**Data Warehouse Assignment-2**

**Notes:**

# What is a Slowly Changing Dimension?

A Slowly Changing Dimension (SCD) is a dimension that stores and manages both current and historical data over time in a data warehouse,

|  |  |
| --- | --- |
| **SCD Type** | **Summary** |
| Type 1 | Overwrite the changes |
| Type 2 | History will be added as a new row. |
| Type 3 | History will be added as a new column. |
| Type 4 | A new dimension will be added |

# What is an outrigger table in data warehouse?

Outriggers are tables or entities that are shared by more than one dimension. A table or entity that is included in a hierarchy but is not directly related to the fact table are known as outriggers.

**Question 1:**

**1.What are the strength and weakness of each option?**

Option A

Strength: Allows the instructor dimension to be included in the fact table, which may be useful for certain queries. Allows to represent courses with multiple instructors without needing to change the grain of the fact table.

Weakness: Requires addition of special rows to the instructor dimension table to represent instructor team , which may make the dimension table more complex and harder to maintain .May not be intuitive for users to understand how instructor teams are represented in the fact table.

Option B

Strength: Allows you to include multiple instructors in the fact table,which may be useful for certain queries.Does not require the addition of special rows to the instructor dimension table.

Weakness:Changes the grain of the fact table,which may take it harder to compare enrollment counts between courses with different numbers of instructors.Requires the use of fractional enrollement counts to allocate enrollments equally among multiple instructors which may be confusing for the users.

Option C

Strength: Allows you to include the instructor dimension in queries that need it, while still allowing the use of a simpler fact table for other queries. Does not require the addition of special rows to the instructor dimension table or the use of fractional enrollment counts.

Weakness: Requires the creation and maintenance of two separate fact tables, which may be more complex and time-consuming than using a single fact table. May be confusing for users to understand which fact table to use for different types of queries.

**Question 2. Which option would you choose and why?**

Ans:I will be choosing option C

Why?

Allows you to include the instructor dimension in queries that need it, while still allowing the use of a simpler fact table for other queries. Does not require the addition of special rows to the instructor dimension table or the use of fractional enrollment counts.

**Question 3:Would your answer to Question 2 be different if the majority of classes had multiple instructors?**

If the majority of classes had multiple instructors, Option C (creating two fact tables) may be a more attractive option because it allows you to include the instructor dimension in queries that need it, while still allowing the use of a simpler fact table for other queries.

How about if only one or two classes had multiple instructors?

If only one or two classes had multiple instructors, Option C (creating two fact tables) may be less attractive because it requires the creation and maintenance of two separate fact tables, which may be more complex and time-consuming than using a single fact table. It may also be confusing for users to understand which fact table to use for different types of queries.

**Scenario 2:**

**Question 5. What are the strengths and weaknesses of each option?**

Option A (Type 1 Slowly Changing Dimension):

Strengths:

1. Simple to implement and maintain
2. Does not require additional storage space for multiple versions of the customer scores

Weaknesses:

1. Loses the ability to track how customer scores have changed over time

Option B (Type 2 Slowly Changing Dimension):

Strengths:

1. Allows you to track how customer scores have changed over time
2. Supports queries that need to filter or group by the current or previous value of a customer score

Weaknesses:

1. Requires more storage space to maintain multiple versions of the customer scores
2. More complex to implement and maintain than a Type 1 SCD

Option C (Separate CustomerScores dimension table):

Strengths:

1. Supports queries that need to filter or group by customer score
2. Allows you to store additional customer score attributes (if needed)
3. Simplifies the Trades fact table by removing the need to store customer score attributes

Weaknesses:

1. Requires additional storage space and maintenance effort to maintain the CustomerScores dimension table
2. More complex to implement than a Type 1 or Type 2 SCD

Option D (CustomerScores outrigger table):

Strengths:

1. Supports queries that need to filter or group by customer score
2. Allows you to store additional customer score attributes (if needed)
3. Simplifies the Trades fact table by removing the need to store customer score attributes

Weaknesses:

1. Requires additional storage space and maintenance effort to maintain the CustomerScores outrigger table
2. More complex to implement than a Type 1 or Type 2 SCD

**Question 6. Which option would you choose and why?**

I would recommend Option B (Type 2 Slowly Changing Dimension). This option allows you to track how customer scores have changed over time, which can be useful for understanding how and why the activity level or profitability of some customers changes. It also supports queries that need to filter or group by the current or previous value of a customer score.

**Question 7. Would your answer to Question 6 be different if the number of customers and/or the time interval between score recalculations was much larger or much smaller? (Explain your answer.)**

The answer to the previous question would not be different if the number of customers or the time interval between score recalculations was much larger or much smaller. Option B (Type 2 Slowly Changing Dimension) would still be a good choice for tracking how customer scores have changed over time and supporting queries that need to filter or group by the current or previous value of a customer score.