### **Potential Solutions:**

#### **Solution 1: Maintain Backward Compatibility with Old Surrogate Keys**

Introduce a **dual-key approach** where the model retains both:

**Old surrogate key (HASH(A, B, C))** for existing records.

**New surrogate key (HASH(A, B, C, D))** for new records.

Power BI handles both keys by creating **calculated columns** that select the appropriate key dynamically.

This ensures old records are still linked while supporting the new column structure.

##### **Implementation:**

Add a column in dim\_sales\_attributes:

sql

SELECT A, B, C, D,  
 MD5(CONCAT(A, '|', B, '|', C)) AS old\_key,   
 MD5(CONCAT(A, '|', B, '|', C, '|', D)) AS new\_key   
FROM detailed\_sales   
GROUP BY A, B, C, D;

Modify the fact\_aggregated\_sales table to store both keys.

Power BI model updates relationships to use the correct key.

#### **Solution 2: Use a Mapping Table for Key Translation**

Create a **mapping table** (key\_translation) to link **old surrogate keys** to **new surrogate keys**.

This allows Power BI to look up the appropriate key dynamically.

##### **Implementation:**

Generate a mapping table with old and new keys:

sql

CREATE TABLE key\_translation AS   
SELECT   
 MD5(CONCAT(A, '|', B, '|', C)) AS old\_key,   
 MD5(CONCAT(A, '|', B, '|', C, '|', D)) AS new\_key   
FROM detailed\_sales   
GROUP BY A, B, C, D;

Modify fact\_detailed\_sales and fact\_aggregated\_sales to join with key\_translation for key resolution.

Power BI uses a **lookup table** to translate keys dynamically.

#### **Solution 3: Implement a Rolling Migration Strategy**

Instead of regenerating all keys at once, implement a **rolling migration** where:

New records use the **new surrogate key (HASH(A, B, C, D))**.

Old records gradually **transition** using an ETL update process.

Power BI reports will continue working while historical data is phased into the new format.

##### **Implementation:**

Keep existing surrogate keys and only update when necessary:

sql

UPDATE fact\_detailed\_sales   
SET surrogate\_key = MD5(CONCAT(A, '|', B, '|', C, '|', D))   
WHERE D IS NOT NULL;

Define a **data refresh policy** in Power BI to gradually transition old data.

### **Recommendation & Next Steps**

Each approach has trade-offs in terms of implementation complexity and system impact. The choice depends on:

**Performance considerations** – Dual-key and mapping table approaches introduce additional joins.

**Data consistency requirements** – Rolling migration may introduce temporary inconsistencies.

**Ease of implementation** – Mapping tables provide an immediate resolution but require additional ETL logic.

I recommend **Solution 2 (Mapping Table)** for its flexibility and minimal disruption. Let me know if you’d like to discuss further or schedule a session to evaluate feasibility.

Best regards,  
[Your Name]  
[Your Position]  
[Your Contact Information]