

## Physical Database Design (2) (ch. 16 & ch. 6)

Database Management  
COP4540, SCS, FIU

## Clustering Tables

- In some relational DBMSs, related records from different tables can be stored together in the same disk area.
- Data clustering in Oracle:

```
CREATE CLUSTER ORDERING (CLUSTERKEY CHAR(25));
```

```
CREATE TABLE CUSTOMER (  
  CUSTOMER_ID VARCHAR(25) NOT NULL,  
  CUSTOMER_ADDRESS VARCHAR(15) )  
CLUSTER ORDERING (CUSTOMER_ID);
```

```
CREATE TABLE ORDER (  
  ORDER_ID VARCHAR(20) NOT NULL,  
  CUSTOMER_ID VARCHAR(25) NOT NULL,  
  ORDER_DATE DATE )  
CLUSTER ORDERING (CUSTOMER_ID);
```

Database Management  
COP4540, SCS, FIU

## Indexing Commands in Oracle

CUSTOMER(CID, Name, Street, City, ST, ZIP)

- Primary Key Index:

```
CREATE UNIQUE INDEX CUS_INX ON CUSTOMER (CID);
```

- Secondary Key Index:

```
CREATE INDEX ZIP_INX ON CUSTOMER (ZIP);  
CREATE INDEX ST_INX ON CUSTOMER (ST);
```

- Bitmap Index:

```
CREATE BITMAP INDEX CITY_INX ON CUSTOMER (CITY);
```

Database Management  
COP4540, SCS, FIU

## Rules for Using Indexes (1)

- Use on larger tables.
- Index the primary key of each table.
- Index search fields.
- Fields in ORDER BY and GROUP BY commands.
- When there are >100 values but not when there are <30 values.

Database Management  
COP4540, SCS, FIU

## Rules for Using Indexes (2)

- DBMS may have limit on number of indexes per table and number of bytes per indexed field(s).
- Null values will not be referenced from an index.
- Use indexes heavily for non-volatile databases; limit the use of indexes for volatile databases.

Database Management  
COP4540, SCS, FIU

## Rules for Adding Derived Columns

- Use when aggregate values are regularly retrieved and costly to calculate
- Do not permit updating to derived data
- Create triggers to cascade changes from source data.

Database Management  
COP4540, SCS, FIU

## Redundant Array of Inexpensive Disks (RAID)

- To improve file access performance by parallel processing of multiple disks
- Disk Striping: To distribute a sequence of data blocks horizontally on all disks so that the retrieval speed is increased
- There are several ways of striping the data and error correction codes on RAID, leading to RAID-0, -1, -2, -3, -4, and -5.

Database Management  
COP4540, SCS, FIU

## Tuning Queries

- Two indications that suggest that query tuning may be needed
  - A query issues too many disk accesses.
  - The query plan shows that relevant indexes are not being used

Database Management  
COP4540, SCS, FIU

## Tuning Guidelines (1)

- Many query optimizers do not use indexes in the presence of arithmetic expressions (such as  $\text{salary}/365 > 10.5$ ), NULL comparisons (such as  $\text{BDATE IS NULL}$ ), and substring comparisons (such as  $\text{LNAME LIKE \%MANN\%}$ ).
- Indexes are often not used for nested queries using IN.

```
SELECT SSN FROM EMPLOYEE
WHERE DNO IN (SELECT DNUMBER FROM DEPT
              WHERE MGRSSN = '333445555');
```

```
SELECT SSN
FROM EMPLOYEE, DEPT
WHERE DNO = DNUMBER AND DEPT.MGRSSN = '333445555';
```

Database Management  
COP4540, SCS, FIU

## Tuning Guidelines (2)

- Some DISTINCT may be redundant and can be avoided without changing the result.
- Avoid to use unnecessary temporary table. However, in some situation, temporaries are useful.

```
SELECT DISTINCT SSN, NAME, ADDRESS FROM EMPLOYEE

SELECT SSN
FROM EMPLOYEE AS E
WHERE SALARY = (SELECT MAX(SALARY) FROM EMPLOYEE AS M
               WHERE M.DNO = E.DNO);
```

```
SELECT MAX(SALARY) AS HIGHSALARY, DNO INTO TEMP
FROM EMPLOYEE GROUP BY DNO;
SELECT SSN FROM EMPLOYEE, TEMP WHERE SALARY = HIGHSALARY
AND EMPLOYEE.DNO = TEMP.DNO;
DROP TABLE TEMP;
```

Database Management  
COP4540, SCS, FIU

## Tuning Guidelines (3)

- If multiple options for join condition are possible, choose the one that uses a clustering index and avoid that contain string comparison.
- Sometimes the order of the table in the from clause may affect the join processing.
- Some query optimizer perform worse on nested queries compared to their equivalent unnested counterparts.
  - Embedded SELECT blocks using IN, = ALL, and = SOME may be replaced by joins.
- Usually, it is good to transform NOT condition into a positive expression.

Database Management  
COP4540, SCS, FIU

## Tuning Guidelines (4)

- If an equality join is set up between two tables, the rang predicate on the joining attribute set up in one table may be repeated for the other table.
- Query optimizer may not use indexes on OR condition.

```
SELECT FNAME, LNAME SALARY, AGE
FROM EMPLOYEE
WHERE AGE > 45 OR SALARY < 5000;
```

```
SELECT FNAME, LNAME, SALARY, AGE
FROM EMPLOYEE
WHERE AGE > 45
OR
SELECT FNAME, LNAME, SALARY, AGE
FROM EMPLOYEE
WHERE SALARY < 5000
```

Database Management  
COP4540, SCS, FIU

## Tuning Guidelines (4)

- WHERE conditions may be rewritten to utilize the indexes on multiple columns.

```
SELECT RegionNo, Prod_type, Month, Sales
FROM SALES_STATICS
WHERE RegionNo = 3 AND ( Prod_type BETWEEN 1 AND 3) OR
(Prod_type BETWEEN 8 AND 10));
```

```
SELECT RegionNo, Prod_type, Month, Sales
FROM SALES_STATICS
WHERE RegionNo = 3 AND (Prod_type BETWEEN 1 AND 3) OR
      RegionNo = 3 AND (Prod_type BETWEEN 8 AND 10);
```

Note: if there is a composite index defined on (RegionNo, Prod\_type)

*Database Management*  
COP4540, SCS, FIU