# CS614 – Date Warehousing
## Reference Short Notes for Mid Term Papers

Muhammad Faisal Dar
MIT 4th Semester
faisalgrw123@gmail.com

**Difference between OLTP & DWH** **(page21)**

| OLTP | DWH |
|---|---|
| Primary key used | Primary key NOT used |
| No concept of Primary Index | Primary index used |
| May use a single table | Uses multiple tables |
| Few rows returned | Many rows returned |
| High selectivity of query | Low selectivity of query |
| Indexing on primary key (unique) | Indexing on primary index (non-unique) |

**What is a DWH?** **(page9)**

A complete repository of historical corporate data extracted from transaction systems that is available for ad-hoc access by knowledge workers

**Quantifying size of data** **(page 6)**

| | | |
|---|---|---|
| 1 MB | $2^{20}$ or $10^{6}$ bytes | Small novel – 31/2 Disk |
| 1 GB | $2^{30}$ or $10^{9}$ bytes | Paper rims that could fill the back of a pickup van |
| 1 TB | $2^{40}$ or $10^{12}$ bytes | 50,000 trees chopped and converted into paper and printed |
| 2 PB | 1 PB = $2^{50}$ or $10^{15}$ bytes | Academic research libraries across the U.S. |
| 5 EB | 1 EB = $2^{60}$ or $10^{18}$ bytes | All words ever spoken by human beings |

**Typical Applications of DWH** **(page27)**

There are, and there can be many applications of a data warehouse. It is not possible to discuss all of them. Some representative applications are listed to be discussed as follows:

- Fraud detection
- Profitability analysis
- Direct mail/database marketing
- Credit risk prediction
- Customer retention modeling
- Yield management
- Inventory management

**Fraud detection** **(page28)**

- By observing data usage patterns
- People have typical purchase patterns
- Deviation from patterns
- Certain cities notorious for fraud
- Certain items bought by stolen cards
- Similar behavior for stolen phone cards

**What are the goals of normalization?** **(page32)**

- Eliminate redundant data.
- Ensure data dependencies make sense.

**Five principal De-normalization Techniques** **(page43)**

1) Collapsing Tables.
    - Two entities with a One-to-One relationship.
    - Two entities with a Many- to-Many relationship.
2) 2. Pre-Joining.
3) 3. Splitting Tables (Horizontal/Vertical Splitting).
4) 4. Adding Redundant Columns (Reference Data).
5) 5. Derived Attributes (Summary, Total, Balance etc).

**Splitting Tables: Horizontal splitting** **(page46)**

**ADVANTAGE**

- Enhance security of data.
- Organizing tables differently for different queries.
- Reduced I/O overhead.
- Graceful degradation of database in case of table damage.
- Fewer rows result in flatter B- trees and fast data retrieval.

**Issues of De-Normalization** **(page53)**

- Storage
- Performance
- Maintenance
- Ease-of-use

**OLAP: Facts & Dimensions** **(page65)**

❖ FACTS: Quantitative values (numbers) or "measures."
e.g., units sold, sales $, Co, Kg etc.

❖ DIMENSIONS: Descriptive categories.
e.g., time, geography, product etc.

**OLAP FASMI Test** **(page67)**

**Fast:** Delivers information to the user at a fairly constant rate i.e. O(1) time. Most queries answered in less than 5 seconds.

**Analysis:** Performs basic numerical and statistical analysis of the data, pre-defined by an application developer or defined ad-hocly by the user.

**Shared:** Implements the security requirements necessary for sharing potentially confidential data across a large user population.

**Multi-dimensional:** The essential characteristic of OLAP.

**Information:** Accesses all the data and information necessary and relevant for the application, wherever it may reside and not limited by volume.

**OLAP Implementations** **(page69)**

1. **MOLAP: OLAP** implemented with a multi-dimensional data structure.
2. **ROLAP: OLAP** implemented with a relational database.
3. **HOLAP: OLAP** implemented as a hybrid of MOLAP and ROLAP.
4. **DOLAP: OLAP** implemented for desktop decision support environments.

**Cube Operations** **(page71)**

- Rollup: summarize data
  e.g., given sales data, summarize sales for last year by product category and region

- Drill down: get more details

e.g., given summarized sales as above, find breakup of sales by city within each region, or within Sindh

- Slice and dice: select and project
  e.g.: Sales of soft -drinks in Karachi during last quarter

- Pivot: change the view of data

**Advantages of MOLAP** **(page74)**

- Instant response (pre-calculated aggregates).
- Impossible to ask question without an answer.
- Value added functions (ranking, % change).

**MOLAP Implementation Issues** **(page75)**

**Maintenance issue:** Every data item received must be aggregated into every cube (assuming "to-date" summaries are maintained). Lot of work

**Storage issue:** As dimensions get less detailed (e.g., year vs. day) cubes get much smaller, but storage consequences for building hundreds of cubes can be significant. Lot of space

**Scalability:** Often have difficulty scaling when the size of dimensions becomes large. The breakpoint is typically around 64,000 cardinality of a dimension.

**Partitioned Cubes** **(page76)**

- To overcome the space limitation of MOLAP, the cube is partitioned.
- One logical cube of data can be spread across multiple physical cubes on separate (or same) servers.
- The divide &conquer cube partitioning approach helps alleviate the scalability limitations of MOLAP implementation.
- Ideal cube partitioning is completely invisible to end users.
- Performance degradation does occur in case of a join across partitioned cubes.

**Virtual Cubes** **(page77)**

Used to query two dissimilar cubes by creating a third "virtual" cube by a join between two cubes.

- Logically similar to a relational view i.e. linking two (or more) cubes along common dimension(s).
- Biggest advantage is saving in space by eliminating storage of redundant information.

## Why ROLAP? (page78)

**Issue of scalability i.e. curse of dimensionality for MOLAP**

- Deployment of significantly large dimension tables as compared to MOLAP using secondary storage
- Aggregate awareness allows using pre -built summary tables by some front -end tools.
- Star schema designs usually used to facilitate ROLAP querying (in next lecture).

**ROLAP Issues** **(page82)**

- Maintenance.
- Non standard hierarchy of dimensions.
- Non standard conventions.
- Explosion of storage space requirement.
- Aggregation pit-falls.

**HOLAP** **(page87)**

- Target is to get the best of both worlds.

- HOLAP (Hybrid OLAP) allows co-existence of pre-built MOLAP cubes alongside relational OLAP or ROLAP structures.

- How much to pre -build?

**How to simplify an ER data model?** **(page94)**

Two general methods:

- De-Normalization
- Dimensional Modeling (DM)

**The Process of Dimensional Modeling** **(page99)**
Four Step Method from ER to DM

1. Choose the Business Process
2. Choose the Grain
3. Choose the Facts
4. Choose the Dimensions

**Classification of Aggregation Functions** **(page108)**
How hard to compute aggregate from sub-aggregates?

Three classes of aggregates:

(1) Distributive
(2) Algebraic
(3) Holistic

**Not recording Facts** **(page109)**

Transactional fact tables don't have records for events that don't occur

This has both advantage and disadvantage.

**Advantage:**

- Benefit of sparsity of data
- Significantly less data to store for "rare" events

**Disadvantage**

- Lack of information

**Handling Multi-valued Dimensions?** **(page110)**

One of the following approaches is adopted:

- Drop the dimension.
- Use a primary value as a single value.
- Add multiple values in the dimension table.
- Use "Helper" tables.

## Step-4: Pros and Cons of Handling (page 115)

**Option-1: Overwrite existing value**

- Simple to implement
- No tracking of history

**Option-2: Add a new dimension row**

- Accurate historical reporting
- Pre-computed aggregates unaffected
- Dimension table grows over time

**Option-3: Add a new field**

- Accurate historical reporting to last TWO changes
- Record keys are unaffected
- Dimension table size increases

## Types of Data Extraction (page120)

**Logical Extraction**

- Full Extraction
- Incremental Extraction

**Physical Extraction**

- Online Extraction
- Offline Extraction
- Legacy vs. OLTP

## Logical Data Extraction (page 121)

### Full Extraction

- The data extracted completely from the source system.
- No need to keep track of changes.
- Source data made available as-is w/o any additional information.

### Incremental Extraction

- Data extracted after a well defined point/event in time.
- Mechanism used to reflect/record the temporal changes in data (column or table).
- Sometimes entire tables off-loaded from source system into the DWH.
- Can have significant performance impacts on the data warehouse server.

## Physical Data Extraction (page 122)

### Online Extraction

- Data extracted directly from the source system.
- May access source tables through an intermediate system.
- Intermediate system usually similar to the source system.

### Offline Extraction

- Data NOT extracted directly from the source system, instead staged explicitly outside the original source system.
- Data is either already structured or was created by an extraction routine.

## Data Transformation (page 123)

**Basic tasks**

I. Selection

II. Splitting/Joining

III. Conversion

IV. Summarization

V. Enrichment

**Three Loading Strategies (page 127)**

- Once we have transformed data, there are three primary loading strategies:
- Full data refresh with BLOCK INSERT or 'block slamming' into empty table.
- Incremental data refresh with BLOCK INSERT or 'block slamming' into existing (populated) tables.
- Trickle/continuous feed with constant data collection and loading using row level insert and update operations.

**Why ETL Issues? (page 128)**

- Things would have been simpler in the presence of operational systems, but that is not always the case

- Manual data collection and entry. Nothing wrong with that, but potential to introduces lots of problems

- Data is never perfect. The cost of perfection, extremely high vs. its value.

**"Some" Issues**

- Usually, if not always underestimated
- Diversity in source systems and platforms
- Inconsistent data representations
- Complexity of transformations
- Rigidity and unavailability of legacy systems
- Volume of legacy data
- Web scrapping

**Rigidity and unavailability of legacy systems** **(page 132)**

- Very difficult to add logic to or increase performance of legacy systems.
- Utilization of expensive legacy systems is optimized.
- Therefore, want to off-load transformation cycles to open systems environment.
- This often requires new skill sets.
- Need efficient and easy way to deal with incompatible mainframe data formats.

**Volume of legacy data** **(page 133)**

- ❖ Talking about not weekly data, but data spread over years.
- ❖ Historical data on tapes that are serial and very slow to mount etc.
- ❖ Need lots of processing and I/O to effectively handle large data volumes.
- ❖ Need efficient interconnect bandwidth to transfer large amounts of data from legacy sources to DWH.

## Web scrapping **(page 134)**

Lot of data in a web page, but is mixed with a lot of "junk".

**Web scrapping Problems:**

- Limited query interfaces

  - Fill in forms

- "Free text" fields

  - E.g. addresses

- Inconsistent output

  - i.e., html tags which mark interesting fields might be different on different pages.

- Rapid change without notice.

**Beware of data quality (or lack of it)** **(page 135)**

- Data quality is always worse than expected.
- Will have a couple of lectures on data quality and its management.
- It is not a matter of few hundred rows.
- Data recorded for running operations is not usually good enough for decision support.

## ETL vs. ELT (page 135)

There are two fundamental approaches to data acquisition:

**ETL:**
Extract, Transform, Load in which data transformation takes place on a separate transformation server.

**ELT:**
Extract, Load, Transform in which data transformation takes place on the data warehouse server.

**Lighter Side of Dirty Data** **(page146)**

- Year of birth 1995 current year 2005
- Born in 1986 hired in 1985
- Who would take it seriously? Computers while summarizing, aggregating, populating etc.
- Small discrepancies become irrelevant for large averages, but what about sums, medians, maximum, minimum etc.?

**Serious Problems due to dirty data** **(page 147)**

> Decisions taken at government level using wrong data resulting in undesirable results.
> In direct mail marketing sending letters to wrong addresses loss of money and bad reputation.

## 3 Classes of Anomalies (page 148)

**Syntactically Dirty Data**

- Lexical Errors
- Irregularities

**Semantically Dirty Data**

- Integrity Constraint Violation
- Business rule contradiction
- Duplication

**Coverage Anomalies**

- Missing Attributes
- Missing Records

**Handling missing data** **(page 150)**

- Dropping records.
- "Manually" filling missing values.
- Using a global constant as filler.
- Using the attribute mean (or median) as filler.
- Using the most probable value as filler.

## Key Based Classification of Problems **(page 150)**

- Primary key problems
- Non-Primary key problems

**Primary key problems**

1. Same PK but different data.
2. Same entity with different keys.
3. PK in one system but not in other.
4. Same PK but in different formats.

**Non primary key problems**

1. Different encoding in different sources.
2. Multiple ways to represent the same information.
3. Sources might contain invalid data.
4. Two fields with different data but same name.

**Automatic Data Cleansing** **(page 152)**

1) Statistical
2) Pattern Based
3) Clustering
4) Association Rules

**Why data duplicated?** **(page 153)**

A data warehouse is created from heterogeneous sources, with heterogeneous databases (different schema/representation) of the same entity.

The data coming from outside the organization owning the DWH can have even lower quality data i.e. different representation for same entity, transcription or typographical errors.

**Problems due to data duplication** **(page 153)**

Data duplication can result in costly errors, such as:

- False frequency distributions.
- Incorrect aggregates due to double counting.
- Difficulty with catching fabricated identities by credit card companies.

**Need & Tool Support** **(page 156)**

> Logical solution to dirty data is to clean it in some way.
> Doing it manually is very slow and prone to errors.
> Tools are required to do it "cost" effectively to achieve reasonable quality.
> Tools are there, some for specific fields, others for specific cleaning phase.
> Since application specific, so work very well, but need support from other tools for broad spectrum of cleaning problems.

## Basic Sorted Neighborhood (BSN) Method (page 157)

### Steps 1: Create Keys

- Compute a key for each record in the list by extracting relevant fields or portions of fields
- Effectiveness of the this method highly depends on a properly chosen key

### Step 2: Sort Data

- Sort the records in the data list using the key of step 1

### Step 3: Merge

- Move a fixed size window through the sequential list of records limiting the comparisons for matching records to those records in the window
- If the size of the window is w records then every new record entering the window is compared with the previous w-1 records.

## BSN Method: Selection of Keys (page 159)

### Selection of Keys

- Effectiveness highly dependent on the key selected to sort the records middle name vs. last name,
- A key is a sequence of a subset of attributes or sub -strings within the attributes chosen from the record.
- The keys are used for sorting the entire dataset with the intention that matched candidates will appear close to each other.

### BSN Method: Problem with keys (page160)

- Since data is dirty, so keys WILL also be dirty, and matching records will not come together.

- Data becomes dirty due to data entry errors or use of abbreviations.

- Solution is to use external standard source files to validate the data and resolve any data conflicts.

**BSN Method: Equational Theory** **(page 162)**

To specify the inferences we need equational Theory.

- Logic is NOT based on string equivalence.
- Logic based on domain equivalence.
- Requires declarative rule language.

## Limitations of BSN Method **(page 164)**

**BSN Method Limitations**

- No single key is sufficient to catch all matching records.

- Fields that appear first in the key have higher discriminating power than those appearing after them.

- If NID number is first attribute in the key 81684854432 and 18684854432 are highly likely to fall in windows that are far apart.

**Possible Modifications to BSN Method**

- Increase w, the size of window
- Multiple passes over the data set

## Orr's Laws of Data Quality **(page 169)**

**Law #1 -** "Data that is not used cannot be correct!"

**Law #2 -** "Data quality is a function of its use, not its collection!"

**Law #3 -** "Data will be no better than its most stringent use!"

**Law #4 -** "Data quality problems increase with the age of the system!"

**Law #5 –** "The less likely something is to occur, the more traumatic it will be when it happens!"

**Total Quality Management (TQM)** **(page 169)**

TQM approach is advocating the involvement of all employees in the continuous improvement process, the ultimate goal being the customer satisfaction.

## Cost of Data Quality Defects (page 170)

**Controllable Costs**

- Recurring costs for analyzing, correcting, and preventing data errors

**Resultant Costs**

- Internal and external failure costs of business opportunities missed.

**Equipment & Training Costs**

**Characteristics or Dimensions of Data Quality** **(page 172)**

- Accuracy
- Completeness
- Consistency
- Timeliness
- Timeliness
- Uniqueness
- Interpretability
- Interpretability
- Interpretability

## Data Quality Assessment Techniques (page 173)

- Ratios
- Min-Max

**Simple Ratios**

- Free-of-error
- Completeness
- Consistency

**Min-Max**

- Believability
- Appropriate Amount of Data

**Min-Max**

- Timeliness
- Accessibility

**Data Quality Validation Techniques** **(page 176)**

- Referential Integrity (RI).
- Attribute domain.
- Using Data Quality Rules.
- Data Histograming.

**3 steps of Attribute Domain Validation** **(page 177)**

- The occurrences of each domain value within each coded attribute of the database.
- Actual content of attributes against set of valid values.
- Exceptions to determine cause and impact of the data quality defects.

----------Wish U Best of L|U|C|K for EXAMS ----------