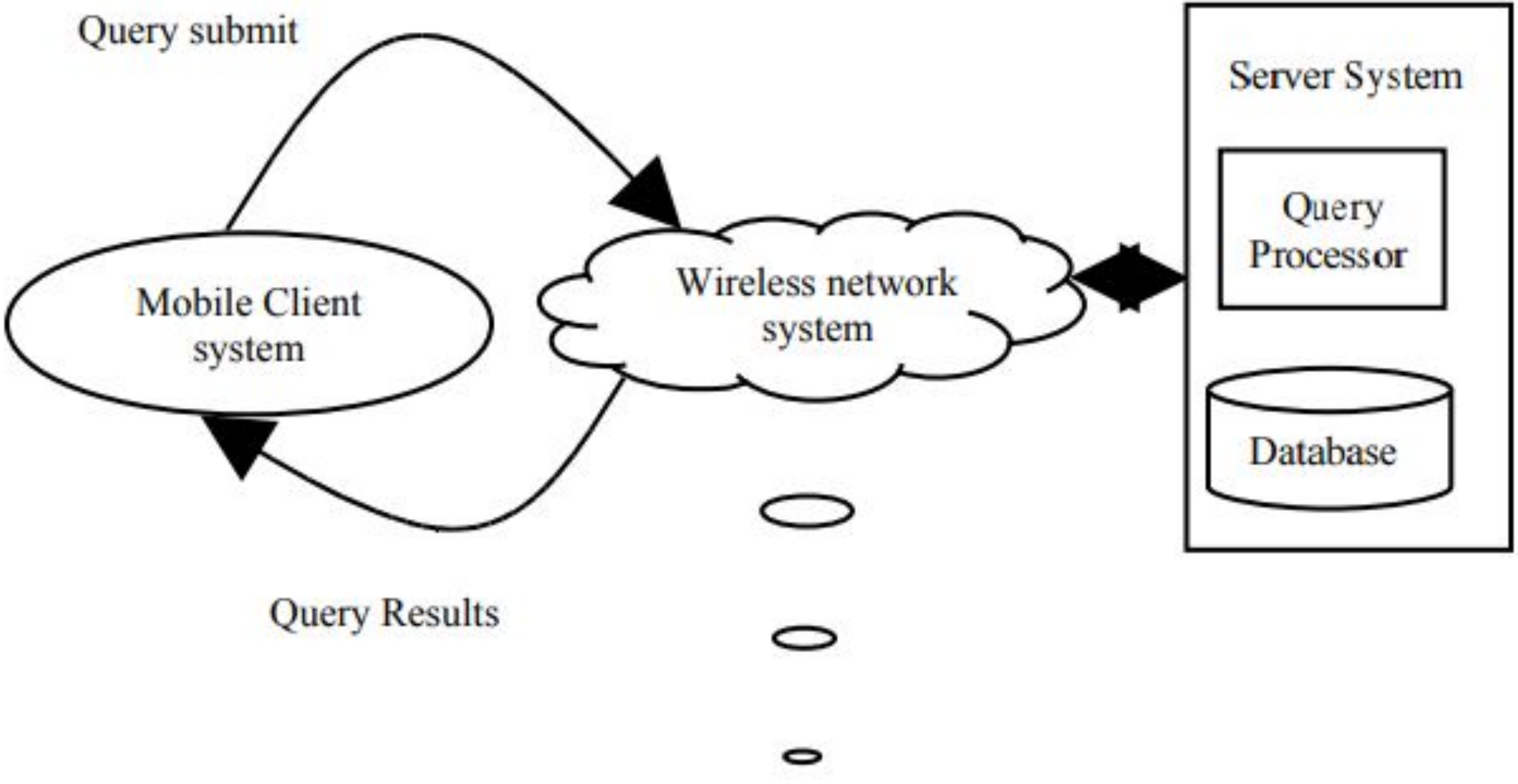


Figure Query Processing in Mobile databases

Mobile Client System

Mobile client system defines a number of strategies to maintain cached data items in client's local storage.

Wireless network System

The Wireless Network System component assumes that all messages are of equal priority that will be served on a First-Come First-Served (FCFS) basis with a service rate of Network Bandwidth.

Server System

Server system considers the problem of pull-based broadcast scheduling where mobile clients send queries to the server through wireless network system, server process the query,and send the result back to the client.

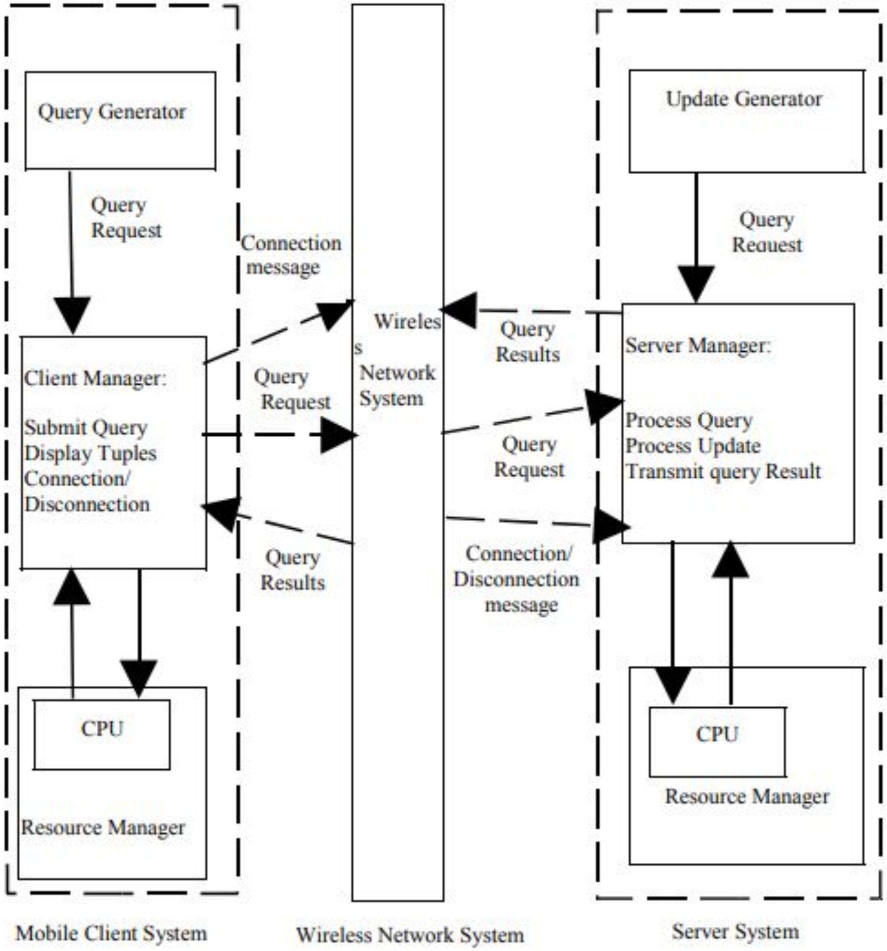


Figure Complete System for Mobile Query Processing

Conclusion:

The new networking technologies allow spontaneous connectivity among mobile devices. Mobile devices can suddenly become both sources and consumers of information. Recent developments in wireless technology enables nomadic people are now able to access email, news, weather, and query to the central database server using wireless devices. Since mobile database is a new dimension of database application, the types of query, query processing system, and communication technology that involves in the application are different than what applies in traditional databases. We have defined role of mobility as well as query processing system in mobile databases.

Thank You!!

Mobile Database System: Role of Mobility on the Query Processing

Article · April 2010

Source: arXiv

2 authors:

Samidha D. Sharma
Saudi Electronic University
2 PUBLICATIONS **9** CITATIONS
SEE PROFILE

Sanjay KASANA Kasana
Sun Pharmaceutical Industries
58 PUBLICATIONS **521** CITATIONS
SEE PROFILE

Mobile Database System: Role of Mobility on the Query Processing

Samidha Dwivedi Sharma[1]
[1]Department Of Computer Science & Application
Dr. H.S. Gour University
Sagar, MP (India)
samidhad2000@gmail.com

D.r R.S.Kasana[2]
[2]Department Of Computer Science & Application
Dr. H.S. Gour University
Sagar, MP (India)
irkasana7158@gmail.com

***Abstract*—The rapidly expanding technology of mobile communication will give mobile users capability of accessing information from anywhere and any time. The wireless technology has made it possible to achieve continuous connectivity in mobile environment. When the query is specified as continuous, the requesting mobile user can obtain continuously changing result. In order to provide accurate and timely outcome to requesting mobile user, the locations of moving object has to be closely monitored. The objective of paper is to discuss the problem related to the role of personal and terminal mobility and query processing in the mobile environment.**

Keywords- Mobile Computing, Mobile Database, Location Management, Location Dependent Data

I. Introduction

Recent advances in hardware technology and wireless communication networks have directed to the emergence of mobile database systems [1,2]. The mobile computing environment provides database applications with useful aspects of wireless technology, which is known as mobile databases. This advance technology has created a new age of nomadic database users. These users are accessing a database through a network. Basically, a user with a wireless connection to the information network does not require maintaining a fixed position in the network.

In mobile environment, elements of the network are very dynamic and can be extremely volatile. Consider a database representing information about moving objects and their position in addition to information about stationary objects [5]. For example, a mobile user looking for a restaurant will obtain different results based on the time and the place he/she issued the query. As the location of other devices changes with respect to other entities and data sources are constantly in movement it may not be possible to collect information about available data sources at any given point of time. As the mobility is the most distinguishing feature of the mobile computing system, location becomes an important piece of information for location-dependent queries [16, 17]. Query may be issued from a moving object (e.g., car of a mobile user) or from a stationary user. Consequently, the answer to a location dependent query may depend on the location of the mobile host (MH) which issued the query and/or the locations of the objects represented in the database. Therefore, an optimal query processing subsystem of a mobile database has to take the strategy used by the location management component into account for answering queries.

The remainder of the paper is organized as follows. In Section 2, we introduce architecture of mobile environment. In Section 3, we describe the role of mobility. Section 4 presents the effect of mobility in mobile environments. Section 5 we study the Query processing system in mobile environment. Finally section 6 concludes the paper.

II. Architeure of Mobile Environment

Figure 1 shows the existing and widely architectural model of a system that supports mobile computing [19,20]. We have added a number of DBSs (database servers) to incorporate database processing of capability without affecting any aspect of the generic mobile network. A set of general purpose computers (PCs, workstations etc) are interconnected through a high speed wired network.

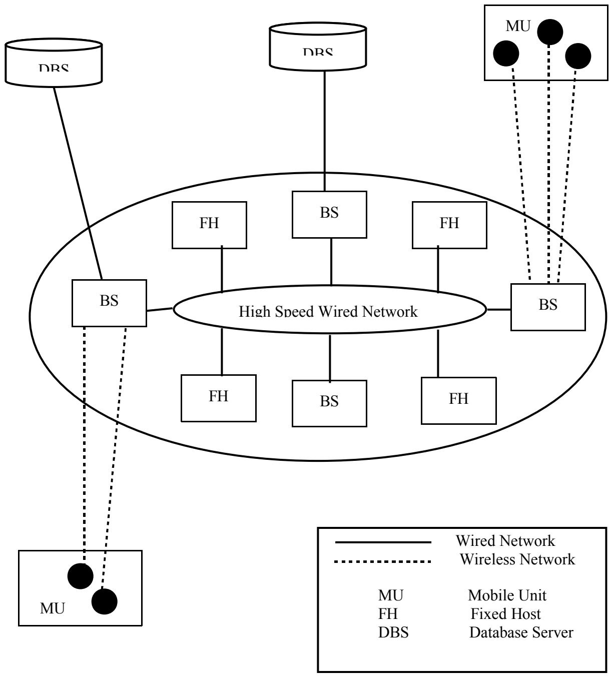


Figure1. Architecture of Mobile Environment

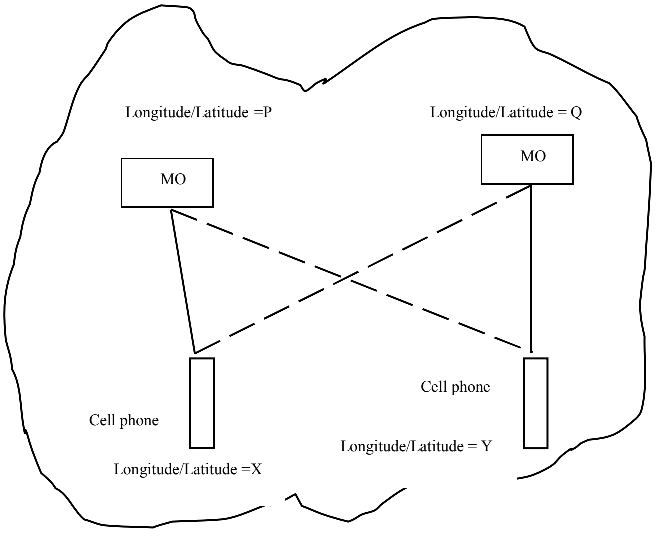
Components in the fixed network are either Fixed Hosts that are not capable of connecting to a mobile unit, or Base Stations which are computers capable of connecting with a mobile unit and are equipped with a database server and wireless interface (they are also known as Mobile Support Stations).A number of mobile computers (laptop, PDAs, etc) referred to as Mobile Hosts (MH) or Mobile units (MU) are connected to wired network components only through BSs via wireless channels. A BS maintains and communicates with its MUs via the wireless interface [4] and has some processing capability. Base Stationed hosts are connected together via a high-speed network (Mbps to Gbps).

III. Role of Mobility

The location of mobile units is an important parameter when locating a mobile station that may hold the required data and when selecting information especially for location dependent information services [15, 18]. In this case the search cost, to locate mobile units, is added to the cost of each communication involving them. A mobile framework is composed of wired and wireless components and human users. The mobile discipline define two types of mobility :(i) terminal mobility and (ii)personal mobility.

A. Terminal Mobility:

In terminal mobility, the connection is established between two points and not between the two persons calling each other. This type of connection in a session allows the use of communication devices to be shared among anybody. It allows a mobile unit (laptop, cell phone, PDA, etc.) to access desired services from any location while in motion or stationary, irrespective of who is carrying the unit. For example a cell phone can be used by its owner and it can also be borrowed by some one else for use. In terminal mobility, it is the responsibility of the wireless network to identify the communication device.



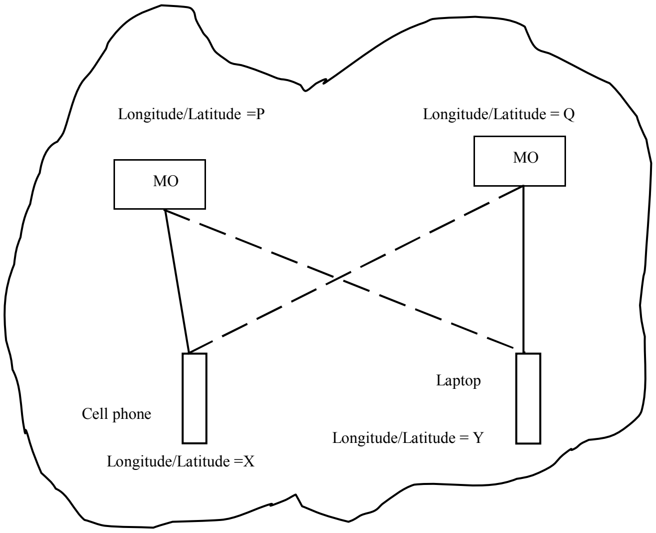
MO-Mobile Object (Car, Bike etc)

Figure 2. Terminal Mobility

Fig 2 illustrates the notion of terminal mobility. A person at location X (Longitude/Latitude = X) uses the mobile unit to communicate with the car driver at location P. He can still establish communication with the driver from a new location Y irrespective of the movement of the car from P to Q. The use of a phone card works on this principle. It can be used from different locations and from different machines such as cell phones, residential phones, etc.

B. Personal Mobility

In personal mobility the mobility of a person is supported. Thus, a user does not have to carry any communication equipment with him; he/she can use any communication device for establishing contact with the other party. This facility requires an identification scheme to verify the person wishing to communicate. Figure 3 illustrates the notion of personal mobility. P person at location X communicates with the car at location P using his PDA, and from location Y also he/she can communicate with the car at location P using his laptop. At present, personal mobility is available through the web. A user can log on to the web from different machines located at different places and access his/her e-mail. The mobile system extends this facility so that the user can use any mobile device for reaching the internet. In personal mobility each person has to be uniquely identified, and one way to do this is via a unique identification number. There is no dependency relationship between terminal and personal mobility; each can exist without the other. In personal mobility the party is free to move, and in terminal mobility the communication unit is free to move. Voice or data communication can be supported by either types of mobility. However, to visualize a complete mobile database management system both types of mobility are essential.



MO-Mobile Object (Car, Bike etc)

Figure 3. Personal Mobility

IV. Effect of mobility on Query processing

The effects of mobility on query processing require that algorithms employed must be capable of managing frequent loss and appearance of mobile device in the network, and that overhead should be minimized during periods of low connectivity [7, 10]. In this environment we can distinguish many characteristics that are listed in the next sub sections.

A. Different Location Models

Location-based applications require a well-formed representation of spatial knowledge. Current location models can be classified into symbolic or geometric models. In geometric model locations are specified as an n-dimensional coordinate (longitude-latitude pair) or a set of coordinates defining an area's bounding geometric shape (such as a polygon). Symbolic models refer to a location by some abstract symbols. Such a representation allows a reference to a place simply by abstract symbol or name, which makes it very convenient for human interaction [14].

B. Different Query Types

In this section, we classify type of queries in mobile databases. The queries can be entirely new and specifically applied in the wireless environment, while the other can be a common type of query in traditional databases. The mobility in a mobile environment introduces three types of entities: (i) mobile client that submits a query, (ii) mobile server that processes a query or a part of it, and (iii) moving object which represents the data targeted by the query. According to these entities queries can be classified into five categories.

1) Non Location Related Query (NLRQ)

If all the predicates and attributes used in a query are non location related then it is called a Non Location Related Query (NLRQ). For example: ”select all restaurants with South India specialty”.

2) Location Dependent Queries (LDQ)

If a query results depend on the location of the query issuer then the query is called Location Dependent Query [8]. For example: ”Find me the closest hotel within 10 miles of my current position”.

3) Location Aware Query (LAQ)

If a query has at least one Location Related simple predicate or one Location Related attribute then it is called Location Aware Query[3]. For example: ”How is the weather in Hyderabad?”.

4) Continuous Query (CQ)

This type of queries includes all queries issued by mobile terminals and querying objects which are themselves moving. For example a query of this type could be : ”Find all the cars within 100 feet of my car”. The result of the query is a set of cars position that varies continuously with the movement of the driver.

5) Ad Hoc Query

Ad Hoc queries are commonly utilized queries in traditional DBMS. This type of query explicitly mentions the required information in the query statement, and does not involve any context awareness information. Thus, the query result is only based on the actual query itself. For example a query of this type could be: ”University student wants to retrieve his/her academic record or personal details.

C. Query Optimization

Query optimization methods try in general to obtain execution plans which minimize CPU, input/output and communication costs. In centralized environments the cost that affects most is the input/output whereas in distributed environments, communication cost is the most important. In a mobile distributed environment, the communication costs are much more difficult to estimate because the mobile host may be situated in different locations. The best site from which to access data depends on where the mobile computer is located. In general, it is not worth calculating plans and their associated costs statically, but rather, dynamic optimization strategies are required in this mobile distributed context.

D. Query Execution

In static systems, query processing execution sites are determined in advance, i.e., which steps are performed on the client and which one on the server. In a mobile environment, where users are moving, such assumption is inadequate. Thus, mobile database systems must be able to choose an execution site for the different phases of query processing depending on their current environment and should be able to revise that decision as flexible as possible.

E. The Impact of Portable Devices Limitations

If we reference dynamic location information in a query, we have to use a location management component to get this information. Thus, depending on the offered localization strategy, we have different possibilities to use this information. The cost evaluation of a query execution plan is guided by required resources of the plan. The main factors that are used in stationary systems are CPU-usage and the number of hard disk access. In mobile systems, additional varying factors like energy consumption, available memory and CPU-speed may be included.

F. The Impact of Wireless Communication

The new networking technologies allow spontaneous connectivity among mobile devices, including hand helds, computers in vehicles, computers embedded in the physical infrastructure, and (nano) sensors. Mobile devices can suddenly become both sources and consumers of information [13]. There is no longer a clean distinction between clients and servers, instead devices are now peers. Furthermore, there is no longer a guarantee of infrastructure support. Consequently, for obtaining data, devices cannot simply depend on a help of some fixed, centralized server. Instead, the devices must be able to cooperate with others in their proximity in order to pursue individual and collective tasks.

V. Query Processing in Mobile Databases

A. Current Work on Query Processing

Query processing deals with designing algorithms that analyze queries and convert them into a series of data manipulation operations. The main function of a query processor is to transform high-level query (typically, in relational calculus) into an equivalent lower-level query (typically, in some variation of relational algebra). Figure 3 shows the classic architecture for query processing. This architecture can be used for any kind of database system including centralized, distributed, or parallel systems. The query processor receives a query as input, translates and optimizes this query in several phases into an executable query plan, and executes the plan in order to obtain the results of the query.

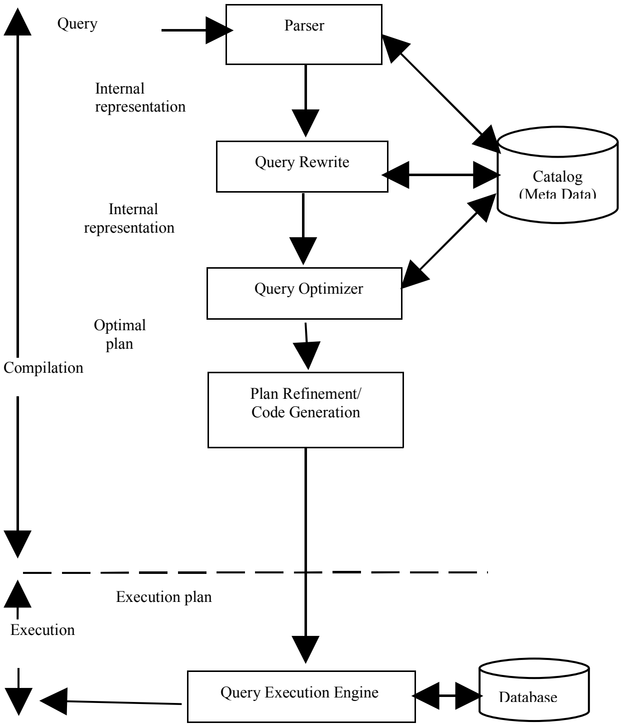


Figure 4. Phase of Query Processing

Not much work has been done and reported on query processing in mobile environment so for. But most of then maintain an approximate information about MUs location which increases the paging cost. MUs are able to access locally unavailable files/data from other remote BSs at the cost of huge message exchange [6, 9, 12].

B. Types of Query Processing Systems

The query processing systems are able to give answer of different types of query by exchanging a less number of messages among the various component of the network. Figure 5 shows the query processing in mobile databases.

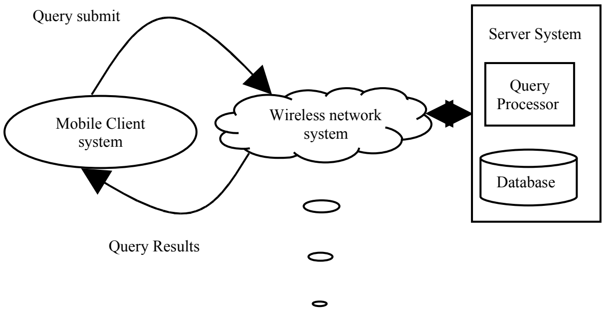


Figure 5. Query Processing in Mobile databases

We divide query-processing system for mobile databases into three parts, namely: (i) mobile client system, (ii) wireless network system and, (iii) server system.

Mobile clients system relates to how client manipulates and maintains the data in its cache efficiently and effectively. Wireless network system communicates data using broadcasting systems [11]. With this system, the number of mobile users does not affect the query performance. Server system relates to designing techniques for the server to accommodate multiple requests so that the request can be processed as efficiently as possible. We say that query processing for mobile databases is very much centered around the issues of caching, broadcasting, and scheduling.

1) Mobile Client System

Each mobile client is composed of three modules as shown in figure 6 a Resource Manager which manage the client CPU for handling the query results, a Client Manager which processes the query requests and passes them to the server, models the disconnection operation, and receives and processes the tuples transmitted from the server and a Query Generator which generates the query requests, and Client queries are submitted from an MH to the server to be processed and a message (messages) containing the tuples that form the answer to the query is (are) transmitted back to the MH. The messages containing the tuples are processed by the MH and the tuples are displayed on the screen of the MH accordingly.

Mobile client system defines a number of strategies to maintain cached data items in client's local storage. Basically, wireless communication channel in general suffers to narrow bandwidth while it is also inherently asymmetric communication, in which the downstream communication bandwidth from a server to the mobile client is much larger than the upstream communication bandwidth from clients back to server. Due to the above reason, caching of frequently accessed data items in a client's local storage becomes important for enhancing the performance and data availability of data access queries. Another advantage of caching is the ability to handle fault tolerance. This is related to the characteristics of mobile computing in which each mobile client connects or disconnects from the network frequently. In some situation, the BS may not be accessible due to problem like signal distortion. However, a query can still be partially processed from caches and at least some of the query results can be returned to the user.

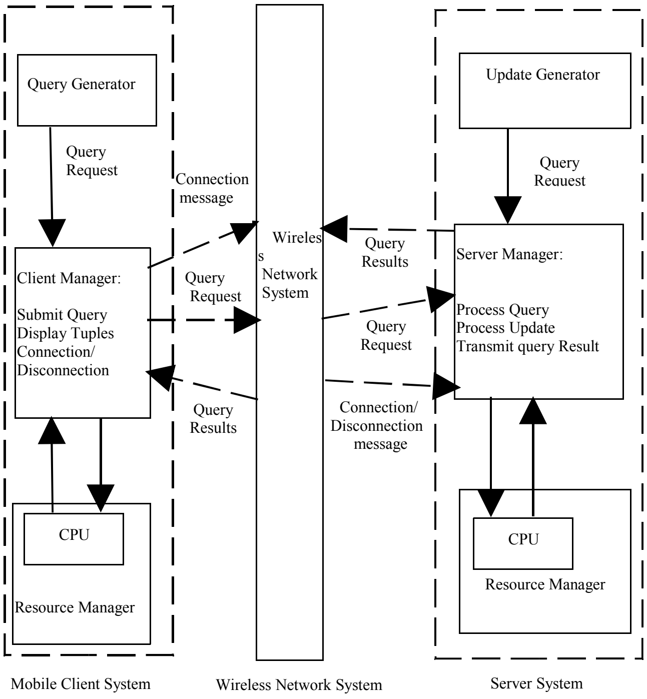


Figure 6. Complete System for Mobile Query Processing

2) Wireless network System

The Wireless Network System component assumes that all messages are of equal priority that will be served on a First-Come First-Served (FCFS) basis with a service rate of Network Bandwidth. When the Wireless Network Manager finds out (i.e., while sending a message to an MH) that an MH is disconnected, it informs the Server Manager about the disconnection so that the transmission of the tuples to the MH can be paused until the MH reconnects to the network. In this scheme, the server also periodically broadcasts the frequently accessed data items to clients through one or more broadcast channels, but the clients may not cache the data item of interest. This situation might occur when the client does not have enough memory or space to cache the required data.

3) Server System

This system considers the problem of pull-based broadcast scheduling where mobile clients send queries to the server through wireless network system, server process the query, and send the result back to the client. The server system has three subsystems as shown in Fig. 6. A Resource Manager which manage the server CPU time for query and update processing, an Update Generator which generates update requests, and a Server Manager which coordinates the query requests from MHs and update requests from the Update Generator.

When a CQ is issued by an MH, it is processed by the Server Manager and the set of tuples satisfying the query are determined. The number of tuples in the answer set of a CQ is determined randomly using a maximum value of Maximum Number of tuples. The Server Manager also decides when and which tuples should be transmitted. The strategy concerns with broadcast and disk scheduling. Broadcast scheduling is to determine how queries to be served efficiently in the server considering a number of factors such as the length of the query, the wait time and the popularity of the items.

A database server is able to enhance the data retrieval performance by incorporating its own main memory and cache to store those database items frequently accessed by most mobile clients. A query can be processed either in the disk server or cache server. If the relevant data items have been retrieved earlier then the query is processed in the cache server. After processing a query, the results are transmitted to the transmitter queue, which subsequently send the data items through the wireless channel.

VI. Conclusion

Recent developments in wireless technology enables nomadic people are now able to access email, news, weather, and query to the central database server using wireless devices. The present work gives response to different types of queries with the help of mobile database stored at BS. Mobile database focuses on the query issue that is the main operation in mobile computing. Since mobile database is a new dimension of database application, the types of query, query processing system, and communication technology that involves in the application are different than what applies in traditional databases. We have defined role of mobility as well as query processing system in mobile databases. Query processing in mobile databases includes mobile client, wireless network, and server system.

The present work may be extended in future by capturing the query arrival pattern in the various BSs of the environment , to minimize the number of message exchanges and delay in getting answer for such query. A secure online mobile database and optimized query processing system needs to be developed to understand this possible as a reality.

VII. References

[1] E.Pitoura, B.Bhargava, Dealing with mobility: issues and research challenges, Technical Report TR-97-070, Department of Computer Sciences, Purdue University, 1993.

[2] P.K.Chrysanthis, Transaction processing in mobile computing environment, in: Proceedings of the IEEE Workshop on Advances in Parallel and Distributed Systems, Princeton, NJ, pages 77-83, October 1993.

[3] H.-E.Kottkamp H.-E. , O.Zukunft, Location-Aware Query Processing in Mobile Database Systems, ACM Symposium on Applied Computing, pages 416-423, February, 1998.

[4] T. Imielinski, B.R.Badrinath, Data management for mobile computing, ACM Sigmod Record 22 (1) pages 34-39, 1993.

[5] T.Imielinski, B.R. Badrinath, Mobile wireless computing: challenges in data management,Communication of ACM 37 (10),pages 18-28, 1994.

[6] T.Imielinski and B.R.Badrinath, Querying in highly mobile distributed environments, In 18th International Conference on Very Large Databases, pages 41-52, 1992

[7] G.H.Forman, J.Zahorjen, The challenges of mobile computing, IEEE Computer 27 (6) (1994).

[8] A.Y.Seydim, M.H.Dunham, and V.Kumar. Location dependent query processing. In *MobiDE*, pages 47–53, 2001.

[9] A.P.Sistla, O.Wolfson, S.Chamberlain, S.Dao, Modeling and querying moving objects, in: Proceedings of the 13th International Conference on Data Engineering, Birmingham, UK, pages 422-432, April 1997.

[10] A.P.Sistla, O.Wolfson, S.Chamberlain, S.Dao, Querying the uncertain position of moving objects in: Temporal Databases: Research and Practice, Lecture Notes in Computer Science, Springer, Berlin, pages 310-337, 1998.

[11] J.W.Wong, Broadcast Delivery, in proceedings of the IEEE, vol.76, no-12,Dec 1998.

[12] O.Ulusoy, Real-time data management for mobile computing, in: Proceedings of the First International Workshop on Issues and Applications of Database Technology, Berlin, Germany, July 1998, pp. 233-240.

[13] Raj Panda,Mobile and Personal Communication Systems and Services, Prentice Hall India,2000

[14] O.Wolfson, P.Sistla, S.Chamberlain, Y.Yesha, Updating and querying databases that track mobile units, Distributed and Parallel Databases Journal 7 (3) , pages 1-31,1993.

[15] O.Wolfson, B.Xu, S.Chamberlain, L.Jiang. Moving objects databases: issues and solutions, in: Proceedings of the 10th International Conference on Scientific and Statistical Database Management, Capri, Italy, pages 111-122, July 1998.

[16] S.Ilarri, E.Mena, and A.Illarramendi, Monitoring Continuous Location Queries Using Mobile Agents, Proc. Sixth East-European Conf. Advances in Databases and Information Systems (ADBIS '02), pages 92-105, Sept. 2002.

[17] H.G.Gok and O. Ulusoy, "Transmission of Continuous Query Results in Mobile Computing Systems," Information Sciences-Applications: An Int'l J., vol. 125, nos. 1-4, pages 37-63, June 2000.

[18] G.Y.Liu and G.Q. Maguire, "Efficient Mobility Management Support for Wireless Data Services," Proc. 45th IEEE Vehicular Technology Conf., July 1995.

[19] T.Imielinski and B.R.Badrinath, Mobile Wireless Computing, Communications of the ACM,37(10), pages 19-28,Oct 1994.

[20] T.Imielinski and H.Korth. Mobile Compurmg. Kluwer Academic Publishers, 1996.

AUTHORS PROFILE

Samidha Dwivedi Sharma is currently a research scholar at Department of Computer Science and Applications, Dr. H. S. Gour Central University (formerly, Sagar University) Sagar, M P, India. She completed her bachelor's degree in Science (B.Sc.) with Mathematics subject in 1992 from Dr. H. S. Gour Central University (formerly, Sagar University) Sagar, M P, India. She received her Master degree in Computer Application in the year 1997 from Barkatullah University, Bhopal, India. Her fields of interests are Database Management Systems, Mobile database, Data Structure and mobile computing. She has published more than 4 research papers. She is a life member of ISTE.

Dr. R. S. Kasana completed his bachelor's degree in 1969 from Meerut University, Meerut, UP, India. He completed his master's degree in Science (M.Sc.-Physics) and master's degree in technology (M. Tech.-Applied Optics) from I.I.T. New Delhi, India. He completed his dorctoral and post doctoral studies from Ujjain University in 1976 in Physics and from P. T. B. Braunschweig and Berlin, Germany & R.D. Univ. Jabalpur correspondingly. He is a senior Professor and HoD of Computer Science and Applications Department of Dr. H. S. Gour University, Sagar, M P, India. During his tenure he has worked as vice chancellor, Dean of Science Faculty, Chairman Board of studies. He has more than 34 years of experience in the field of academics and research. Twelve Ph. D. has awarded under his supervision and more than 110 research articles/ papers have published.