


# National University of Computer and Emerging Sciences, Lahore Campus

	Course Name:	Data Warehousing & Data Mining	Course Code:	CS409
	Program:	BS(CS)	Semester:	Fall 2018
	Duration:	60 Minutes	Total Marks:	30
	Paper Date:	Sat 17-Nov-2018	Weight	12.5%
	Section:	CS	Page(s):	4
	Exam Type:	Midterm 2		

**Student : Name:** \_\_\_\_\_ **Roll No.** \_\_\_\_\_

**Section: CS**

- Instructions/Notes:**
1. Scratch sheet can be used for rough work however, all the questions and steps are to be shown on question paper. No extra/rough sheets should be submitted with question paper.
  2. You will not get any credit if you do not show proper working, reasoning and steps as asked in question statements.
  3. Calculators are ALLOWED.

**Q1. (10 points)**

- a) Assume that 40,000 rows out of 20 million rows in Customer dimension table changes on each data refresh. Which loading strategy should you follow? Explain the reasons for your selection. Also suggest some practical steps that expedite the data loading process.
- b) What are the two most inexpensive data extraction techniques to implement? Explain briefly why?
- c) Explain the difference between destructive merge and constructive merge for applying data to the data warehouse repository. When do you use these modes?

**Ans:**

**a) Incremental data refresh.**

**b) Extraction using Transaction Log, Date & Timestamp, Database Triggers.**

**c) Text Book Ch12 (ETL)**



**Consider the following description for next Questions:**

Consider the following tables and statistics which are part of a student registration system:  
*Student* (RollNo, Name, gpa, DeptID, BatchID, DegreeID, .....); *Attendance* (RollNo, CourseCode,  
Semester, AttFlag, .....);

Assume student and attendance tables containing 128,000 and 1,280,000 rows respectively (Student:Attendance ratio is 1:10). Each table row and each index entry takes 128 bytes and 8 bytes space respectively. Data block size is 8KB and available memory size is 50 blocks. Suppose *degree*= 'MS' has a selectivity of 3%, *batch*= ('2015' or '2014') has a selectivity of (5% + 2%), and *dept*= ('CS or 'EE') has a selectivity of (40% + 20%). Assume cluster (hash based) indexes exist on *RollNo* column of student table and also on *RollNo* column of attendance table. Suppose three secondary (hash based) indexes are also exist on *deptID*, *BatchID*, and *DegreeID* columns of student table separately.

**Q2. (10 points)**

How many data blocks need to be accessed to answer the following query?

```
SELECT COUNT(*) FROM student
WHERE DegreeID='MS' AND (BatchID='2015' OR BatchID='2014') AND (DeptID='CS' OR
DeptID='EE');
```

Examine and use the best possible access path. Justify your selection and show all steps clearly.

**Ans:**

**Best path:** Using combining multiple indexes path (Base table access is not required here):

Combine selectivity is 3% of (7% of (60% of (128,000))) = 162 rows

Index cost: **dept** (CS or EE) 60%= 76,800/1024=75, **batch** (2015 or 2014) 7%= 8960/1024=9, **degree** (MS) 3%= 3840/1024=4,

Total cost (index access cost only) = 75+9+4 = 88 blocks

2- 2<sup>nd</sup> possible path: Using dynamic bitmap indexes path (base table access is required due to false positives)

Total cost (index access cost + base table cost) = 88+162= **250 blocks**

3- 3<sup>rd</sup> possible path: FTS = **2000 blocks**

4- 4<sup>th</sup> possible path: Using single index access with best selectivity (i.e. degree=3%) =

Total cost (index access cost + base table cost) = (3840/1024=4) + 2000= **2004 blocks**

**Static bitmap index cost: (which is not given in question here)**

**One bitmap access cost = 128,000/(1024\*8\*8)= 2 block**

**Total cost (to access 5 bitmaps only) = 10 blocks**

**Q3. (10 points)**

How many data blocks need to be accessed to answer the following query?

```
SELECT RollNo, COUNT(*) AS TotalAttendance FROM student JOIN attendance ON
student.rollno=attendance.rollno
WHERE DegreeID='MS' AND (BatchID='2015' OR BatchID='2014') AND (DeptID='CS' OR
DeptID='EE')
GROUP BY RollNo;
```

You are supposed to filter the condition first and then join. Examine and use the best possible joining technique. Justify your selection and show all steps clearly.

**Ans:**  $R=128$ ;  $R_i=8$ ;  $r_{std}=128,000$ ;  $r_{attn}=1,280,000$ ;  $B=8K$ ;  $K=10$ ;  $bfr=64$ ;  $bfr_i=1024$ ;  $b_{(std.)}=2000$ ;  $b_{(attn.)}=20,000$ ;  $bi_{(std.)}=125$ ;  $bi_{(attn.)}=1250$

As the combine selectivity of student is 3% of (7% of (60% of (128,000))) = 162 rows, so

- **Best option is NLJ (i.e. selectivity is very high): cost**

= student's filter cost + qualifying rows \* (attendance index access cost only; base table access not required)

=  $(88+162) + 162 * (1) = 312$  (to read hash based index access on attendance table only)

=  $(88+162) + 162 * (\log(1280,000/1024)=11) = 2032$  (to read traditional B-tree index access on attendance table only)

=  $(88+162) + 162 * (1+1) = 574$  blocks (for hash based index access and base table access)

=  $(88+162) + 162 * (11+1) = 2194$  blocks (for traditional B-tree index access and base table access)

- **Poor options: HJ/MJ; it is the best case of both, but even though NLJ will be efficient due to highly selectivity query.**

Build input table size after considering qualifying rows is  $162/64=3$  blocks.

**HJ cost** = student's filter cost + hashing cost (using attendance index access only) =  $(88+162) + (3+1250) = 1503$

**HJ cost** = student's filter cost + hashing cost (using attendance table access) =  $(88+162) + (3+20,000) = 20,253$

**Merge Join** cost will be same as of hash join cost because both operand tables are pre-sorted due to clustered indexes on joining column.