Agenda

a. Star Schema
b. Snowflake Schema
c. Galaxy Fact Schema
d. Slowly Changing Dimensions
e. Surrogate Keys
f. Problem statement
g. Existing Solution
h. What are data warehouses and what is OLAP?
i. How does Apache hive act as a data warehouse?
j. What makes hive as Hive? (Architecture)

Amazon Books

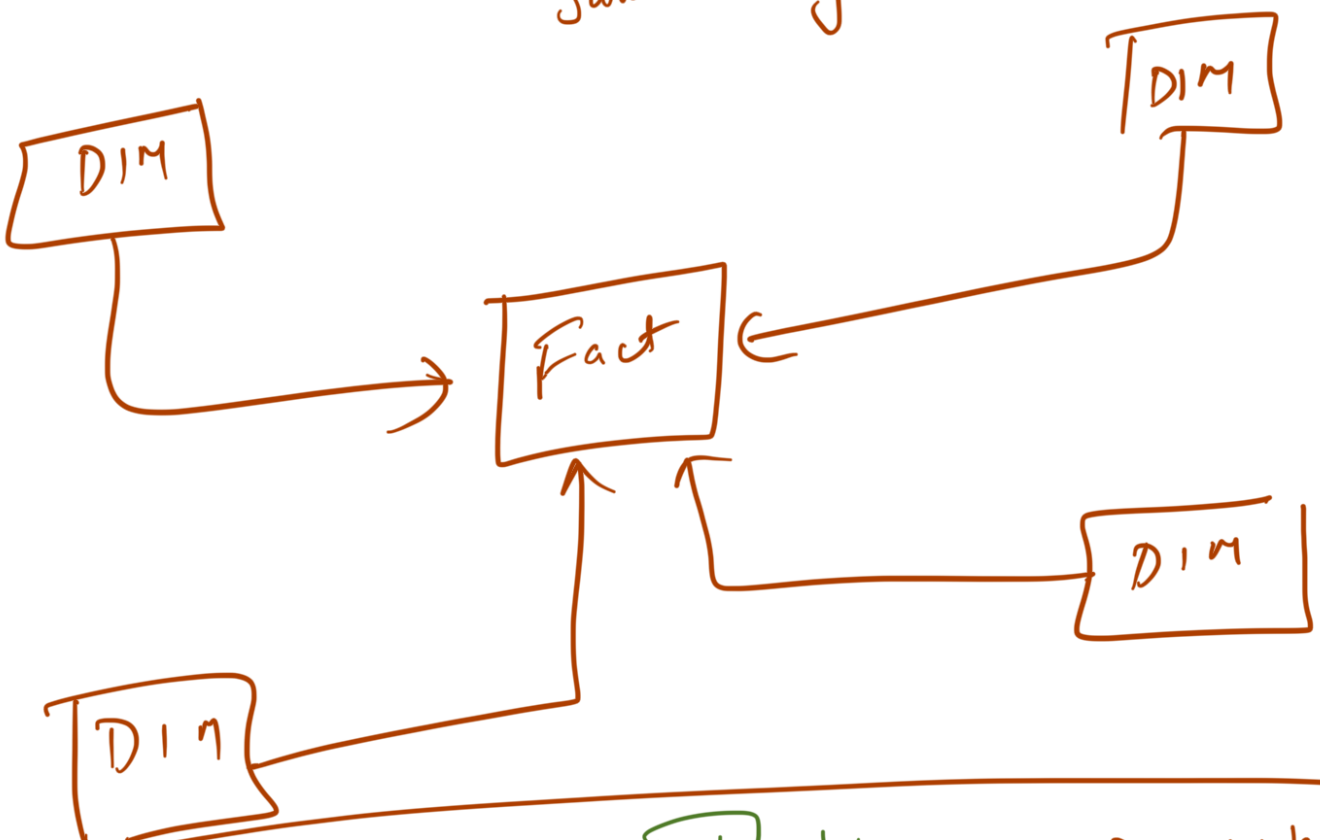
→ Fact table (Numerical + FK)

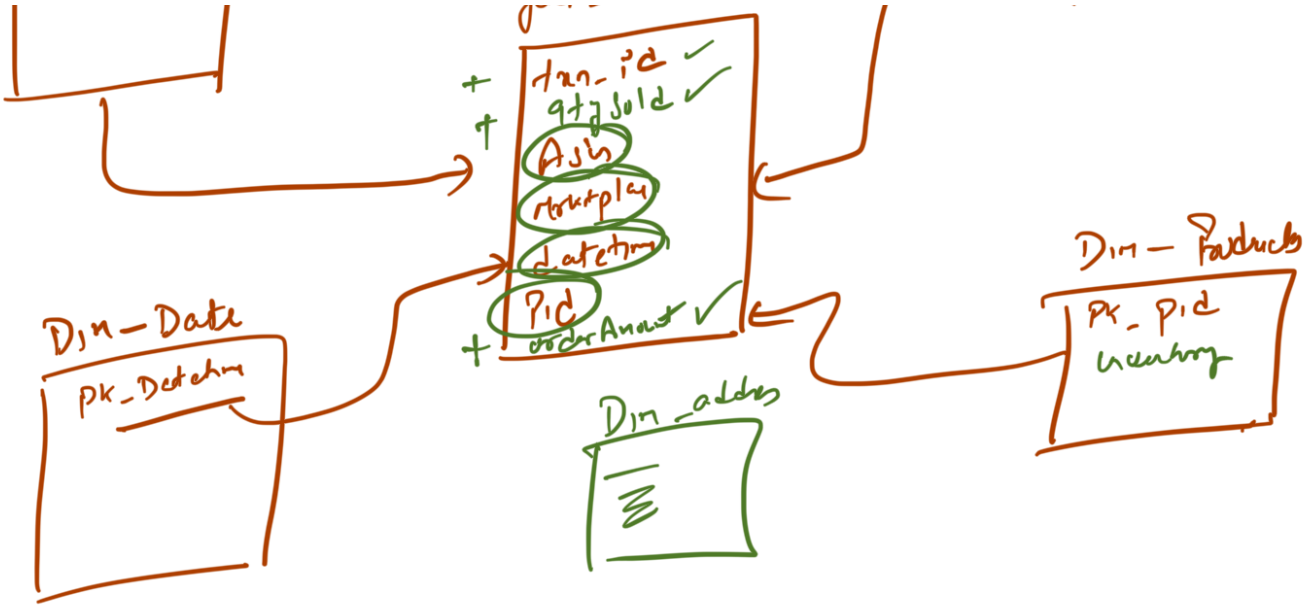
→ Dimension table (Pk + Contextual)

① Star Schema

↳ Simplest form of data relationship structure.

↳ Fact table is at center, surrounded by a series of dimensions.





(2) Snow flake Schema !!

↳ extended form of star schema by normalizing dimension table.

Snowflake (N)	star (D)
(1) No redundancy, hence more easy to maintain & change.	(1) has redudat Data and hence less easy to maintain/change.
(2) More Complex queries and less easy to understand	(2) less Complex queries and easy to understand.
(3) More foreign keys and hence more execution time	(3) less no. of keys and hence lesser execution time
(4) Good to use for Dwh Core to Simplify relationships (M:M)	(4) Good for data marts with Simple relations (1:1, 1 to many)
	fewer joins

Compu

(5) Higher no. of joins

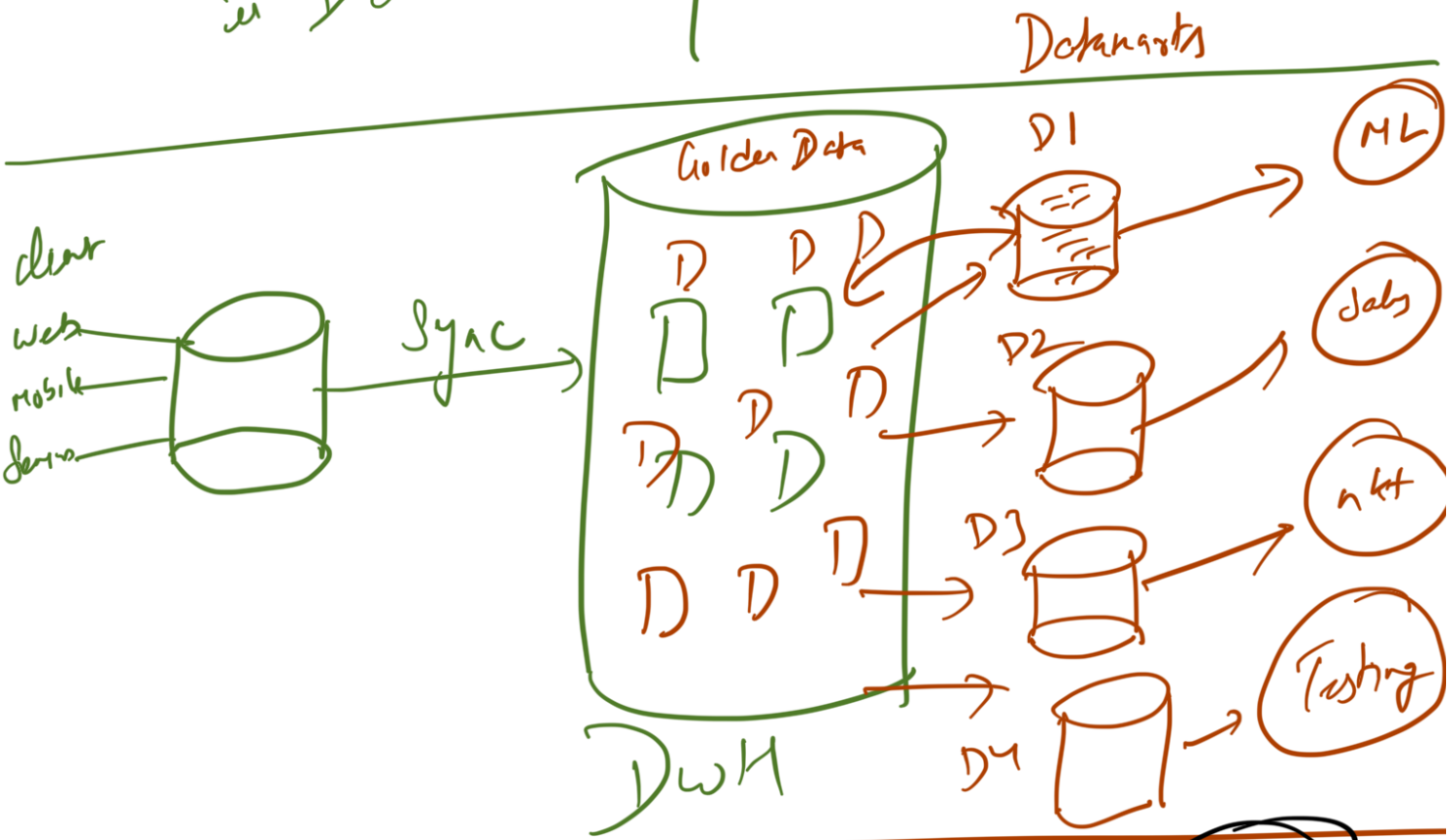
(6) when the dimension table is relatively big in size, Snowflaking is a better choice as it reduces data redundancy

(7) Dim tables are in Normalized form but fact table is still in Denormalized

(5)

(6) When the Dim table Contains less no. of rows, Go with star schema

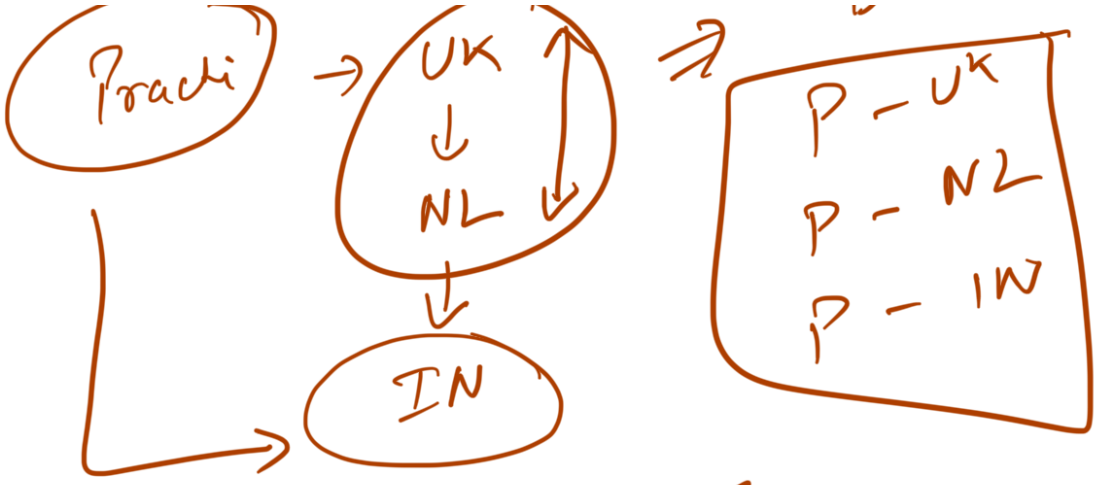
(7) Both Dim/fact are in Denormalized form.



Slowly changing Dimensions (SCD)

DWH

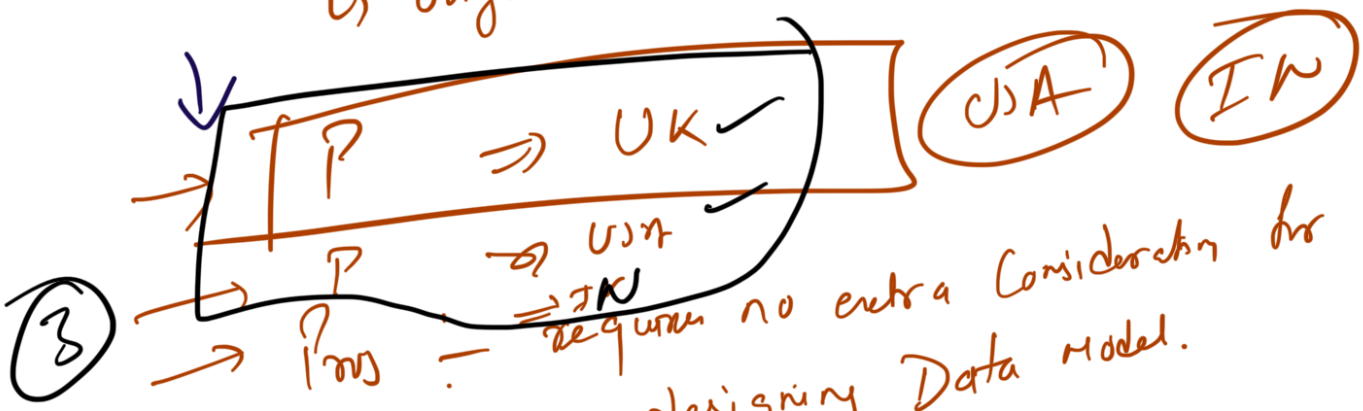
DWH



SCD Type 0 : The Passive Method

↳ No change is Captured.

↳ original Data will be retained.



Pros :- requires no extra Consideration for designing Data Model.

Cons :- Provide no Control over recording & analysing historical data & state change.

② SCD Type 1 :- overwriting the old values

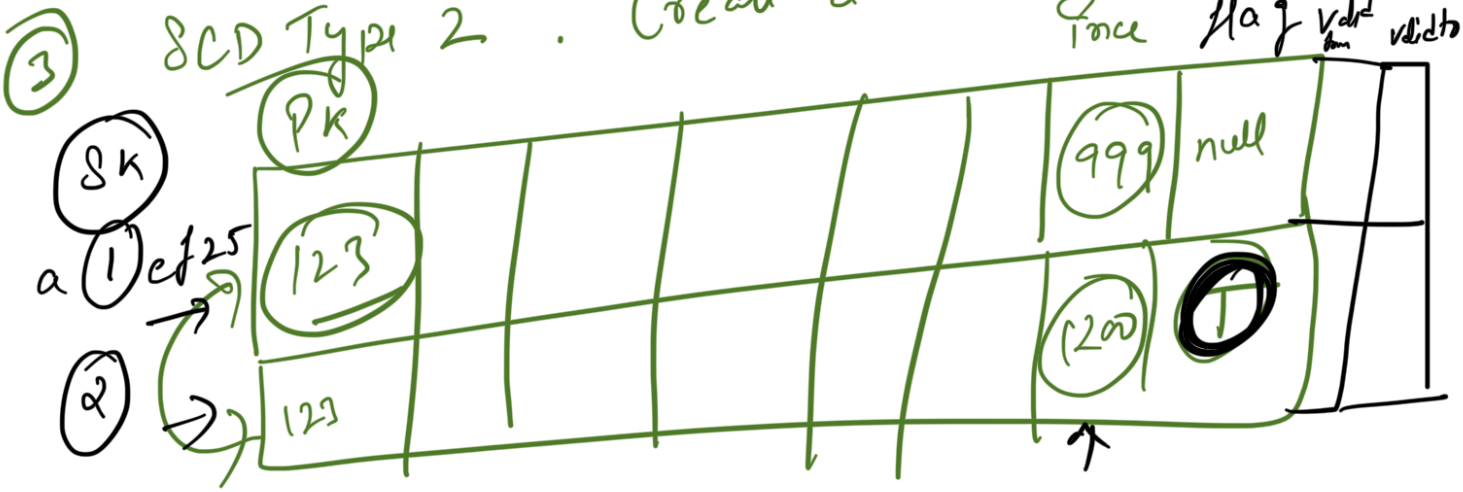
Pros :- Straightforward to implement

↳ no previous values.

↳ Uses less disk space only 1 current record exists.

Cons :- ↳ Stores no historical data for analysis
↳ Hard to find previous states before reaching the current one.

③ SCD Type 2 : Create a additional record



Surrogate keys

| PK | FK | CK | BK | UK |

⇒ natural keys

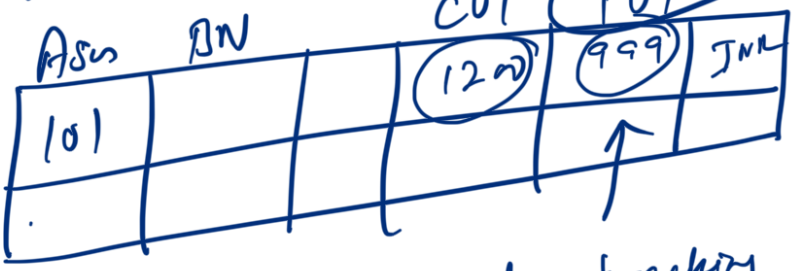
Artificial key ⇒ generated by DB

SK	ASIN			Flag	Validfrom	Validto
a2b3c	123		999	0	01/01/23	01/01/24
a2b3d	123	-	200	0	02/01/24	09/09/24
			800	1	09/09/24	10/04/24

Pros :- Unlimited legacy
↳ Store more info as to the datatypes.
↳ Original table schema doesn't change.

Cons :- More disk space.
↳ Schema design must be decided at the time of Data modelling.
↳ Changes in SCD-2 could be very expensive ops.

SCD Type 3 = Adding a new Column



Pros :- Suitable for tracking only the most recent change.

Cons :- Design limited to no of columns designated for storing historical data.

SCD Type 4 :- Using historical table

distinction b/w

Pros :- Provides a [illegible]
historical as Current data
↳ audit table works

Cons :- Two tables every time for every Dimension or Fact table
→ Disk Space is highest

* SCD Type 6 :-

SCD (1)+(2)+(3)