ETL

It is used to decission System.

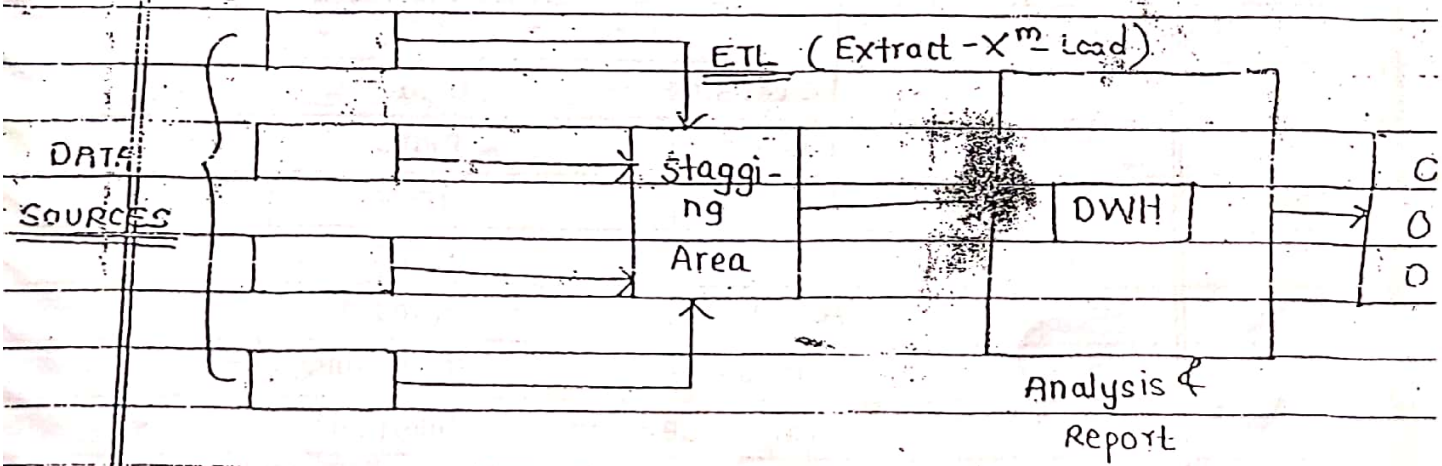
* DWH. – For Analysis & reporting purpose.

Defⁿ → DWH is a dB designed for query & analysis rather than transaction processing.

1. It contains historical data derived from transactional data (operational/current)

2. DWH seperates analysis workload from transactional workload and enables to consolidate from various DB (sources)

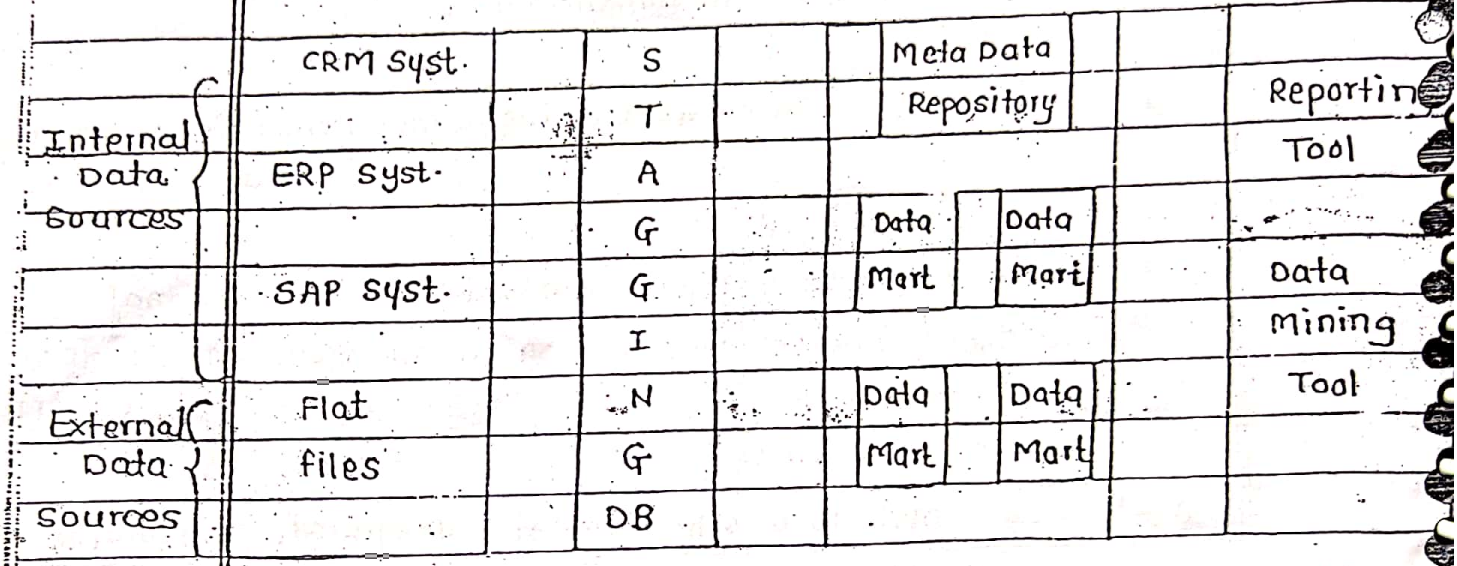
Defⁿ → DWH is a subj. oriented, integrated, nonvolatile, time varying collⁿ of data in support of management decⁿ making process.



* Data Ware House
Business Ware House
Decⁿ support syst
Business Intelligence solⁿ
} various names.

Date 17-8-

ARCHITECTURE OF DWH →



Data Source Layer	Staggling Area	DWH Layer	Front Tier
	↓	↓	
	Cleansing	Data	
	Filtering	Mart	
	ordering	↓	
	↓		
	For storing	Stores	
	Temporary	subjectwise	
	database.	informatn.	

1. 1st Layer — Data source Layer.

— which refers to various data sources in multiple format like relational dB, flatfiles & others.

— contains Transactional / current / operational data.

2. 2nd Layer — Stagging Area

— Intermediate area stage betn source to target where required business rules [Transformn] are applied.

— This layer takes care of data processing methods like
  1. Data cleansing
  2. Filtering
  3. Merging
  4. Splitting, etc.

  } To avoid duplicate Data.

— After this all data put it into DWH
i.e. DWH Layer / Data store / Data Meta Repository.

3. 3rd Layer — DWH Layer

— In this layer :- cleaned
Integrated
Transformed
ordered data is present in multidimensional environment

4. 4th Layer — Front Tier

— In front tier data in DWH layer is used for reporting analysis with the help of reporting & data minning tool.

* Stagging Area →

— Also called as landing zone.

— Intermediate storage Area used for data processing during ETL process.

* Need of stagging Area →

— In the abscence of stagging area data load will have to go from OLTP to OLAP syst. directly.

— This can hampers the performance of OLTP syst.

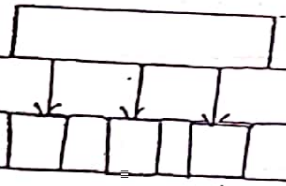
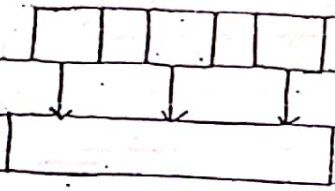
— This is the primary reason of existance of stagging area.

* Data Mart →

— is a subset of DWH which is limited to specific fun area (subj. area) or group of users.

It is a condensed version of DWH focussed on single fun area at a time.

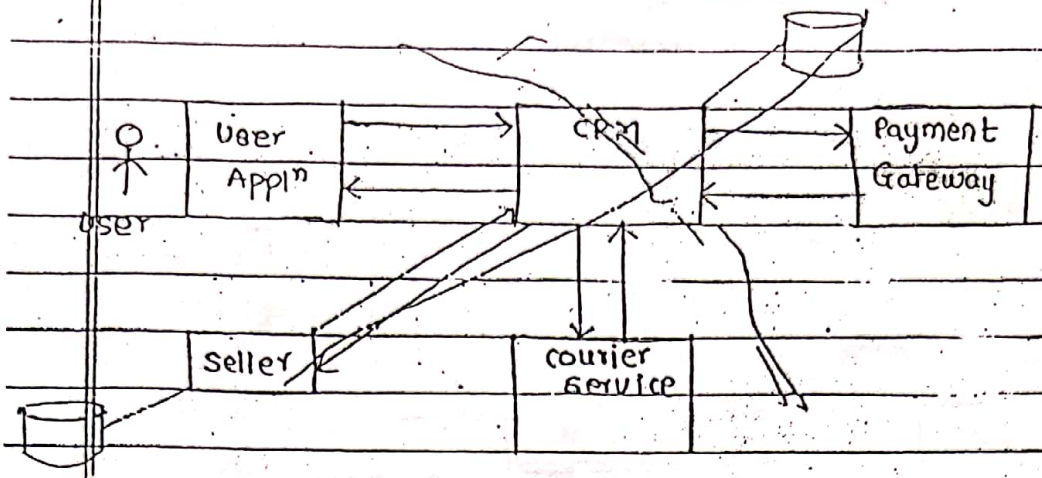
2 Approaches → Top Down
Bottom Up

Top Down	Bottom Up
	
– Data Mart can be build later to form DWH.	– Data Mart can be build before or in parallel with DWH.
– High cost	– Low cost
– Less Time consuming.	– Less Time consuming.
– More Planning & designing.	– Less Planning & designing.

Data Ware Housing	Data Mart
1. Union of all data.	1. Data of single subj. area.
2. Implementⁿ – Time consuming.	2. Implementⁿ – less time consuming.
3. Enterprise view ↓ No. of users more	3. Departmental view ↓ No. of users ...

Date 18-8-18
Page

* OLTP VS OLAP —



I. OLTP :- online Transact^{n} Processing.

— OLTP DB → detailed and current data is stored, of transactional syst. Mostly data is in 3NF form.

— It is mainely used for data processing not for analysis.

II. OLAP :- online Analytical Processing.

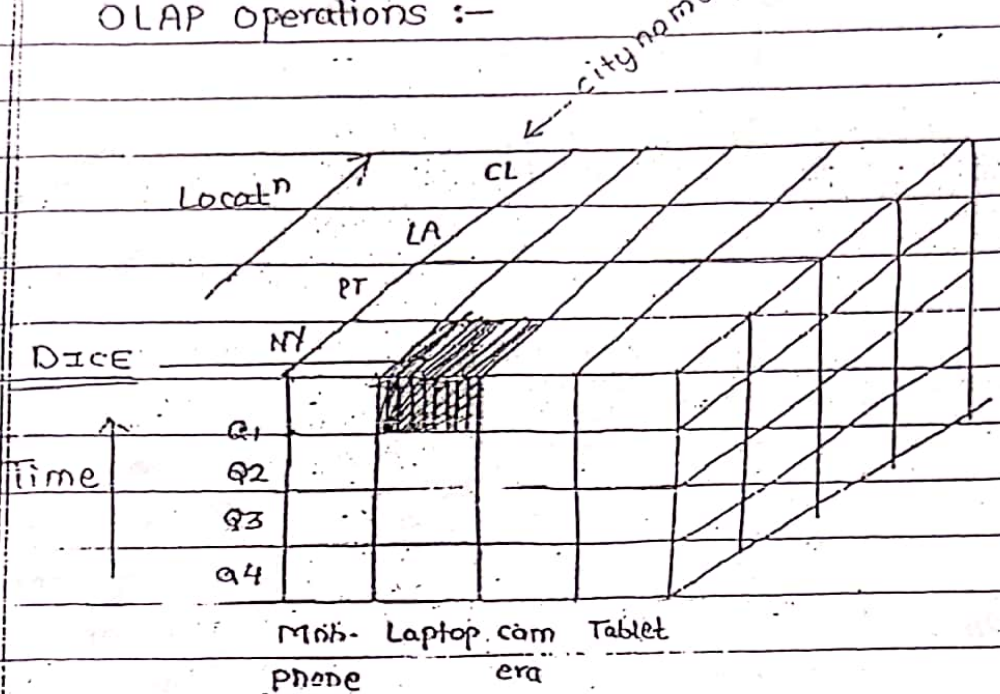
— It allows user to analyze info. from multiple DB syst. at once.

— mainely used for data analysis not for processing.

— It provides single platform for all platforms of business needs which include
↓ planning, designing, analyzing info & reporting.

OLTP	OLAP
1. Used for Transactn processing.	1. Used for Query & Analysis.
2. Data 1) current data/operational. 2) Data is in normalised form. 3) Stores all data. 4) Volatile data.	2. Data 1) Historical data / sh 2) Data is in denormalised form. 3) stores only relevant data. 4) Nonvolatile data.
3. To run fundamental business task.	3. To help with planning prob. solving, & decn support.
4. No. of users are more.	4. comparatively less users.
5. Applⁿ driven.	5. Analysis driven.
6. Used to store the data into dB.	6. Used for reading data from DWH.

OLAP Operations :-



operations —

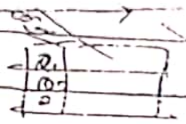
Or navigate data from less detail to high de

1. Drill Down. — Add dimension from data cube.
2. Roll up — Remove dimension from -11-
3. Dice
4. Slice
5. Pivot

1. Drill Down :- physically data is present in quarter wise user want it month wise

2. Roll up :- physically data present city wise user want region wise.

3. Dice → User wants Particular data e.g. User wants data (laptop) in 1st quarter

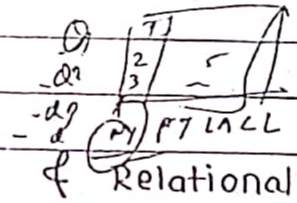


4. Slice → Detailing data in 2D. 3D → 2D

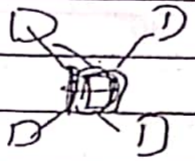
Q1				
Q2				
Q3				
Q4				
	NY	PT	LA	CL

5. Pivot → Rows becomes column & column only becomes rows.

NY	1	2	3	4
PT				
LA				
CL				



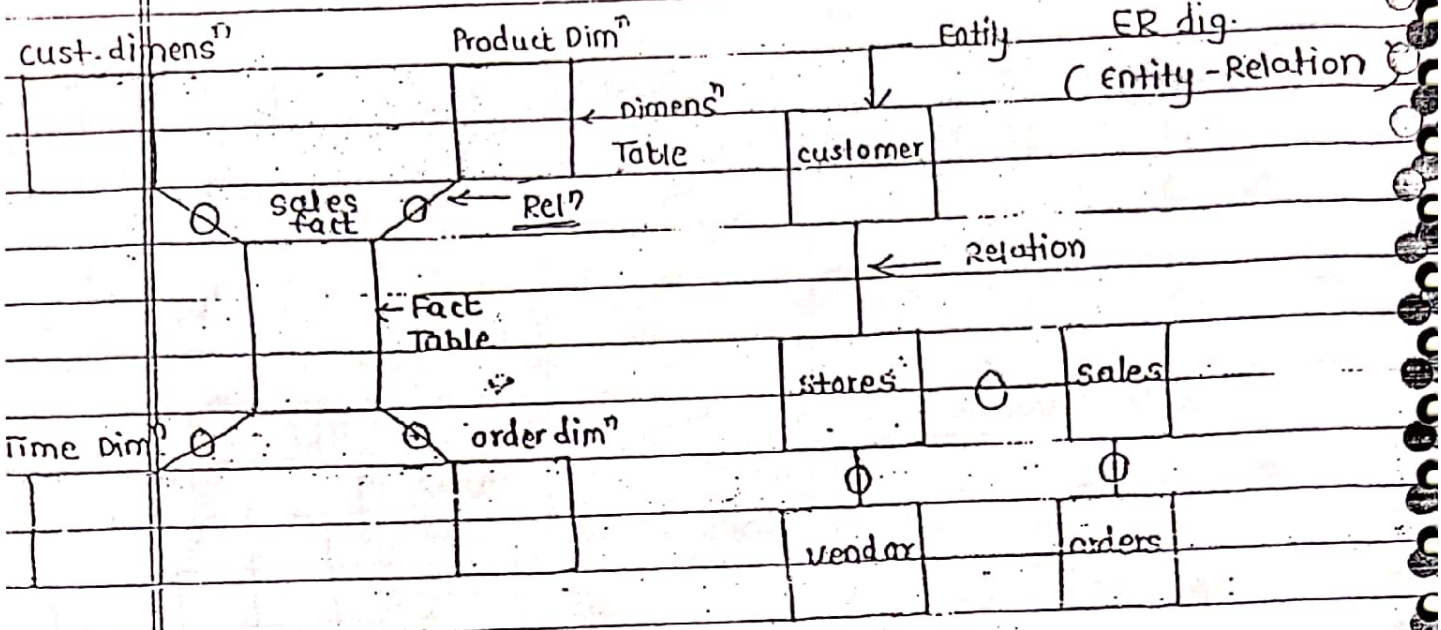
* Dimensional Model & Relational Model →



Dimensional	Relational
1. Design of data for business processing. (Analysis)	1. Design of data for Data processing.
2. Captures the fact along with their dimens^{n}.	2. Removes the data redundancy & ensures the data consistency & integrity.

Dimensional

Relational



* Design : model →

1. conceptual
2. logical
3. Physical

1. conceptual Model —

— i) Important entities & their relation

↓

— 2) No attributes, No datatype (defined) in entity defined

— 3) High level design of dB

2 Retrieving ... diff data

Regretion --

2. Logical Model — 1) It defines the data as much as possible to show & how they can be physically implemented in dB.

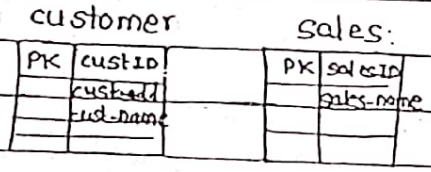
↓

2) Imp. entities & rel^n & their possible Attributes

↓

3) Primary key of each entity is specified.

4) Foreign key may be specified.



3. Physical Model — 1) It defines how the models are physically exists in syst.

2) Displays all the required constraints

3) Data types — also shown.

↓

like int

varchar2 (20)

* Diff. betⁿ —

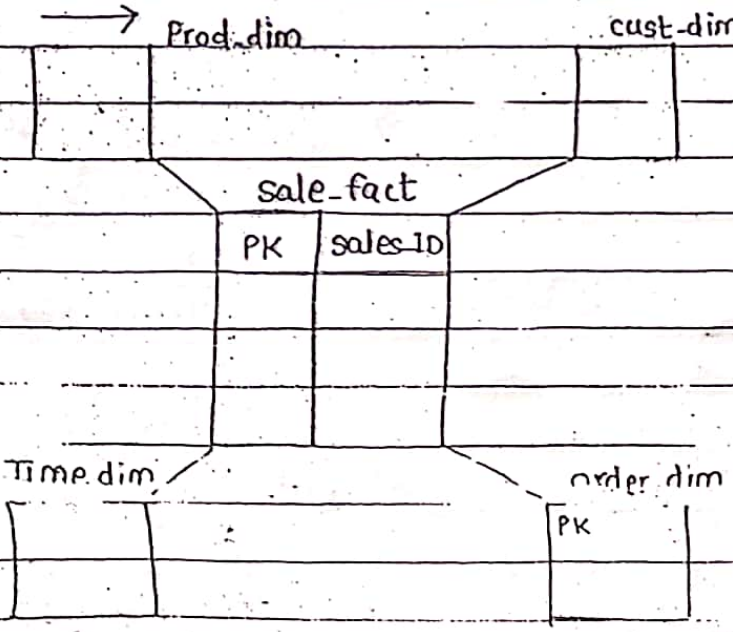
— conceptual model & Logical Model & physical Model

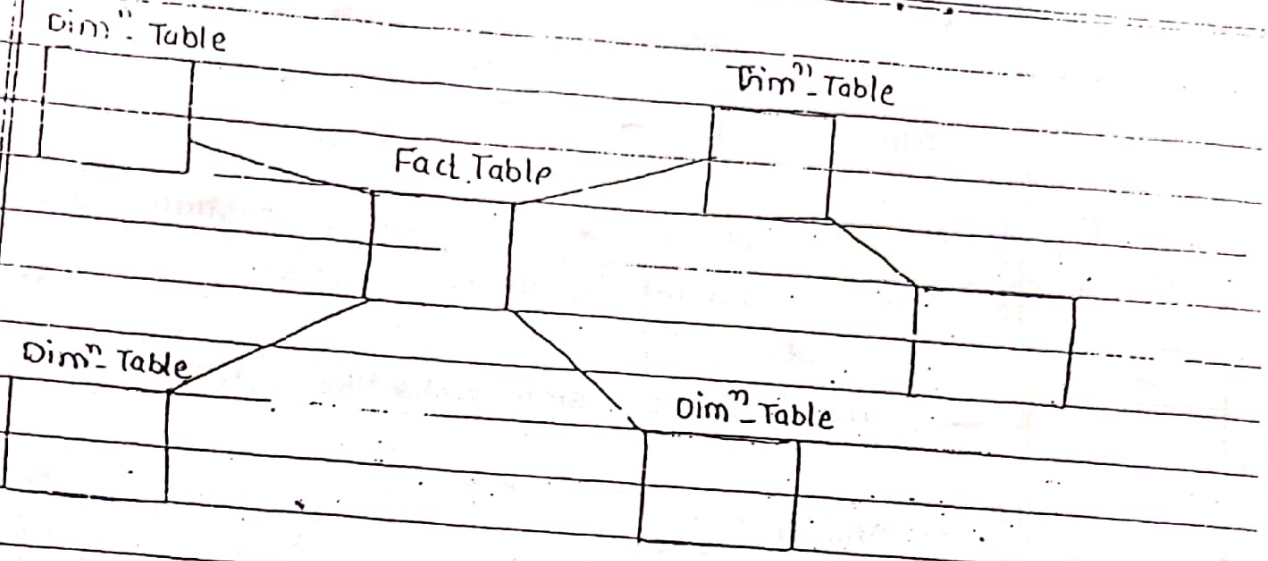
→ 1) conceptual Model → High level representⁿ of entities which is used as a part of dB design.

2) Logical Model → It is a next level representⁿ of conceptual model in which entities & attributes are specified & relatⁿ betⁿ entities.

3) Physical Model → It is a granular level of representⁿ of dB design of all entities, column, constraints present in detail.

* Facts →





Advantages of Star Schema →

1. Easy to understand.
2. Provides Better Performance.
3. can easily handle future changes.

3. Galaxy Schema →

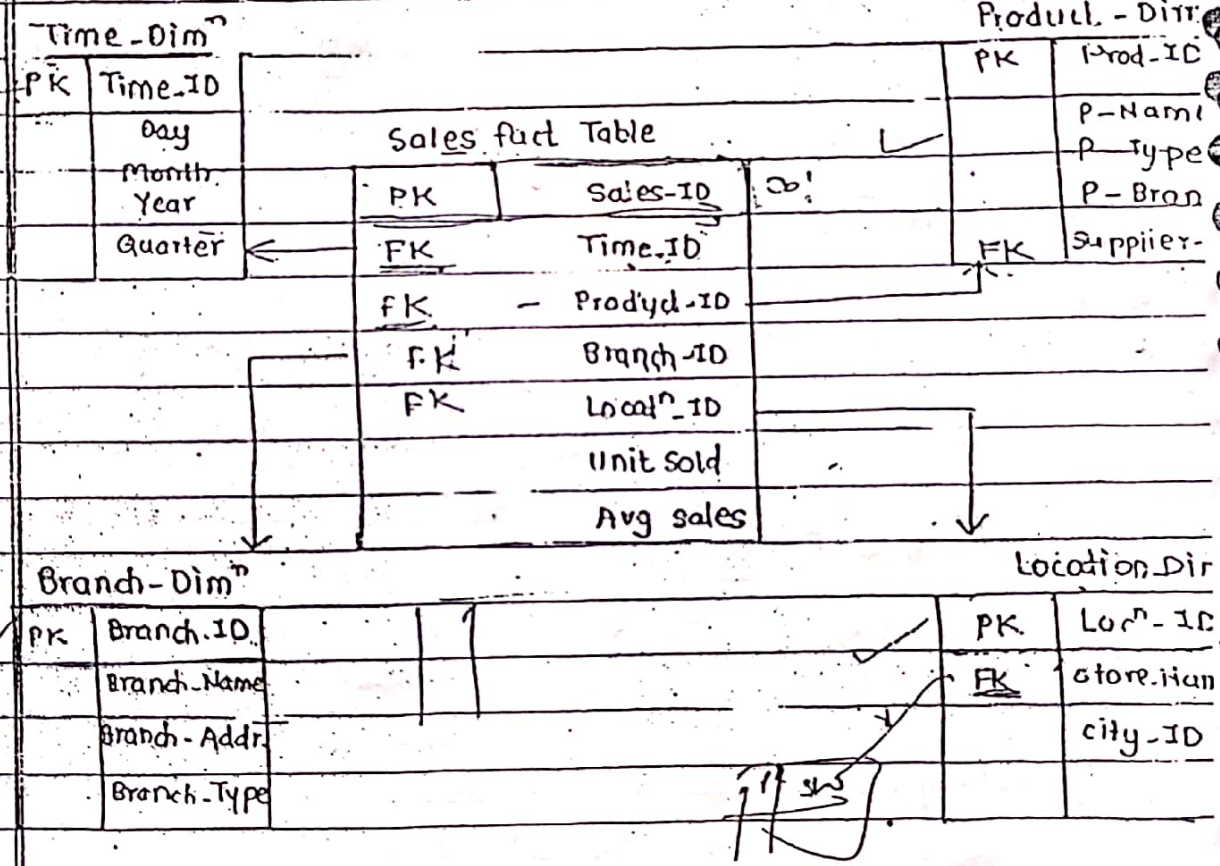
— Also known as Galaxy or fact constellat^n schema.

— Multiple fact table share the same dim^n table viewed as a collection of star schema is called as a Galaxy schema or Fact constellation.

1. Star Schema ⟶

— It is a dB design which contains centrally located fact table surrounded by dimension table.

— The database design looks like a STAR.



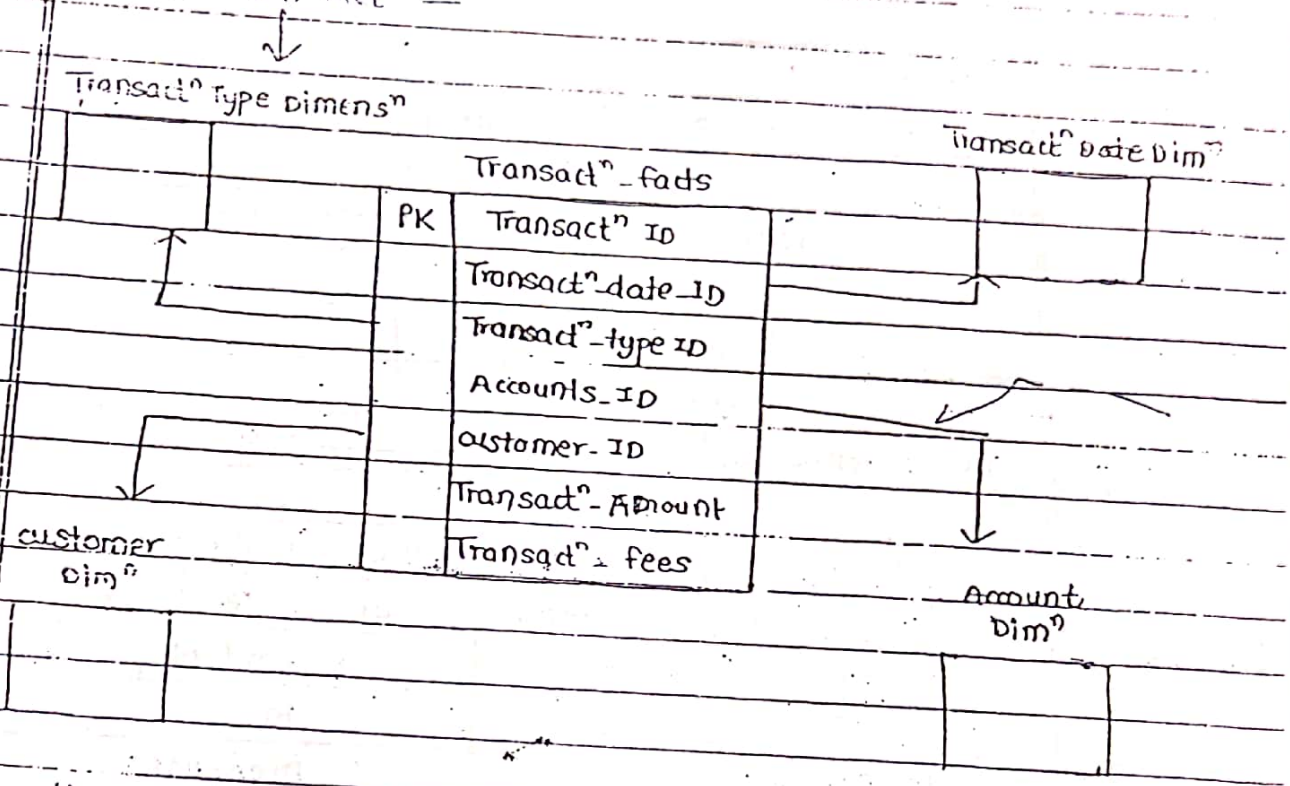
2. Snowflake Schema ⟶

— It is an extension of Star Schema.

— Dimension Table in Snowflake schema are normalized & the process of normalising dimn table is called Snowflaking

* Transaction fact —

eg



* How Data is stored in the fact Table?

ns-ID	Trans. DateID	Trans-Type	Acc-ID	Cust-ID	Trans-Amt	Trans-fees
1	25	2	47	51	5000	0.00

* SCHEMA :-

— Skeleton structure that represents logical view of entire database.

— Schema defines how the data is organised in database.

* e.g. fact & dimens' Table?

Facts :-

— It is a counted or measured event.

* Dimension → contains referential infn. about fact.

No. of Students	57

Dimension ↑ Facts ↑

* Fact Table → 1) central table in dimens^{n} model surrounded by dimens^{n} tables,
contents,
facts,
Measures

2) contains foreign keys of dimens^{n} tables.

* Dimension Table → 1) Dimension Table contains
Dimen^{s} keys
& Attributes,
values

* GALAXY SCHEMA *

* Surrogate Key :-

ID(S-K)	Start Date	End Date	Material Rate	Material Name
101	1 Jan 18	12 Feb 18	50	Steel
102	15 Feb 18	28 Feb 18	175	cement
103	1 Jan 18	28 Feb 18	125	Bugash
104	1 Jan 18	30 Aug 18	100	coalash
105	1 March 18	30 Aug 18	175	cement

— For good practice in dB design we have to maintain primary key for each table.

1. Use part of a data in a table as a primary key

2. Use new field with artificially or autogenerated value to specify a primary key in a table.

This key is known as **'Surrogate Key'.**

— This key itself has no meaning & it may not be visible to end user.

Syntax :- 1) col-name identity (start-val, increment val)

primary key (col-name)

e.g. Id identity (100, 1)

Primary key (Id)

SQL server

2) create sequence sq-name
min value = 100
increment by 1
max value = 1000;

Id sq-name (100, 1)

Primary key (Id);

Oracle

Key in fact Table

Date 20-8-17
Page

* Composite key :-

Primary key Primary key

Order_ID	Date	Product_ID	Produd_Name	Price $	Quantity
E101	22/11/17	P125	Modem	50	1
E101	22/11/17	P16	Router	100	5
E102	23/11/17	P125	Modem	100	2
E102	23/11/17	P51	Printer	25	2

— **composite key** refers to case where more than one column is used to specify primary key in a table.

Syntax :-
```
create table table_name
(
    column1 integer
    column2 varchar2 (20)
    column3 varchar2 (20)
    ;
    Primary key (column1, column2)
);
```

IMP

** Types of Fact :-

1. Additive (Additive)
2. Semi-Additive
3. Non Additive

1. Additive →

— Additive fads can summed up across all dim^n in a table.

		Date_ID	Sales Amount ↓
		Store_ID	
		Product_ID	Day 1 :- 100 $
Additive	→	Sales_Amount	Day 2 :- + 250 $
Non Additive	→	Profit_margin	Day 3 :- + 300 $
			650 $

2 Non-Additive →

— cannot summed up with any dim^n in a table.

Profit Margin
↓
Day 1 :- 5 %
Day 2 :- + 8 %
13 % ✗ (not possible)

3. Semi Additive →

— can summed up with some dim^n in a fact table but not with others.

Date_ID
Account_ID
current_Balance

— current balance can summed up with account ID dim^n but not with Date_ID dim^n.

* Types of Fact Tables :-

1. Transactional Fact Table
2. Periodic Snapshot fact Table.
3. Accumulated Fact Table.
4. Factless Fact Table.

1. Transactional Fact Table :-

— Fact Table that represents an event that occured at instantaneous pt. of time.

Mostly Additive facts

2. Periodic Fact Table ⟶

— Fact Table that describes set of things in a particular instance of time.

— Time period is predictable or regular.

Dimens^{n} Table

Date-ID	Day	Months	Year
1	14	4	2018
2	23	6	2018

* Fact Table *

BatchID	Date-ID	No.of Students

Dimens^{n} Table

Mool	JAVA
Jool	.Net
Nool	Manual

Fact Table

Batch-ID	Date-ID	No.of students
Mool	2	55

3. Accumulated Fact Table ⟶

— used to show activity of process that has well defined beginning & end.

Step 1 –	order Date	NULL	NULL
Step 2 –	order Date	Shipping Date	NULL
Step 3 –	order Date	Shipping Date	Delivery Date

* DIFFERENCE Betn ⟶

	Transactional	Periodic	Accumulated
	Unpredictable	Regular or predictable	Undetermined time span
	Insert	Insert	Insert & Update.
	one Row per Transactn event	one Row per Period.	one Row per life.

Date 21-8-18

* 4. Factless Fact Table →

— It is a fact table which does not contain — Numeric Fact OR Measures & contains only dimension keys.

— It captures an event that happened only at info. level not at calculation level.

— It captures many to many relationships bet^n dim^n table but contains no numeric facts.

* Types of Dimensions →

1. Slowly changing Dim^n
2. confirmed dim^n.
3. Degenerated dim^n.
4. Junk Dim^n.
5. Role Playing dim^n.
6. Rapidly changing dim^n.

*** IMP

1. Slowly changing Dim^n →

— Dim^n attribute that changes slowly over a period of time rather than changing regularly.

Date
Page

* Types of Slowly changing Dim^{n} (SCD) →

1. SCD Type 1 :-

e.g.

cust_ID	cust_Name	Year	location
101	XYZ	2005	London

O/P →

cust_ID	cust_Name	Year	Loc^{n}
101	XYZ	2008	Paris

— Type I :- replaces old entry with new one.

:- only new data is present & old data is completely lost.

2. SCD Type 2 :-

cust_ID	cust_Name	Year	Location
101	xyz	2005	London
101	XYZ	2018	Paris

re will
Surrogate
key

— Type II :- Stores new data as well as old data.

:- New record & old record present in a same table.

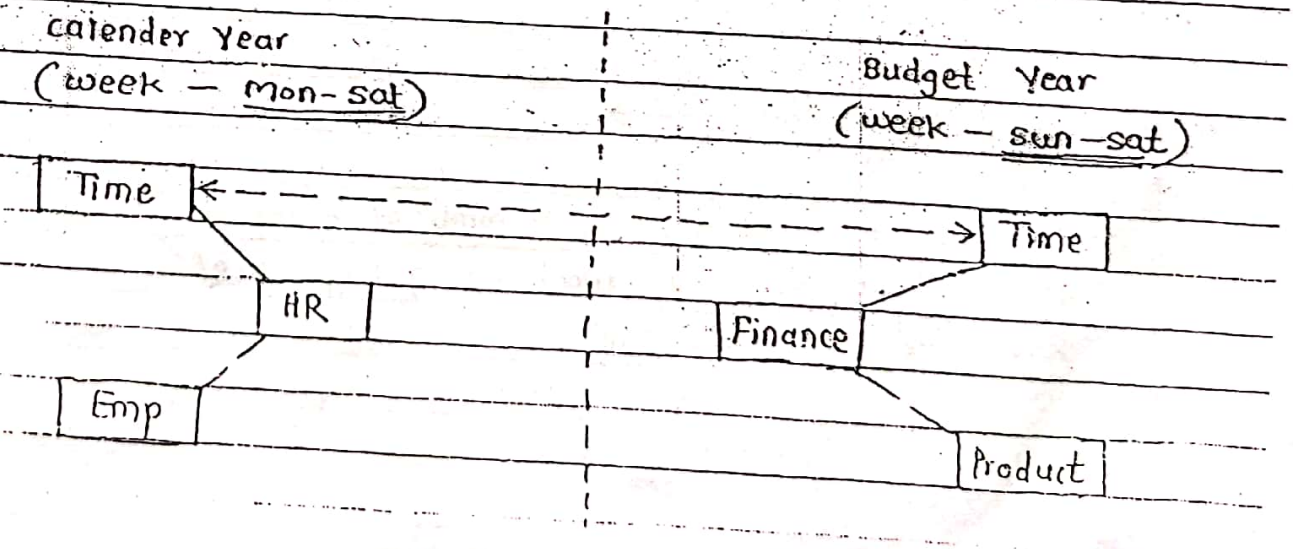
3. SCD Type 3 :-

e.g.

Cust-ID	Cust-Name	old Yr	old Locn	old New Yr	New Locn
101	XYZ	2005	London	2008	Paris

— Type III :- creating new fields in a table & maintain old as well as new data in a same record.

2. confirmed Dimension →



— We can call Time dim^n as a Confirmed dim^n when it contains same descript^n, same contents of one is a subset of another.

ADVANTAGE :— 1) Less Maintainance cost.

2) Easy development.

3) Efficient ETL Work.

4) It can be reused whenever needed.

3. Degenrated Dim^n ⟶

— It is dim^n and attribute stores as a part of fact table not in a seperate dim^n table.

4. Junk Dimension ⟶

— It is a single table with comb^n of different & unrelated attributes to avoid having large no. of foreign keys in a fact table.

5. Role Playing Dimension ⟶

— It is where the same dim^n key along with its associated attributes, can be joined to more than one foreign key in the fact table.

* Oracle → Information Schema

* Examples → Additive
Non Additive
Semi Additive

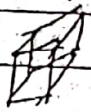
Date 23-8-18
Page

6. Rapidly changing Dimension →

— Dim^n attribute that changes frequently is called as a rapidly changing.

— If you need to track the changes using standard slowly changing dim^n, this tech. can result in a huge inflation of size of the dim^n.

— one sol^n is to move attributes to its own dim^n with a seperate foreign key in a fact table.
This new dim^n is called Rapidly changing Dim^n.

*	DB Testing	ETL Testing
1. Primary Goal	Data validation & Data Integration	Data Extract^{n}, transform & loading for reporting & analysis.
Applicable syst.	Transactional syst where business flow occur.	syst. containing historical data not business flow occur.
Need	Ensuring data integrity from multiple appl^{n}.	Ensured the req. data is moved from source to target.
Modelling	E–R diagram	Multi-Dimensional. 
	Normalised data – more no. of tables	Denormalised data — few no. of tables.

* What is diff. bet^{n} DWH & Data Mining.

— DWH is a huge concept as compared to data mining.

— DWH is a dB & Data mining is a process.

— Data mining involves extracting info. from DWH & interpret it for future predict^{n} (Analysis, Planning, designing, reporting)

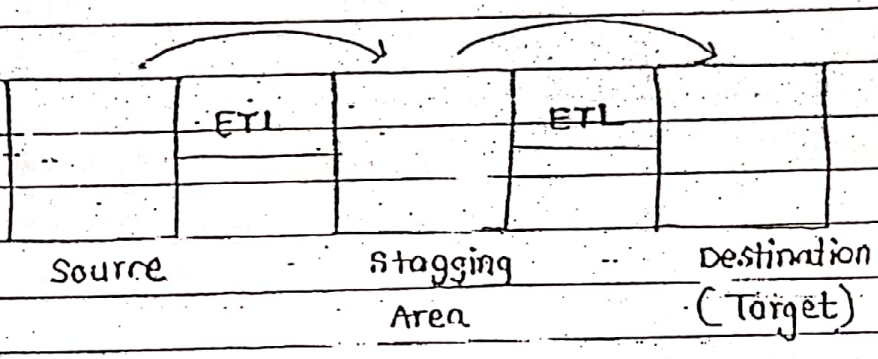
y developer

Date
Page

* Testing :-

* What is ETL ?

→



— ETL defines mechanism of dataflow from source syst. to target syst.

— It is a process of extracting data from source syst, transformed w.r.t. set of rules, after getting req. data data is loaded to target.

* ETL Testing →

defn → ETL Testing is done to ensure that the data that has been loaded from source to target (DWH) after applying set of rules (Business logic) is correct or not.

* (ETL — product° Reconcillation)
& Table balancing

Data Validation
Table-structure valid^{n} -
Data completeness
Data Duplication
Finding Invalid Data

ETL Testing also involves verifica^{n} of data in various middle stages that has been used bet^{n} source to target.

* Challenging during ETL Testing ⟶ Issues

1. Data loss during ETL process.
2. Incomplete, Incorrect, Invalid, Duplicate data is present at target.
3. DWH contains historical data, data volume is too large.
4. Tough to generate test cases and finding scenarios because the data vol. is large.

* Types of ETL Testing ⟶

1. constraint
2. source to target
3. Source to target data valida^{n}
4. Data integrated Testing
5. Incremental & historical process Testing
6. Data completeness
7. Data Transform^{n} Testing.

Date
Page

1.* Constraint Testing →

— During this testing test ER. identifies whether the data is mapped from source to target or not.

count testing.

- Not NULL
- Default
- unique
- check
- Primary key
- Foreign key

} Constraints

1. NOT NULL — ensure that the column cannot have null value.

2. Default — ensure the default value for a column whether the none is specified.

3. Unique — ensure that all values in column are different.

4. Check — make sure that all values in column satisfies certain criteria.

5. Primary — used to identify uniquely row in the table.

6. Foreign — used to ensure referential integrity of data.

File

select count * from tgmeb.table

Date 27-8-18
Page

2. * Source to target count Testing —

— During this test ER ensure that count of source & target data is expected or not.

4. * Data Integration Syst is same as SIT.

5. * Incremental & Historical Process Testing. →

— In this type of testing we are going to verify new data (changed data) or historical data in target syst.

— There are 2 types of loading in ETL process.

1. Incremental loading → only loads the data that changed in source syst.
2. Full Loading:

When
It truncates all existing tables and reloads all the data in source syst.

6. * Data completeness Testing —

— Data completeness testing ensures all the desired data is completely loaded to the target.

2. validate parent to child relationship of new field.

5) Data quality —

1) Data format check — 2) (precision value check)

↓

eg. 19.2356

Duplicate Check Testing —

— In this testing we are going to identify duplicate data in target syst.

— Duplicate data may arise
↓ because
1. Primary key not defined.
2. Wrong development.
3. Environmental Issues

— When there is huge amount of duplicate data in target syst. that may results incorrect analysis & reporting.

* Types Of ETL Bugs :-

1. Table str. Issue
2. Issue with data & source syst.
3. Data count not matching bet^n source & target.
4. Duplicate data loaded issue.
5. X^n rules issue.
6. Data format Issue
7. Index not create after Jobrun.

8. Performance Issue.

* S/w Testing	DWH Testing
1. S/w testing carried out prior to deployment of s/w.	1. DWH Testing carried out post deployment.
2. Source code specific.	2. Content specific.
3. Focussed on Used cases which contains various Test cases.	3. Focussed on querying the test data loaded by ETL Testprocess.

* Mapping Document —

Date 8-9-18
Page

* Mapping Document :-

Source				Target				Logic
S-Name	Tbl-name	col-name	Data type	T-name	Tbl-name	col-name	Data type	
SysLdB	Material	Material type	varchar (50)	DM-name	Material	Material -type	varchar (50)	only materi. which is applicable for weight
dB-name	user -details	user -DOB	Date time	DM Name	user details	user age	varchar (50)	sys − DOB / 365

Que. What is Mapping Document ?

— Mapping document describes relationship betⁿ extreme starting pt. & extreme end pt. of ETL process which is used in DWH.

Que. Why tester needs mapping doc. ?

→ 1. Because it contains details about each & every table which is a part of ETL process from source to target.

2. While working on target tables we have to refer tables in source syst. & mapping doc. contains

complete info. about tables in source syst, target syst along with its X^n logic.

Sr.No. Serial No. — 1

Test case ID — TC_001

Priority — High

Reference — SRS doc

Title/ Summary — Structure valid^n Table-name

Pre-cond^n — source exists

Test data —

Action / Description — To validate table constraint against com-mapping doc. also validate col-name

Expected Result —

Actual Result

Defect ID

Selete month_between (01/01/2018, 01/01/2018)