**Slowly Changing Dimension (SCD) Implementation**

**Document Version:** 1.0  
**Author:** [Your Name]  
**Date:** [DD/MM/YYYY]

**1. Introduction**

This document outlines the implementation of **Slowly Changing Dimension (SCD) Type 2** for tracking changes in customer attributes (e.g., loyalty tier) over time while preserving historical data.

**2. Customer Dimension Schema**

**Table Structure**

| **Column Name** | **Data Type** | **Description** |
| --- | --- | --- |
| customer\_key | INT (PK) | Surrogate key (auto-increment) |
| customer\_id | VARCHAR(10) | Natural business key |
| first\_name | VARCHAR(50) | Customer’s first name |
| last\_name | VARCHAR(50) | Customer’s last name |
| email | VARCHAR(100) | Customer’s email address |
| loyalty\_tier | VARCHAR(20) | Loyalty tier (Bronze/Silver/Gold) |
| start\_date | DATE | Effective date of the record |
| end\_date | DATE | Expiration date (9999-12-31 for current) |
| is\_current\_flag | BOOLEAN | 1 = Active, 0 = Historical |

**3. Implementation Steps**

**Step 1: Create Initial Table**

sql

Copy

CREATE TABLE dim\_customer (

customer\_key INT PRIMARY KEY,

customer\_id VARCHAR(10),

first\_name VARCHAR(50),

last\_name VARCHAR(50),

email VARCHAR(100),

loyalty\_tier VARCHAR(20),

start\_date DATE,

end\_date DATE,

is\_current\_flag BOOLEAN

);

**Step 2: Insert Initial Record (Jan 2022)**

sql

Copy

INSERT INTO dim\_customer VALUES

(1, 'CUST001', 'Jane', 'Smith', 'jane.smith@email.com', 'Bronze',

'2022-01-15', '9999-12-31', TRUE);

**Initial Data:**

| **customer\_key** | **customer\_id** | **first\_name** | **last\_name** | **email** | **loyalty\_tier** | **start\_date** | **end\_date** | **is\_current\_flag** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | CUST001 | Jane | Smith | [jane.smith@email.com](https://mailto:jane.smith@email.com/) | Bronze | 2022-01-15 | 9999-12-31 | 1 |

**Step 3: Simulate Tier Change (July 2023)**

**A. Expire the Old Record**

sql

Copy

UPDATE dim\_customer

SET end\_date = '2023-06-30', is\_current\_flag = FALSE

WHERE customer\_id = 'CUST001' AND is\_current\_flag = TRUE;

**B. Insert the New Record**

sql

Copy

INSERT INTO dim\_customer VALUES

(2, 'CUST001', 'Jane', 'Smith', 'jane.smith@email.com', 'Gold',

'2023-07-01', '9999-12-31', TRUE);

**Final Output After Update**

| **customer\_key** | **customer\_id** | **first\_name** | **last\_name** | **email** | **loyalty\_tier** | **start\_date** | **end\_date** | **is\_current\_flag** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | CUST001 | Jane | Smith | [jane.smith@email.com](https://mailto:jane.smith@email.com/) | Bronze | 2022-01-15 | 2023-06-30 | 0 |
| 2 | CUST001 | Jane | Smith | [jane.smith@email.com](https://mailto:jane.smith@email.com/) | Gold | 2023-07-01 | 9999-12-31 | 1 |

**4. Key Features of SCD Type 2**

✅ **Tracks historical changes** while maintaining current data.  
✅ Uses **surrogate keys** (customer\_key) to distinguish versions.  
✅ **Effective dating** (start\_date/end\_date) for time-based queries.  
✅ **Current flag** (is\_current\_flag) simplifies active record filtering.

**5. Additional Notes**

* For **SCD Type 1** (overwrite), simply update the existing record.
* For **SCD Type 3** (keep limited history), add previous\_loyalty\_tier and change\_date columns.

**Approval:**  
Prepared by: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Reviewed by: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_