Graph View of the Data Model- Verifying Relationships Before Unification

1. Graph View in Salesforce Data Cloud

The **Graph View** in Salesforce Data Cloud provides a **visual representation of your data model**, showing **how Data Model Objects (DMOs)** are **connected to each other** through **relationships and identity fields**.

It is part of the **Data Model Workspace** and is used to:

- Visualize relationships between **objects** (e.g., Customer, Transaction, Product).
- Verify **join conditions** and **data lineage**.
- Ensure **key fields** are properly connected for **identity resolution** and **profile unification**.

In simpler terms, it helps you answer:

"How do my data objects relate to each other before I unify them into a single customer profile?"

---

2. Why Graph View Matters

Before you perform **identity resolution or unification**, you must ensure that:

- Your **data model is connected properly**,
- All **primary and foreign keys** align,
- Relationships between DMOs are defined accurately, and

- There are **no broken or orphan links** between objects.

This ensures a **successful unification process**, resulting in complete and accurate customer 360 profiles.

---

3. Components of the Graph View

Component	Description
Nodes (Circles)	Represent individual Data Model Objects (DMOs), such as Customer, Transaction, or Product.
Edges (Lines)	Represent relationships between DMOs (one-to-one, one-to-many, or many-to-many).
Primary Key Indicator	The field in each object that uniquely identifies a record (e.g., Customer_ID).
Foreign Key Indicator	The linking field used to connect to another object (e.g., Customer_ID in Transaction_DMO).
Labels and Tooltips	Show relationship details when hovering over connections.
Color Coding	Often used to indicate object types (standard, custom, unified).

---

4. Example: Typical Data Model Graph in Data Cloud

Let's consider a retail example where you have three DMOs:

1. **Customer_DMO** – stores customer master information.
2. **Transaction_DMO** – stores transaction or order details.
3. **Product_DMO** – stores product catalog data.

Description of Relationships

- Customer_DMO.customer_id → Primary Key
- Transaction_DMO.customer_id → Foreign Key (links to Customer)
- Transaction_DMO.product_id → Foreign Key (links to Product)
- Loyalty_DMO.customer_id → Foreign Key (links to Customer)

Each line between objects shows how data from various systems relates before unification.

---

5. How to Access and Use the Graph View

Step 1: Navigate to the Data Model

1. Go to **Data Cloud → Data Manager → Data Model**.
2. Click on the **Data Model Overview** tab.

Step 2: Open Graph View

1. Click the **Graph View** icon (network diagram symbol).
2. The visual data model diagram opens.

Step 3: Review Object Relationships

- Each **circle** represents a DMO.
- Each **line** shows a defined relationship.
- Hover over lines to view **relationship details** such as cardinality and keys.

Step 4: Verify Keys and Relationships

Confirm the following:

- Each DMO has a **primary key** defined.
- Foreign keys correctly link to parent DMOs.

- Relationship types (1:1, 1:N, N:M) are accurate.
- There are no disconnected or orphaned objects.

Step 5: Save and Validate

Once verified, click **Validate Model** to ensure that all relationships are logically correct and usable for unification.

---

6. Example: Tabular Representation of Relationships

Parent DMO	Child DMO	Relationship Type	Parent Key	Child Key	Verified
Customer_DMO	Transaction_DMO	1-to-Many	customer_id	customer_id	✓
Transaction_DMO	Product_DMO	Many-to-1	product_id	product_id	✓
Customer_DMO	Loyalty_DMO	1-to-Many	customer_id	customer_id	✓

Interpretation:

- A single **Customer** can have many **Transactions**.
- A single **Product** can appear in many **Transactions**.
- A single **Customer** can have many **Loyalty Program entries**.

---

7. Verification Checklist Before Unification

Before running unification or identity resolution, use this checklist in Graph View:

Validation Area	Description	Action
Primary Keys	Ensure each DMO has a unique ID field.	Mark one key field as the **Primary Key**.

Validation Area	Description	Action
Foreign Keys	Check that all child DMOs reference the correct parent DMO key.	Correct mapping if any mismatch.
Cardinality	Confirm that relationships match real-world structure (1:N, N:1).	Adjust mapping if incorrect.
Data Completeness	Ensure no null values in key fields.	Clean data in DLO if necessary.
Field Type Consistency	Key fields between DMOs must share the same data type.	Convert mismatched field types.
Disconnected Objects	Check if any DMO is not connected to others.	Remove or connect as needed.

---

8. Example Use Case

Scenario

A financial services company is building a unified "Customer 360" profile using data from:

- CRM (Customer_DLO → Customer_DMO)
- Transactions (Transaction_DLO → Transaction_DMO)
- Support Tickets (Support_DLO → Support_DMO)

Verification Outcome

- Primary key customer_id correctly defined in Customer_DMO.
- Foreign key customer_id in both Transaction_DMO and Support_DMO properly mapped.
- Relationship validation passed.

Benefit

- Unification produces complete 360° customer profiles with accurate links to transactions and support history.

---

9. Common Issues Detected via Graph View

Issue	Description	Resolution
Missing foreign key	Child DMO not connected to parent	Add mapping in schema
Mismatched data types	Key field types differ	Convert fields to consistent format
Incorrect cardinality	Relationship defined incorrectly (1:1 vs 1:N)	Adjust relationship definition
Orphaned object	DMO not connected to model	Reconnect or remove unused object
Circular relationship	Two DMOs reference each other recursively	Redesign model to avoid loops

---

10. Best Practices

1. **Always validate Graph View before unification** — prevents identity resolution errors.
2. **Use consistent field naming** (e.g., customer_id, account_id) to simplify mapping.
3. **Keep the data model clean** — remove unnecessary relationships or unused objects.
4. **Perform test unifications** with sample data to validate relationship logic.
5. **Document relationships** — export a table of parent-child mappings for audit purposes.

6. **Check relationship directionality** — ensure parent-to-child flow is correct.
7. **Regularly review Graph View** after adding new data sources or transformations.

---

11. Example Output: Unified Model Snapshot

After successful relationship verification and unification, your model may look like this:

Unified Profile Field	Source Object	Source Field
Customer_ID	Customer_DMO	customer_id
Name	Customer_DMO	full_name
Total_Spend	Transaction_DMO	SUM(amount)
Last_Purchase_Date	Transaction_DMO	MAX(purchase_date)
Total_Tickets	Support_DMO	COUNT(ticket_id)

Each unified profile record is now enriched with linked data from all related DMOs.

---

12. Summary

Concept	Description
Graph View	A visual interface in Salesforce Data Cloud showing how DMOs relate to each other.
Purpose	To validate and verify relationships before running unification.
Benefits	Ensures accurate joins, prevents orphaned data, and improves profile completeness.

Concept	Description
Key Validation Points	Primary keys, foreign keys, cardinality, and data type consistency.

Key Takeaway

The **Graph View** in Salesforce Data Cloud is your **visual validation checkpoint** — ensuring that every DMO relationship is correctly configured before unifying data into Customer 360 profiles.

By verifying keys, relationships, and data integrity early, you prevent unification errors and guarantee reliable, connected insights downstream.