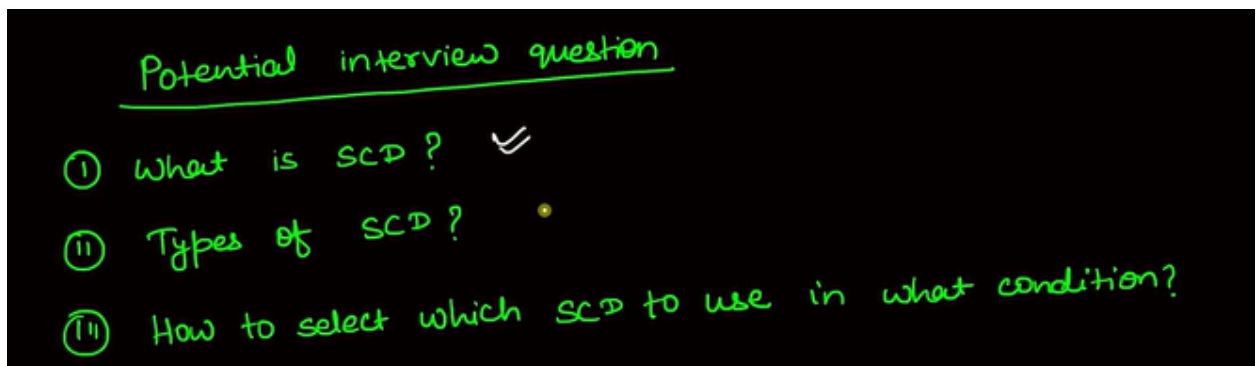


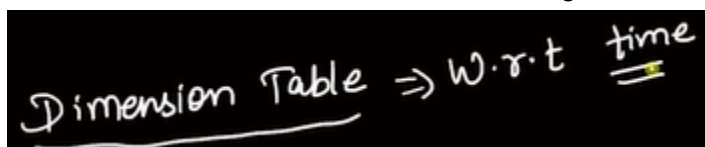
Slowly Changing Dimension

1. Potential Interview Questions



2. Lets understand SCD

3. Till now we have seen DT..that do not change wrt to time



4. Lets consider customer_dim_tab

surrogate-key
↓ • customer - dim - tbl

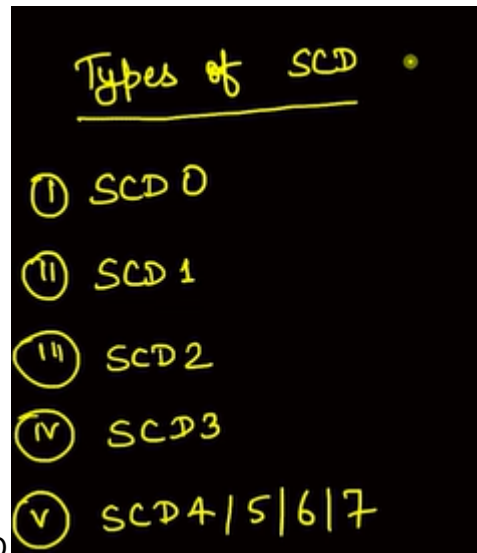
customer-key	customer-id	name	Address	State
1	RM101	Manish	Gurgaon	Haryana
2	RM102	Rohan	Gurgaon	Haryana
3	RM103	Rahul	Benaras	U.P
4	RM104	Vikash	Arwal	Bihar
5	RM105	Pritam	Craya	Bihar

5. Here surrogate key acts as a primary key and customer_id acts as natural key

• Surrogate Key Approach:

- Introduce a new column "Customer ID" as the primary key (auto-generated integer).
- "Customer Email" remains for user identification but is not the key for database operations.
- Benefit: Guaranteed unique identifier simplifies data manipulation and relationship management with other tables.

6.



7. Types of SCD ...SCD0,1,2,3 are imp for interview

8. SCD0 will always remain as original ...there won't be any changes

9. Example date of joining column in the table (there won't be any changes made to DOJ column)

Customer - dim - tbl

Customer-key	Customer-id	name	Address	State	DOJ
1	RM101	Manish	Gurgaon	Haryana	13/1/2023

10. SCD1

11. Here from our example...what if customer changes his address to banglore ..then we have to update the address

Customer - dim - tbl

Customer-key	Customer-id	name	Address	State
1	RM101	Manish	Gurgaon	Haryana

12.

→ overwrite / Inplace update

13. Here we did

14. So here we'll not be having historical record...means we cannot find customer's old location

15. Implementation is easy..but there won't be any historical records

16. SCD2

SCD2

→ History will be retained.

Customer - dim - tbl

Customer-key	Customer-id	name	Address	State
1	RM101	Manish	Gurgaon	Haryana
2	RM102	Rohan	Gurgaon	Haryana
3	RM103	Rahul	Banaras	U.P
4	RM104	Vikash	Aruwal	Bihar
5	RM105	Pritam	Craya	Bihar

17.

18. History retention is very hard to implement

19. Now to implement it..we have to introduce 3 more cols

Customer - dim - tbl

Customer-key	Customer-id	name	Address	State	Status	Start-date	end-date
1	RM101	Manish	Gurgaon	Haryana	Y	13-01-2023	31-12-9999
2	RM102	Rohan	Gurgaon	Haryana	Y		31-12-9999
3	RM103	Rahul	Banaras	U.P	Y		31-12-9999
4	RM104	Vikash	Aruwal	Bihar	Y		31-12-9999
5	RM105	Pritam	Craya	Bihar	Y		31-12-9999

20. Here in end_date instead of keeping null..we'll set that to the infinite date..which helps us in joining

21. Now if there's any update in the column...then it creates a new row for the same customer_id

Feb 2024

Customer - dim - tbl

customer-key	customer-id	name	Address	State	Status	start-date	end-date
1	RM101	Manish	Gurgaon	Haryana	Y	13-01-2023	31-12-2023
2	RM102	Rohan	Gurgaon	Haryana	Y		31-12-2023
3	RM103	Rahul	Banaras	U.P	Y		31-12-2023
4	RM104	Vikash	Aruwal	Bihar	Y		31-12-2023
5	RM105	Pritam	Chaya	Bihar	Y		31-12-2023
6	RM101	Manish	Bangalore	Kannataka	Y	15-01-2024	31-12-2024

22. Here if we can see...customer RM101..has updated his address to blr....we are storing the version's using start_date..see pic and understand
23. Now in this table...there will only be one record which hve its status as active..so will make the old address one inactive

Feb 2024

Customer - dim - tbl

customer-key	customer-id	name	Address	State	Status	start-date	end-date
1	RM101	Manish	Gurgaon	Haryana	N	13-01-2023	31-12-2023
2	RM102	Rohan	Gurgaon	Haryana	Y		31-12-2023
3	RM103	Rahul	Banaras	U.P	Y		31-12-2023
4	RM104	Vikash	Aruwal	Bihar	Y		31-12-2023
5	RM105	Pritam	Chaya	Bihar	Y		31-12-2023
6	RM101	Manish	Bangalore	Kannataka	Y	15-01-2024	31-12-2024

24. And we will also update the...end_date of old record

Feb 2024

Customer - dim - tbl

customer-key	customer-id	name	Address	State	Status	start-date	end-date
1	RM101	Manish	Gurgaon	Haryana	N	13-01-2023	14-01-2024
2	RM102	Rohan	Gurgaon	Haryana	Y		31-12-2023
3	RM103	Rahul	Banaras	U.P	Y		31-12-2023
4	RM104	Vikash	Aruwal	Bihar	Y		31-12-2023
5	RM105	Pritam	Chaya	Bihar	Y		31-12-2023
6	RM101	Manish	Bangalore	Kannataka	Y	15-01-2024	31-12-2024

25. So this how it maintains the history of the data in the SCD2

26. SCD3

A slowly changing dimension (SCD) type 3 is a way to track changes in data warehouse dimensions over time. It focuses on keeping a limited history within the same record.

Here's how it works:

- In the dimension table, specific attributes are designated for SCD type 3 tracking.
- Two columns are added for each chosen attribute: a current value column and a previous value column.
- When the attribute value changes, the current value is shifted to the previous value column, and the new value becomes the current value.

27. Think of it like a table with revision history built-in, but limited to the most recent change.

Example:

Imagine a customer table in a retail data warehouse. One attribute is the customer's address. Addresses can change over time due to relocations.

A Type 3 SCD wouldn't create a new row for each address change. Instead, it would have:

- A column for "Current Address"
- A column for "Previous Address"

Whenever a customer moves, the "Current Address" becomes the "Previous Address," and the new address goes into "Current Address." This allows you to see the current address and the most recent previous address for analysis.

Benefits:

- Simpler to implement compared to other SCD types.
- Efficient for storing limited historical data.

**** drawbacks:****

- Only tracks the most recent historical value, not a complete history.
- Not ideal for situations where multiple historical values are crucial.

28.

Use Cases:

- Tracking changes in customer attributes like name or contact information.
- Monitoring product attribute changes like price or category.

