Data Cloud Self-Service Evaluation Guide

Purpose of this Guide

This guide is designed to help provide you with **a targeted and streamlined evaluation** to determine whether Salesforce Data Cloud is the right fit for your business. Instead of conducting a lengthy proof-of-concept, you'll follow a **streamlined hands-on path** that will provide you with **working proof** that:

- You can ingest Salesforce CRM and external data (e.g., Snowflake) into Data Cloud.
- You can unify identities and resolve duplicates into a single customer profile.
- You can activate insights in Salesforce to drive actions (e.g., trigger tasks when utilization drops).

This guide is divided into three phases:

1. **Connect:** Ingest data into Data Cloud from diverse sources, including systems, cloud platforms, data warehouses, and APIs. **Key topics:** Data Ingestion, Data Federation (Zero-copy Data In), and Data Shares (Zero-copy Data Out).
2. **Transform:** Clean, unify, and enrich that data from various sources to provide a foundation of unified intelligence. **Key Topics:** Identity Resolution and Segmentation.
3. **Action**: Use activated data to drive business outcomes in your organization. This includes utilizing cleaned data for enhanced customer experiences, targeted marketing, business operations, reporting, AI applications, and intelligent agents.

By examining Data Cloud through the Connect, Transform, and Action framework you'll see not just how to set it up, but also why it matters for your business.

Initial Setup of Data Cloud (1hr)

Before you can evaluate Data Cloud, you'll need a working environment. This stage walks you through creating a free Salesforce Developer Edition org, enabling Data Cloud, and getting familiar with the basics. By the end, you'll have your own sandbox with Data Cloud and Agentforce provisioned – a safe place to test connections, transformations, and activations without touching production.

Step 1: Get Familiar with Data Cloud Concepts

Before diving in, complete the "Get Hands-on with Data Cloud" Trailhead trail.

Why this matters: This 30-minute interactive tutorial will give you the foundational knowledge needed for this evaluation.

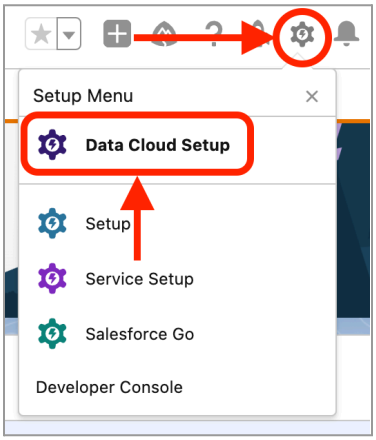
Step 2: Create Your Free Salesforce Developer Edition Org

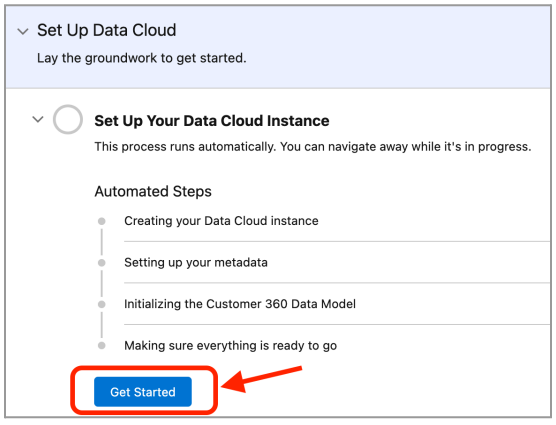
1. Go to developer.salesforce.com/signup
2. Fill out the registration form
3. Check your email and activate your account
4. Log into your new Developer Edition org

Why this matters: This free org gives you a safe environment to evaluate Data Cloud with real functionality, without risk to production.

Step 3: Enable Data Cloud

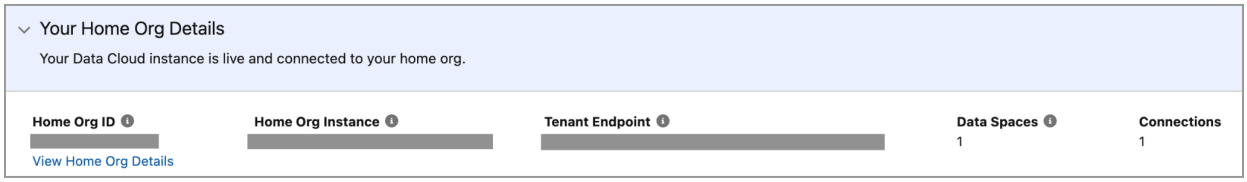
1. In your newly created Developer Edition org, go to **Setup > Data Cloud Setup**
2. Click **"Get Started"**, as shown in the images below.





The configuration will take around one hour to complete. Use this time to complete the Trailhead module if you haven't already.

✓ **CHECKPOINT:** Once provisioning completes, you will get access to your Data Cloud tenant information, like in the image below:



3. Under **"Set Up Data Cloud"**, complete the **"Assign User Permissions"** step by clicking the button. No additional action is needed.

4. Review the remaining setup points

🎉 **Congratulations!** You have successfully configured a Salesforce organization that includes both Data Cloud and Agentforce. You will use this organization for the remainder of your evaluation.

Why this matters: Once provisioned, you now have a working Data Cloud environment to test the three phases of **Connect, Transform, and Action**.

Phase 1: Connect

Goal: Prove that Data Cloud can connect Salesforce CRM and Snowflake data, via Zero-copy integration, without duplication or complex pipelines.

This section details the configuration of various data sources for your evaluation. While you have the flexibility to choose which to implement, we strongly advise, at minimum, configuring the ingestion of your own Salesforce CRM data, as outlined in the initial section below.

1. Ingest Salesforce CRM Data

Data Cloud makes it simple to bring in Salesforce CRM data by handling the heavy lifting for you. It automatically maps raw objects (Data Lake Objects) to the standard Data Model, so Accounts, Contacts, Leads, and Cases are ready to use without manual configuration. This reduces setup time, prevents mapping errors, and ensures your CRM data is structured for analysis and activation from day one.

The next steps will walk you through configuring CRM data ingestion – the foundation for building a unified and actionable customer view.

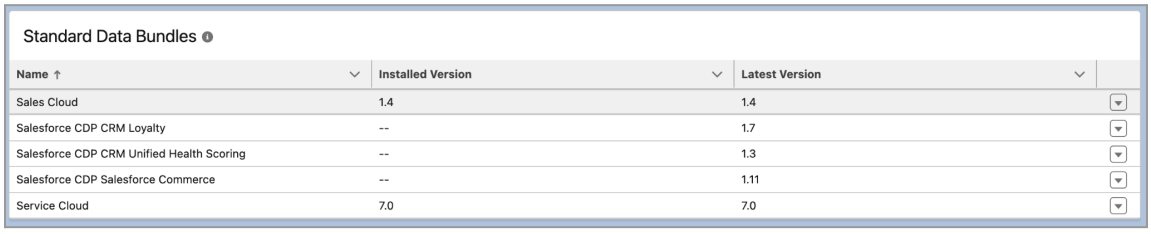
Install the Standard Bundles (Sales Cloud, Service Cloud)

Before you can ingest data, install the required CRM data bundles (powered by data kits). These bundles include the objects and mappings Data Cloud needs to pull in CRM data.

Key Steps:

1. Go to **Standard Data Bundles** in your org.
2. From the list, install at least the **Sales Cloud** and **Service Cloud** bundles (recommended starting points).
    - To install a bundle, click the arrow symbol (🔽) to its right and select **"Install".**
    - Choose whether to install for Admins only or for all users.
3. Wait for a few minutes while the bundles are installed.

    ✅ **Checkpoint:** Once installed, you'll see the version number in the **Installed Version** column, as shown below. Refer to the help guide for instructions on installing the bundles.



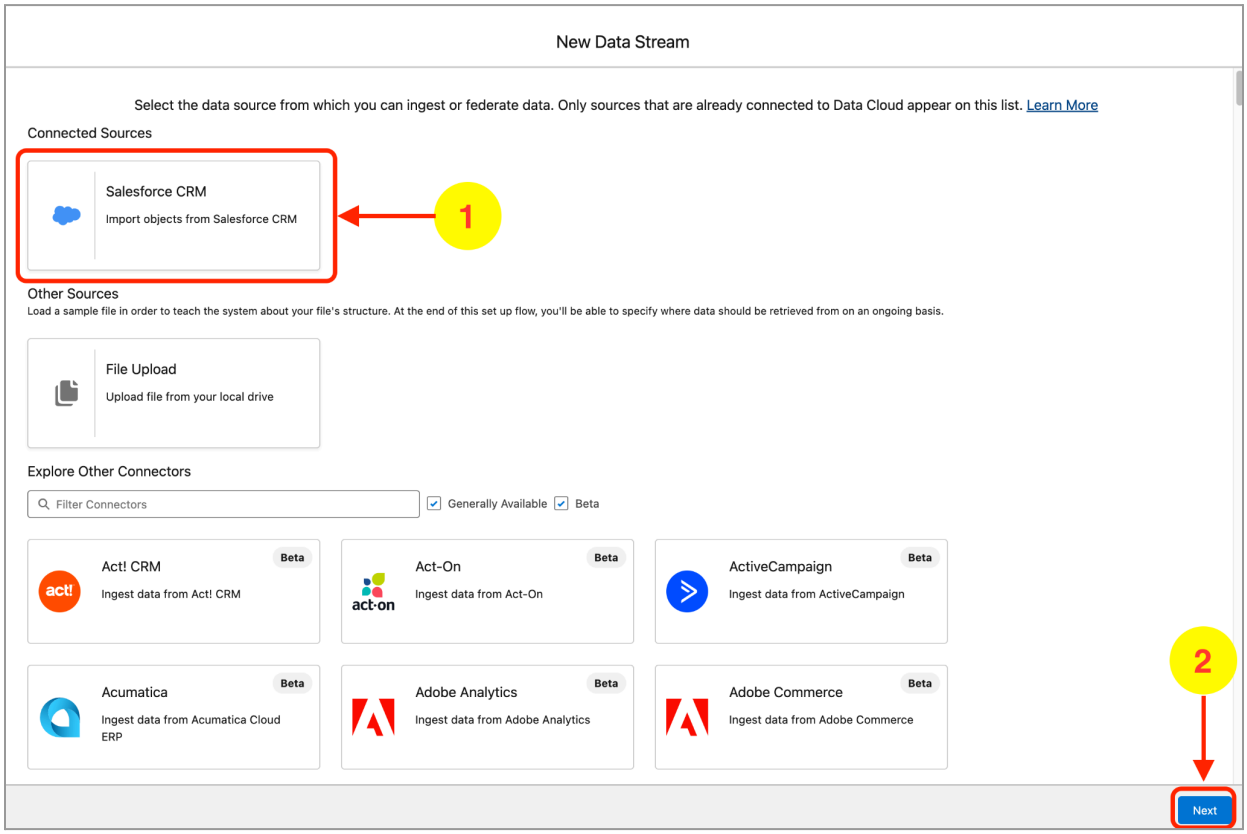
Configure the Data Streams

With the bundles installed, the next step is to set up **Data Streams** so Data Cloud can start ingesting your Salesforce CRM data. Data Streams define which objects (Accounts, Contacts, Leads, Cases, etc.) flow into Data Cloud and keep them updated.

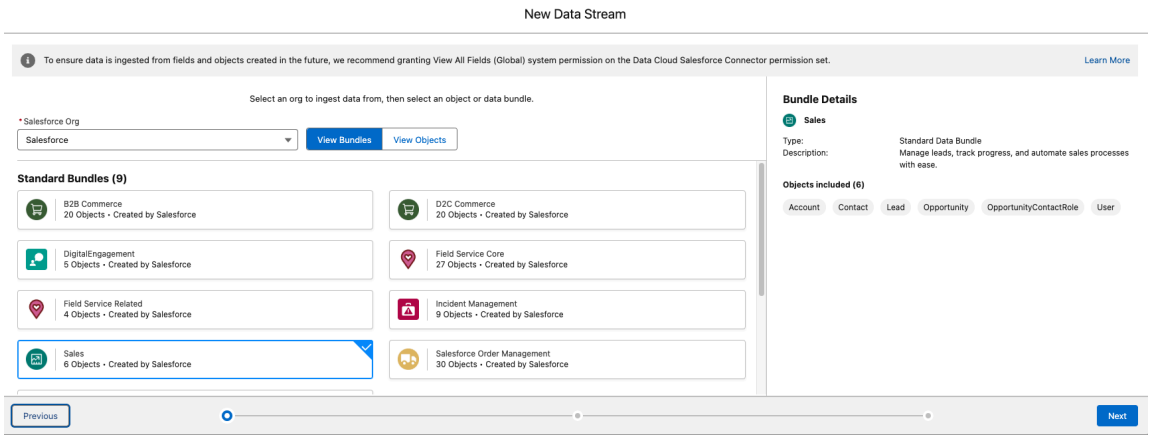
Key Steps:

1. Open **Data Cloud** application in your org.

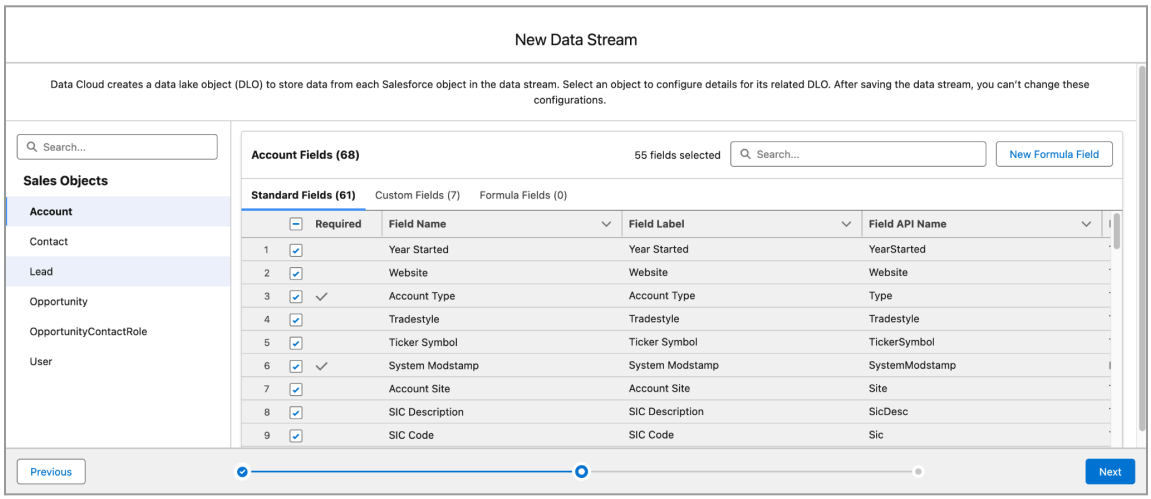
2. Go to the “**Data Streams**” tab → click “**New**”.
3. This action will open the **"New Data Stream"** window
4. Select **“Salesforce CRM”** as your source → Click “**Next**”



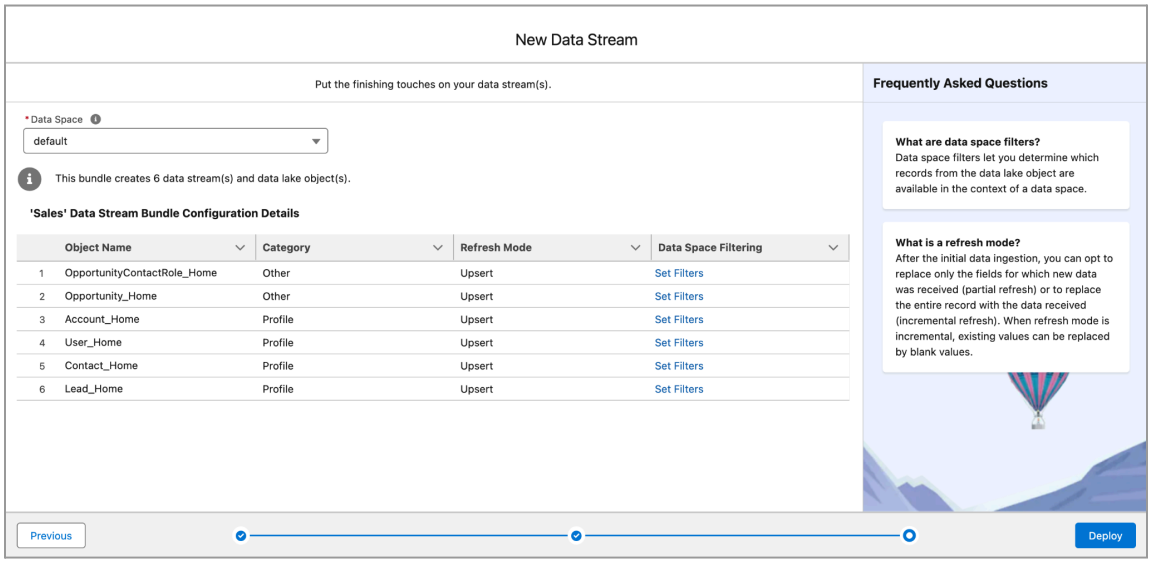
5. Choose the bundle you installed (e.g., Sales Cloud).
    a. These bundles include the key objects you’ll want: Accounts, Contacts, Leads, Cases.



6. While no changes are required, you have the option to configure Formula Fields if needed. Once ready, click **"Next."**



7. On the subsequent page, no modifications are necessary as your Developer organization only contains the **"Default"** data space.
8. Conclude the configuration by clicking **"Deploy."** You can see screenshots illustrating the process of creating a new stream with the Sales bundle below.
   ✅ **Checkpoint:** Your new Data Stream should appear in the list. Repeat the process for each bundle you installed (e.g., Service Cloud).
9. Repeat the process to create data streams for all the bundles you previously configured.
   Note: Additional configuration may be required for the Service bundle. For the Service bundle, you must enable the following before configuring the data stream:
   - **Surveys**
   - The **"Enable skills-based and Directory-to-Agent routing"** omnichannel feature
   - **Email drafts**
   - **Email-to-case**



💡 **Why this matters:** Data Streams are the foundation of Data Cloud ingestion – they ensure your Salesforce CRM data is always accessible, structured, and ready for analysis and activation without custom pipelines.

2. Configure Zero-Copy Data Federation with Snowflake

In the previous step, you configured Data Cloud to ingest Salesforce CRM data–your Accounts, Contacts, Leads, and Cases now flow automatically into a unified data model. But customer data rarely lives in just one system.

Zero-copy federation extends Data Cloud's reach to your data lakes/warehouse such as Snowflake, creating a live connection that queries tables directly without duplicating data. Think of it as creating a window into Snowflake–Data Cloud sees and uses the data as if it lives natively inside Salesforce, but it never leaves your warehouse.

What this proves for your evaluation:

- Data Cloud can unify CRM and data lake/warehouse data without complex ETL pipelines
- You can leverage existing Snowflake investments alongside the CRM data you just configured
- Real-time access to data lake/warehouse data enables current insights, not stale snapshots
- External and native data work together seamlessly in segments and activations

Time estimate: 45-60 minutes
Complexity: Intermediate (requires Snowflake and SQL familiarity)

Business Value: Why Zero-Copy Federation?

Zero copy data query federation with Snowflake offers key advantages when enabled in Data Cloud:

- **No Data Duplication or Movement:** Queries run directly in Snowflake, eliminating bulk transfers, reducing errors, and keeping data accurate at the source.
- **Maximized Existing Investments:** Leverage Snowflake data and infrastructure without duplicate pipelines or extra storage–accelerating time to value and extending ROI.
- **Lower Costs:** Federation queries are far cheaper (70 credits per million records) than batch ingestion (2,000 credits per million), though Snowflake compute/storage costs still apply.
- **Real-Time Access:** Gain secure, near real-time access to current data for better decision-making.
- **Simplified Integration:** Eliminates the need to build and maintain complex sync pipelines, reducing integration overhead and duplicated logic.
- **Native-Like Data Cloud Experience:** Zero-copy Snowflake data behaves like Data Cloud-native data–it can be harmonized, enriched, segmented, and activated. External DLOs simply reference the physical Snowflake data.
- **Consistent Governance:** Data governance applies uniformly across native and zero-copy objects, ensuring a seamless user experience.

Prerequisites Checklist: Before starting, verify you have:

- **Same AWS region:** Your Data Cloud instance and Snowflake account must be in the same region (we'll verify this in Step 1)
- **Snowflake account:** Enterprise edition or trial with ACCOUNTADMIN access
- **Terminal access:** To generate authentication keys (Mac/Linux terminal or Windows PowerShell)
- **SQL familiarity:** Basic comfort running SQL commands in Snowflake worksheets
- **15-20 minutes:** For Snowflake account setup if you don't have one
- If you're new to Snowflake, explore its introductory tutorials here on the website before continuing with this guide as familiarity with the Snowflake environment is a prerequisite for completing the subsequent key steps.

Identify Your Data Cloud Home Region

To complete the zero-copy configuration, ensure your new Data Cloud home and Snowflake instance are in the same AWS region. To find your region:

1. In Data Cloud, navigate to **“Setup Home”**
2. Look for **“Home Org Instance”**
3. In the example below, the instance name is CDP-AWS-PROD8-CACENTRAL1.

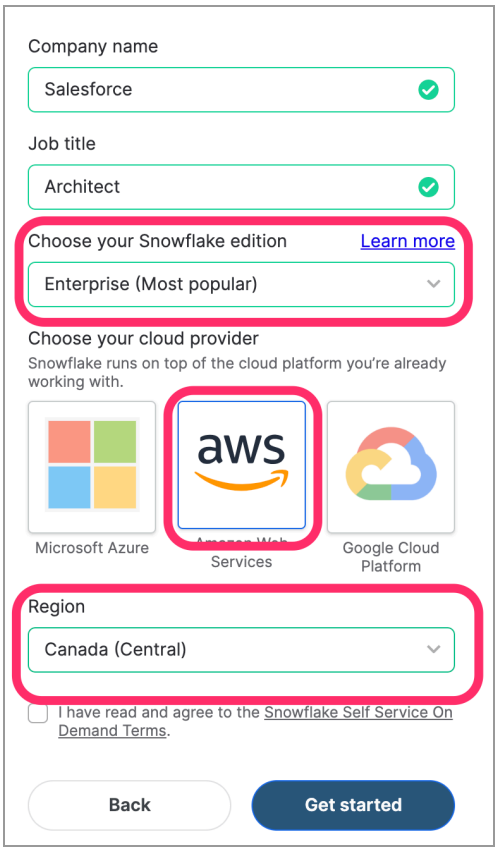
4. Validate the region code using this AWS region mapping site.
5. Using the example above, we can easily determine that the data cloud instance is located in the ca-central-1 AWS region.

Get a Snowflake Account

If you prefer to use your own Snowflake account, you can bypass this step. Otherwise, to sign up for a complimentary 30-day trial, please follow these steps:

1. Open your browser and navigate to www.snowflake.com
2. Click the START FOR FREE button located at the top-right corner of the interface
3. Enter your information into the form
4. Click the CONTINUE button
   You will be moved to the screen where you select the Snowflake edition
5. Select the following values:

Option	Selection
Snowflake Edition	Enterprise
Cloud Provider	AWS
Location	The AWS region you obtained in the previous step.



6. Accept the terms
7. Click the GET STARTED button to continue
8. Skip the questionnaire
    - ☐ Once you have completed this portion of the signup process, Snowflake will send an email to the address entered in the signup form. This is the message allowing you to Activate your new Snowflake account.
9. Log into your email account and find the new message from Snowflake.
10. Click the CLICK TO ACTIVATE button in the email
    This will automatically launch your browser navigating you to a screen where you create your account credentials.
11. Navigate to the **Snowflake Welcome** page
12. Enter a unique **Username** and **Password**
13. Click the Get started button to save the account, and click "Skip for now."

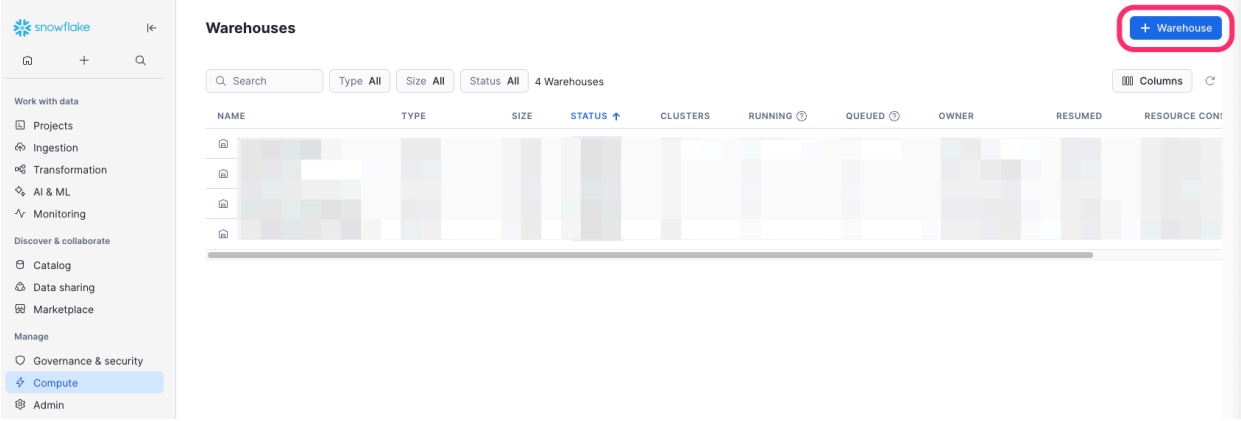
Configure Snowflake

Snowflake warehouses provide compute resources. You'll create one that Data Cloud can access.

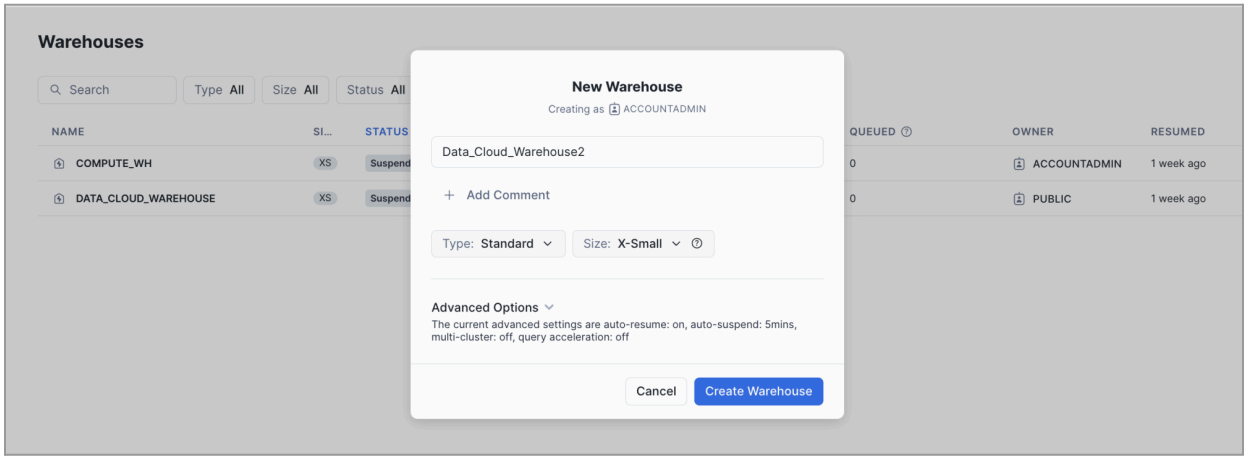
Create the Warehouse

If you are familiar with Snowflake and wish to use your own warehouse for this integration, you may skip this step. Otherwise, please follow these instructions.

1. In Snowflake, navigate to **Compute**, then **Warehouses**
2. Click the **+ Warehouse** button in the top right-hand corner.

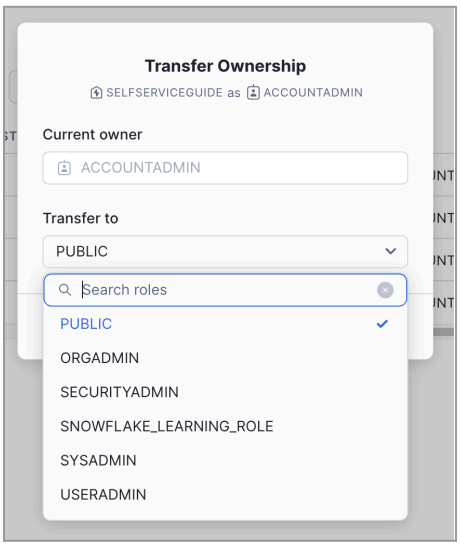


3. Next, name your warehouse and select an appropriate size for your use case→ Click **"Create Warehouse"**



4. Once the warehouse is created, transfer its ownership to a non-administrative role. This allows Data Cloud to access it.
5. To do this, select the three ellipses next to the desired Data Cloud warehouse and choose **Transfer Ownership**.

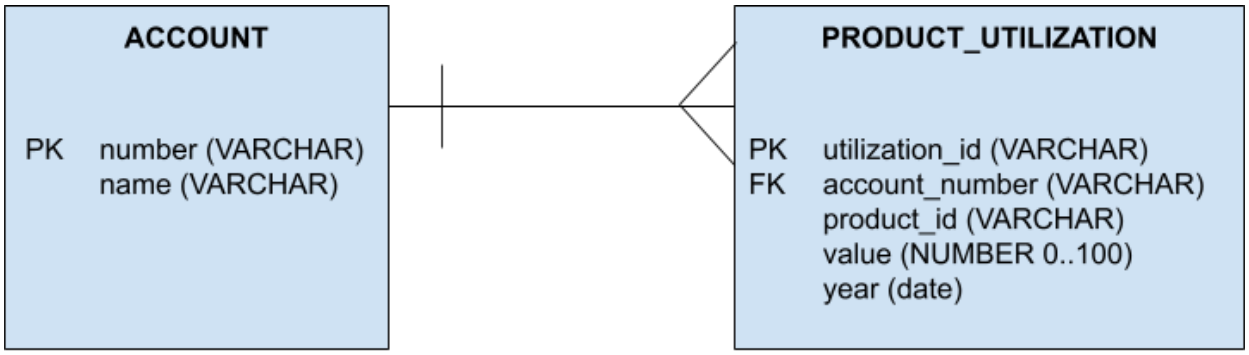
6. In the Transfer ownership dialog box, select the role to which you want to transfer ownership. For example, we'll select the **PUBLIC** role to transfer ownership.



Create Sample Tables

To illustrate the process, we'll use two example tables, which we recommend for understanding the steps involved. Once connections are configured, you can then use your own tables. The rest of the guide assumes that the database name is `SELF_SERVICE_GUIDE`, and the name of the schema is `PUBLIC`. Please adjust accordingly if you are using different names in the following steps.

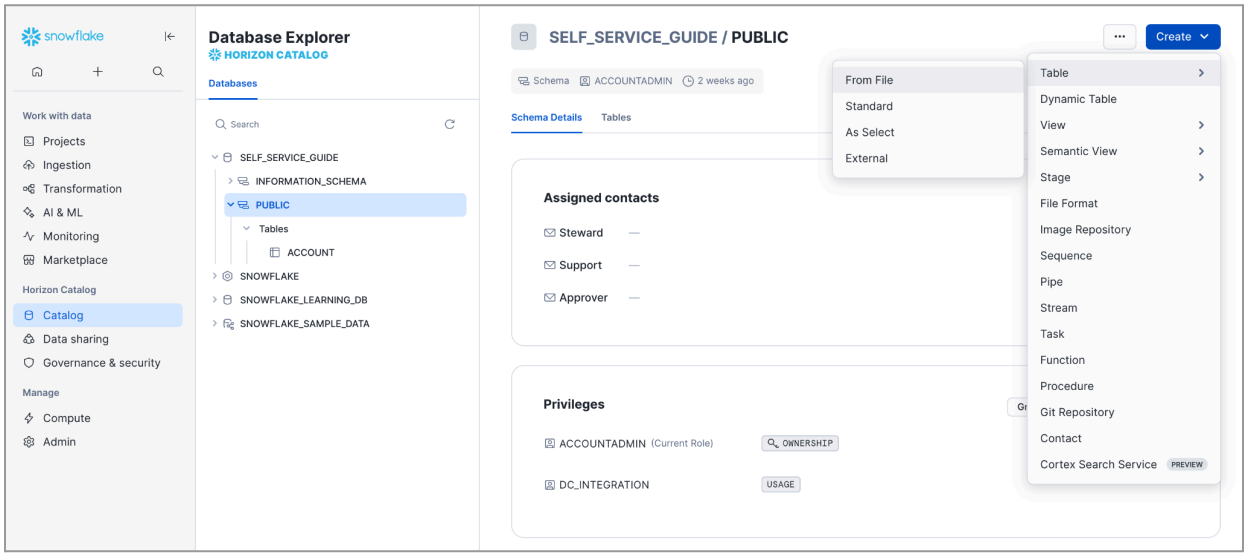
Our data model is very simple and only includes two tables: `Accounts` and `Product Utilization`. For simplicity, each account can only have one product utilization per product. The utilization is measured as a percentage, meaning how much of a product an account is using. The following graph represents the 1:N relationship between the tables `ACCOUNT` and `PRODUCT_UTILIZATION`:



For simplicity, we will omit the third table, which would contain product information, from this exercise.

Create the tables:

1. Download the sample data:
   - ACCOUNT.csv
   - PRODUCT_UTILIZATION.csv
2. In Snowflake, navigate to your schema
3. Choose **Create > Table > From file**, as shown below:



4. Using the files downloaded before, create two tables called `ACCOUNT` and `PRODUCT_UTILIZATION`.
5. Use the data model above to check that all field names and types are the right ones.

Generate a Public-Private Key Pair

You'll create a public/private key pair. Data Cloud stores the private key securely; Snowflake verifies using the public key.

Generate the keys:

1. Open your terminal
2. Run this command to create a private key.

```
None
openssl genrsa 2048 | openssl pkcs8 -topk8 -inform PEM -out rsa_key.p8 -nocrypt
```

The command will generate a file called `rsa_key.p8` (your private key–keep this secure!)

3. Generate the corresponding public key:

```
None
openssl rsa -in rsa_key.p8 -pubout -out rsa_key.pub
```

This creates `rsa_key.pub` that contains the private key.

Create an Integration User in Snowflake

Next, Zero-copy federation requires a dedicated Snowflake user with key-based authentication. This user acts as Data Cloud's identity when querying Snowflake. Create an integration user in Snowflake and assign them the public key from your newly generated public and private key pair:

1. Navigate to **Projects** then **Worksheets** in Snowflake.
2. Select the **+** Icon in the top right-hand corner
3. Choose **SQL Worksheet**.

To create an integration user and assign the System Administrator role, follow the next steps:

Step 1:Prepare Your Public Key

Before creating the user, you need the public key value from the previous step.

1. Open `rsa_key.pub` in a text editor (TextEdit, VS Code, Notepad++, etc.)
2. Copy **only** the text between these delimiters: `-----BEGIN PUBLIC KEY-----` and `-----END PUBLIC KEY-----` delimiters.
3. This is the public key value that you'll use later. .

Step 2: Create the User Account

1. Execute the following statement in your **SQL Worksheet**.
2. This statement creates the new user `dc_fed_user` and assigns them the `SYSADMIN` role by default.
3. The empty strings ('') can be filled in with specific values if needed for your system, but they aren't required for a basic integration user.

```sql
CREATE OR REPLACE USER dc_fed_user
  PASSWORD = ""
  LOGIN_NAME = ""
  DISPLAY_NAME = ""
  FIRST_NAME = ""
  LAST_NAME = ""
  EMAIL = ""
  DEFAULT_ROLE = SYSADMIN;
```

Step 3: Set the RSA Public Key

1. Execute the following statement in your SQL Worksheet. This statement links the public key you copied in the first step to the new `dc_fed_user`.
2. Replace `<PUBLIC_KEY>` with the value you copied from the file in step 1. **Make sure not to include** the `-----BEGIN PUBLIC KEY-----` or `-----END PUBLIC KEY-----` delimitators.

```sql
ALTER USER dc_fed_user SET RSA_PUBLIC_KEY='<PUBLIC_KEY>';
```

What this does: Creates user `dc_fed_user` with the SYSADMIN role as default. This user can't log in yet–we'll add authentication next.

Step 4: Check the user creation and configuration

The following `DESC USER` statement will show the user properties. The property `RSA_PUBLIC_KEY` should contain the value that you just assigned to it in the previous step.

```sql
DESC USER dc_fed_user;
```

Step 5: Grant the right privileges to the integration user

In order to allow Data Cloud access to the Snowflake tables that you want to integrate, you need to grant the proper privileges to the integration user. Using the least privilege access principle, you can create a role in Snowflake, grant that role to the integration user, and grant the following privileges to the role:

- **`USAGE` on the database**: This allows the role to access the database and its metadata. Without this, the user can't even "see" that the database exists.
- **`USAGE` on the schema**: This allows the role to access the schema within the database. It's a prerequisite to accessing any objects, like tables, within that schema.
- **`SELECT` on the tables**: This is the specific privilege that allows the user to query and retrieve data from the table. Without `SELECT`, they can see the table exists (from the `USAGE` grants), but they cannot access its contents.

Here's an example illustrating how to grant privileges. We'll use the database `SELF_SERVICE_GUIDE`, the role `DC_INTEGRATION`, and the tables `ACCOUNT` and `PRODUCT_UTILIZATION` in the `PUBLIC` schema. You can substitute these names with your preferred ones.

This example demonstrates assigning the necessary privileges to the `DC_INTEGRATION` role for the `ACCOUNT` and `PRODUCT_UTILIZATION` tables, located in the `PUBLIC` schema within the `SELF_SERVICE_GUIDE` database:

```sql
-- Step 1: Grant USAGE on the database to the role DC_INTEGRATION
GRANT USAGE ON DATABASE SELF_SERVICE_GUIDE TO ROLE DC_INTEGRATION;

-- Step 2: Grant USAGE on the schema to the role DC_INTEGRATION
GRANT USAGE ON SCHEMA SELF_SERVICE_GUIDE.PUBLIC TO ROLE DC_INTEGRATION;

-- Step 3: Grant SELECT on the table to the role DC_INTEGRATION
GRANT SELECT ON TABLE SELF_SERVICE_GUIDE.PUBLIC.ACCOUNT TO ROLE DC_INTEGRATION;
GRANT SELECT ON TABLE SELF_SERVICE_GUIDE.PUBLIC.PRODUCT_UTILIZATION TO ROLE
DC_INTEGRATION;
```

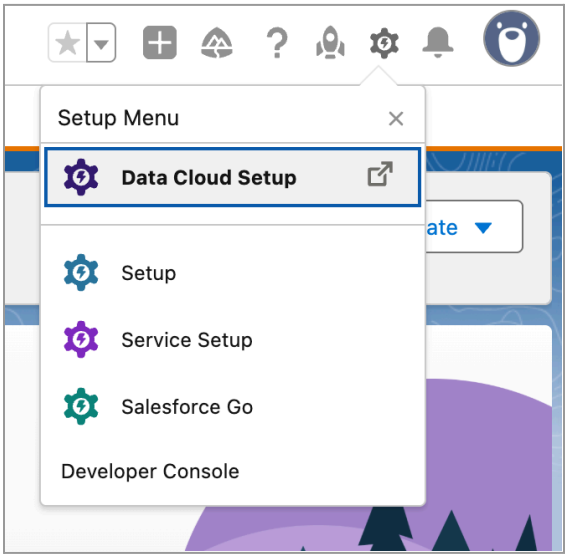
```
-- Step 4: Grant DC_INTEGRATION to the integration user dc_fed_user
GRANT ROLE DC_INTEGRATION TO USER dc_fed_user;
```

Configure Data Cloud to Snowflake

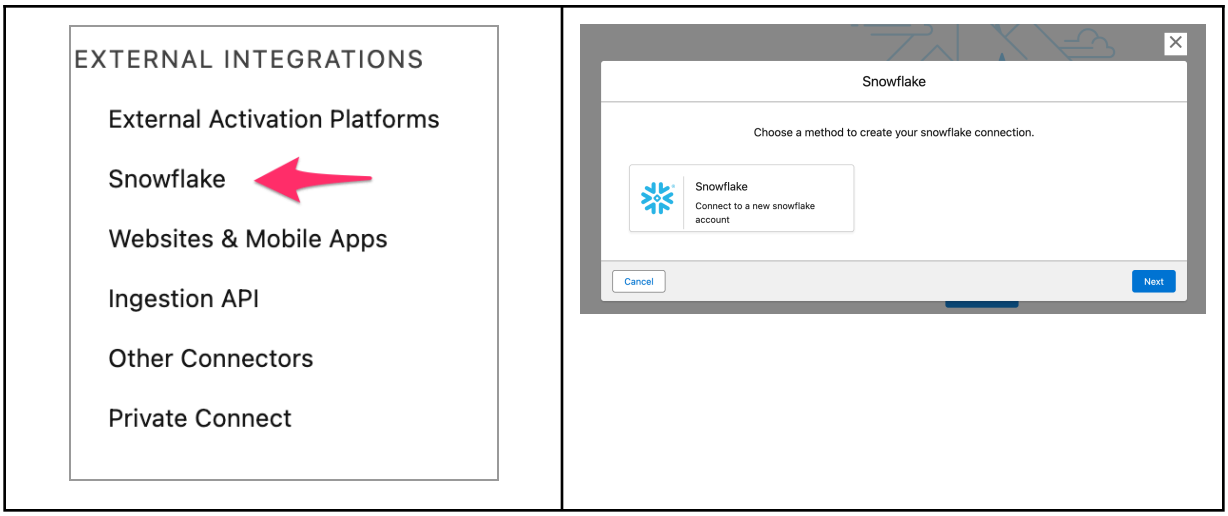
Step 1: Setup Snowflake Connection

Now you'll tell Data Cloud how to reach Snowflake using the integration user you just created. To set up a zero-copy integration with Snowflake, begin by configuring the connection vua the following steps:

1. In Salesforce, click the gear icon (⚙️) → Setup
2. In the Quick Find box, type "Data Cloud Setup"



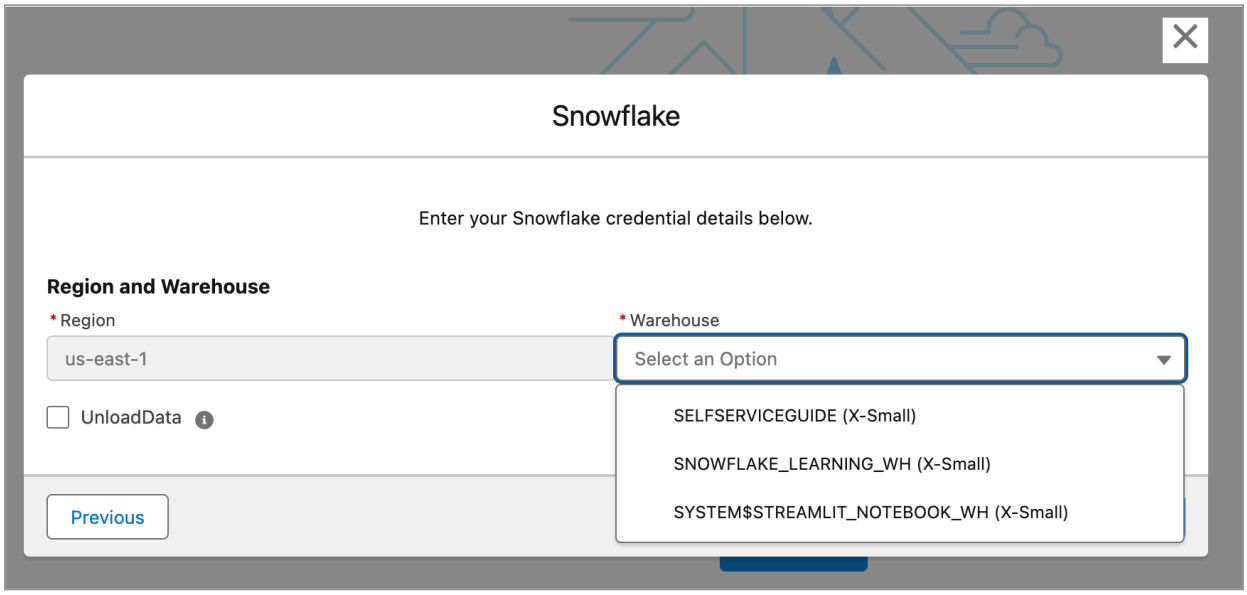
3. Navigate to EXTERNAL INTEGRATIONS → Snowflake Navigate to **"EXTERNAL INTEGRATIONS",** select **"Snowflake"** and then click **"New"** on the subsequent page.
4. Choose **Snowflake** from the options.



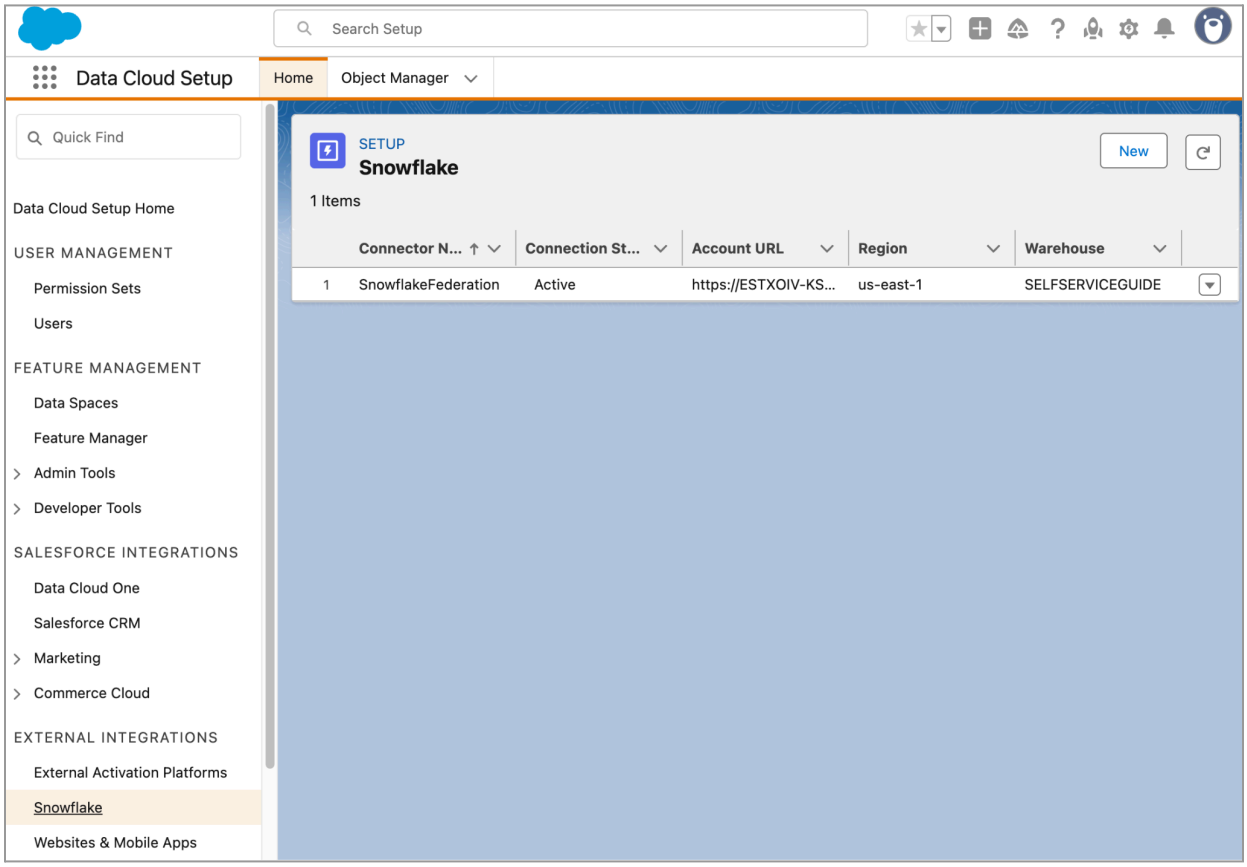
In the configuration window, enter a Connection Name and Connection API Name (you can keep the default). Enter your Snowflake account URL. Then enter the username of your integration user and the private key.

- Account URL: Find it under the account details (bottom left corner of the Snowflake menu). Prefix it with `https://`
- Username: The username you created in the previous step. In our example above, it is `dc_fed_user`
- Private key:
    - Open `rsa_key.p8` in a text editor
    - Copy the **entire contents**. Do not include the delimitators, only the text between `-----BEGIN PRIVATE KEY-----`- and `-----END PRIVATE KEY-----` .

If the connection is successful, you should be able to choose the warehouse you want to use on the next screen, as shown below. In our example, we'll use the warehouse called **SELFSERVICEGUIDE** we created before.



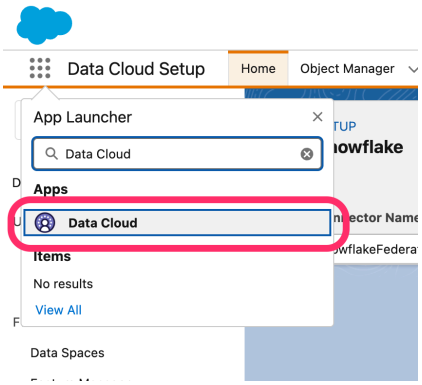
After configuration, the connection setup information appears in the Data Cloud Setup interface, as shown below.

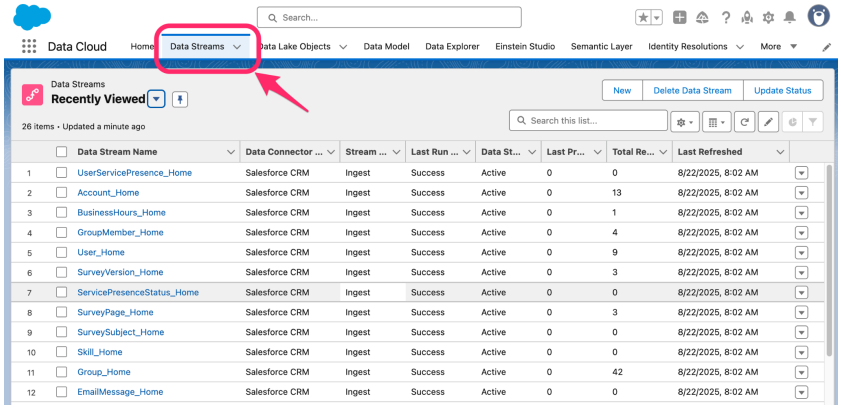


Step 2: Configure Snowflake Data Stream

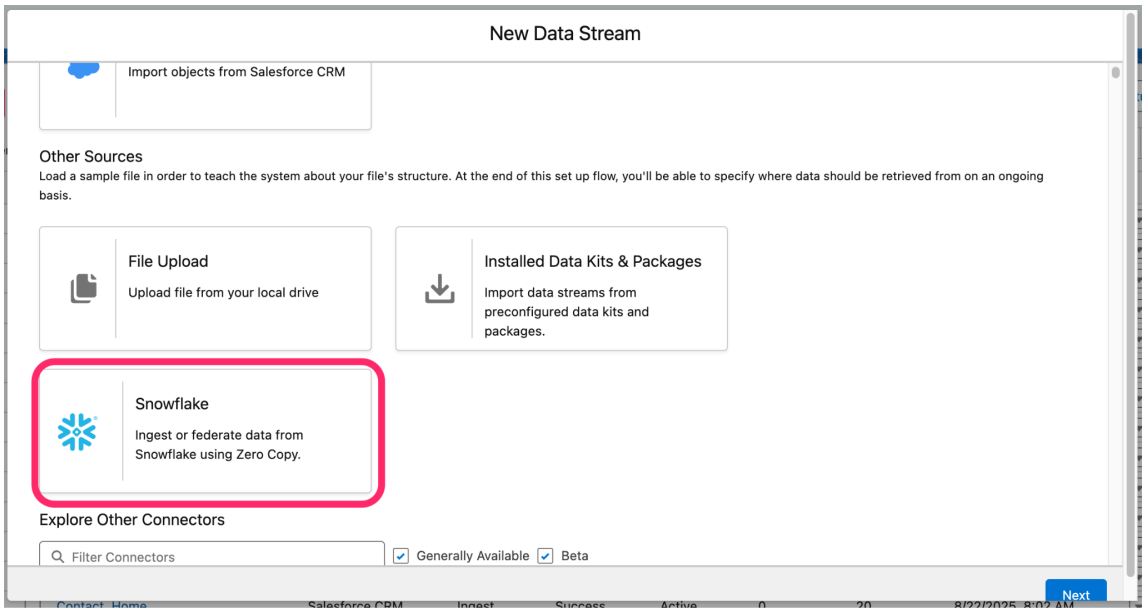
After activating the Snowflake connection, create a Snowflake Data Stream. Data Streams define which Snowflake tables Data Cloud can access and how they map to the Data Model. This can be by the following steps:

1. Navigate to the **"Data Streams"** tab within the **"Data Cloud"** application.

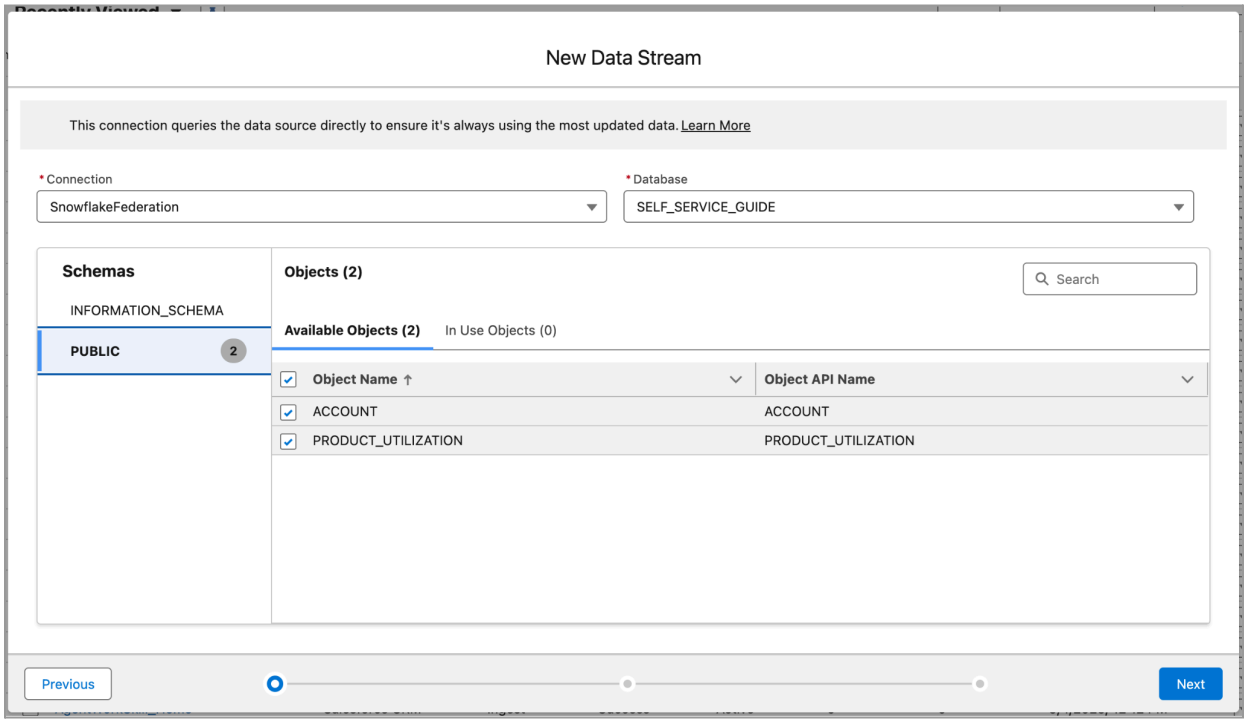




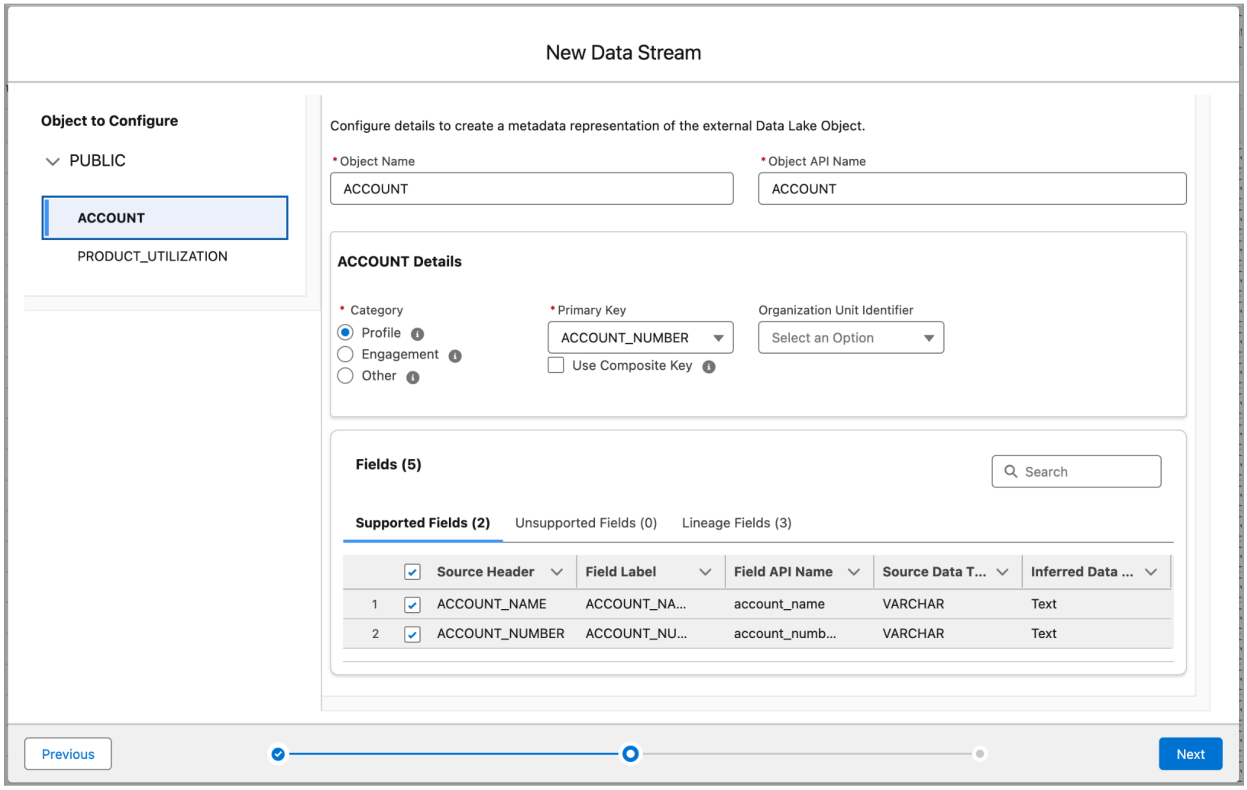
2. In the Data Streams tab, choose **“New”**.
3. A **“New Data Stream”** window will pop up.
4. Scroll down until you see the **“Snowflake”** source, under **“Other Sources”**.
5. Select **“Snowflake”**
6. Click “**Next”**.



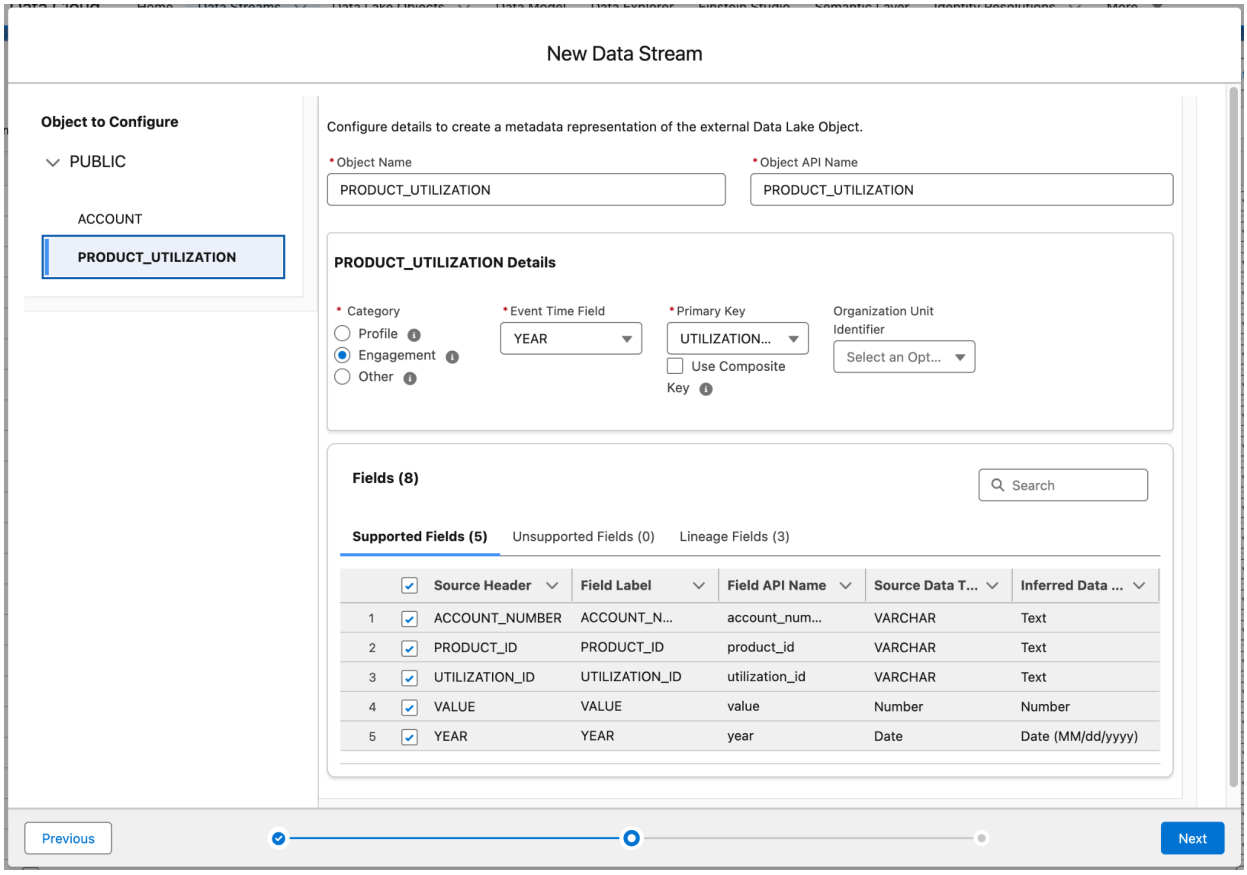
In the next window, choose the connection established in the previous step and the desired database. The dropdown list will display databases accessible to the previously configured integration user. From the tables your integration user can access, select the one you wish to integrate, and click “Next”. In our case, we will choose the `ACCOUNT` and `PRODUCT_UTILIZATION` tables.



The `ACCOUNT` table is the type Profile, and its primary key is `ACCOUNT_NUMBER`. Choose all the supported fields, marking the checkbox, as shown below:



The `PRODUCT_UTILIZATION` object is classified as an Engagement type. Its primary key is `UTILIZATION_ID`, and the `Event Time Field` is `YEAR`. Select all supported fields by checking the corresponding boxes as illustrated below:



After configuring the two objects, click **“Next”.**

On the next screen, choose the dataspace in which the object will reside.

A Note About Zero Copy Acceleration

Zero Copy Acceleration is an optional feature that caches Snowflake data in Data Cloud. When enabled, frequently accessed data is stored temporarily, reducing repeated queries to Snowflake and improving response times.

Enable acceleration when:

- You'll query this data **>10 times per hour**
- Query response time matters (real-time segmentation, dashboards)
- The same queries run repeatedly (e.g., batch processes, reports)

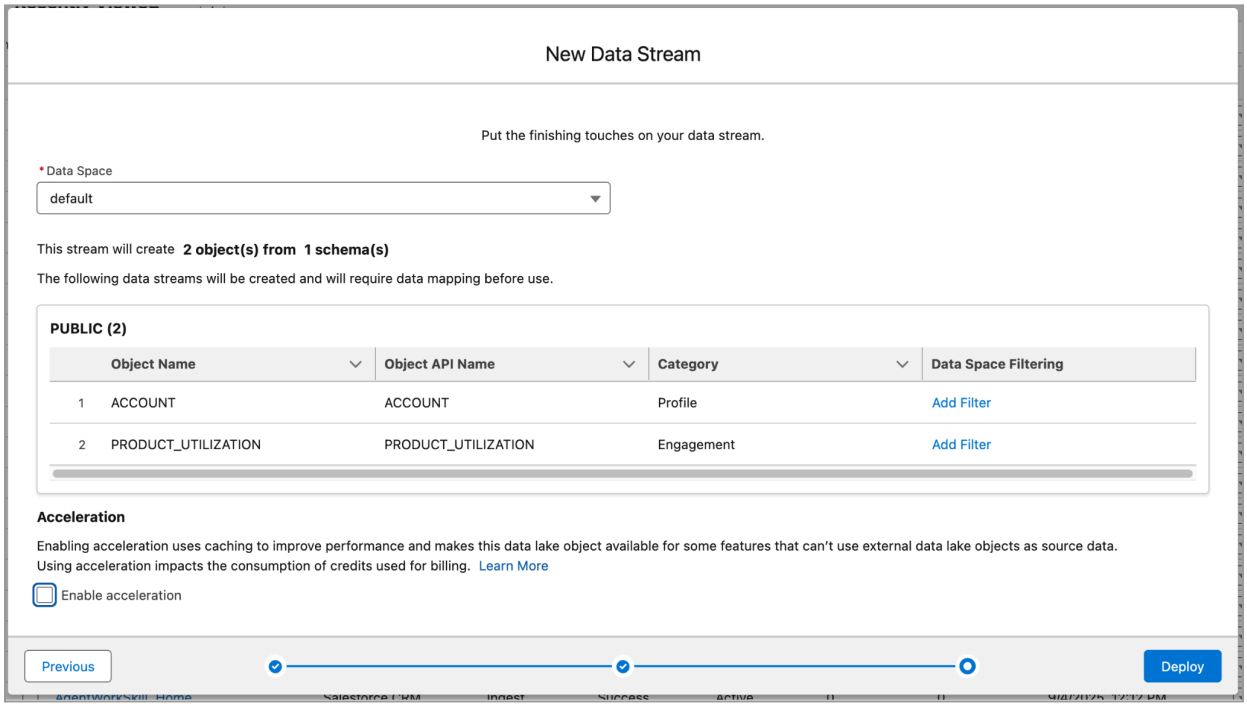
Skip acceleration when:

- Queries are infrequent or ad-hoc

- Data changes constantly and must be real-time
- You're minimizing Data Cloud credit consumption

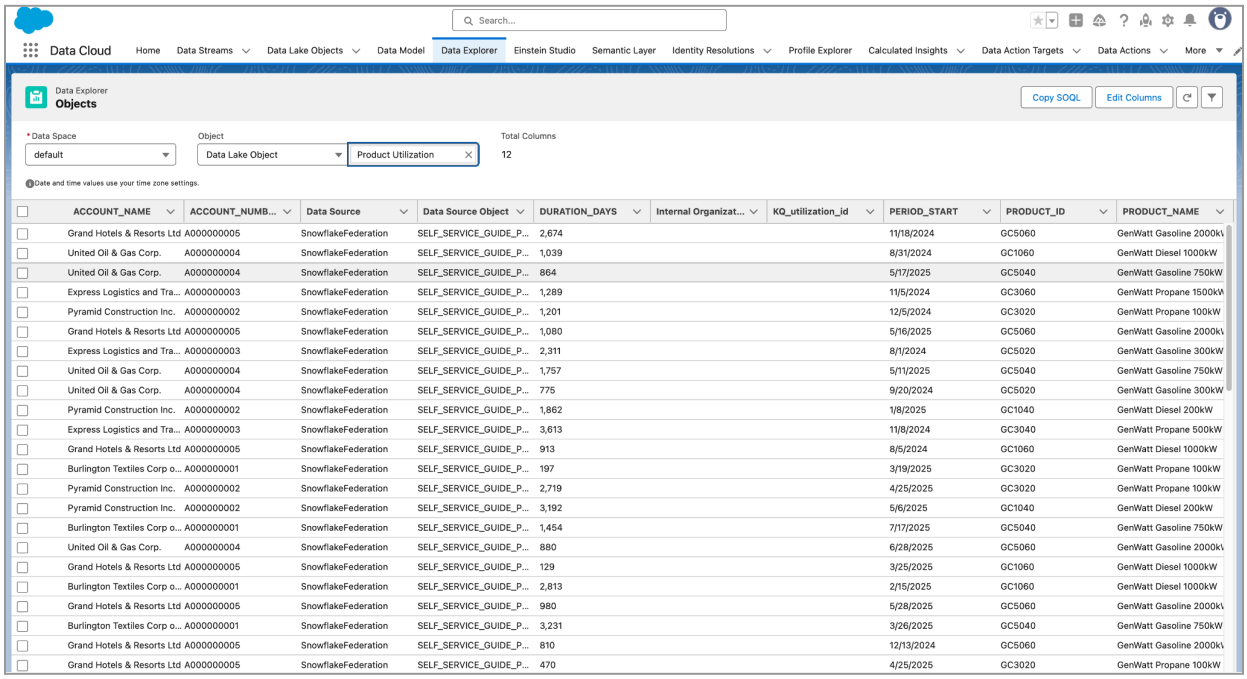
For this evaluation: We recommend starting **without acceleration** to see baseline zero-copy performance. You can enable it later if query speeds become an issue.

If you enable acceleration, you'll need to set refresh frequency (how often Data Cloud updates the cache from Snowflake).



Once you have completed the configuration according to your requirements, click **"Deploy"**.

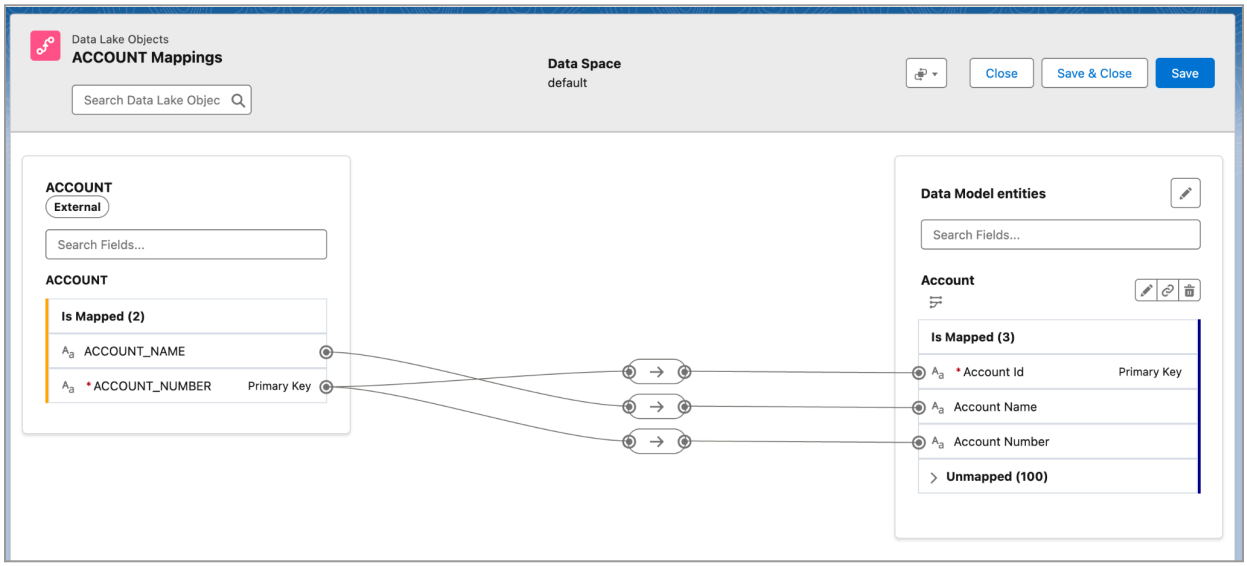
An easy way to confirm the configuration is successful is to use the Data Explorer and inspect the Data Lake Objects that were created during the zero copy configuration.



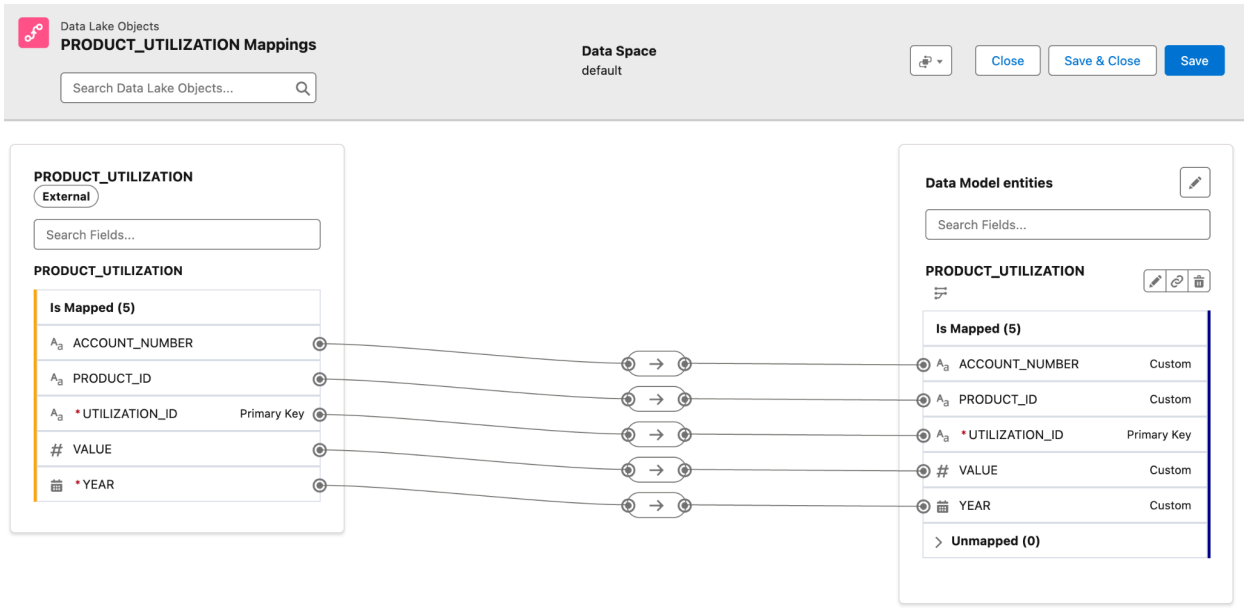
Step 3: Mapping of the DLOs to DMOs

Map the fields of the `ACCOUNT` object to the canonical DMO `Account` as shown in the following image:

- `ACCOUNT_NAME` maps to `Account Name`
- `ACCOUNT_NUMBER` maps to both `Account Id` and `Account Number`



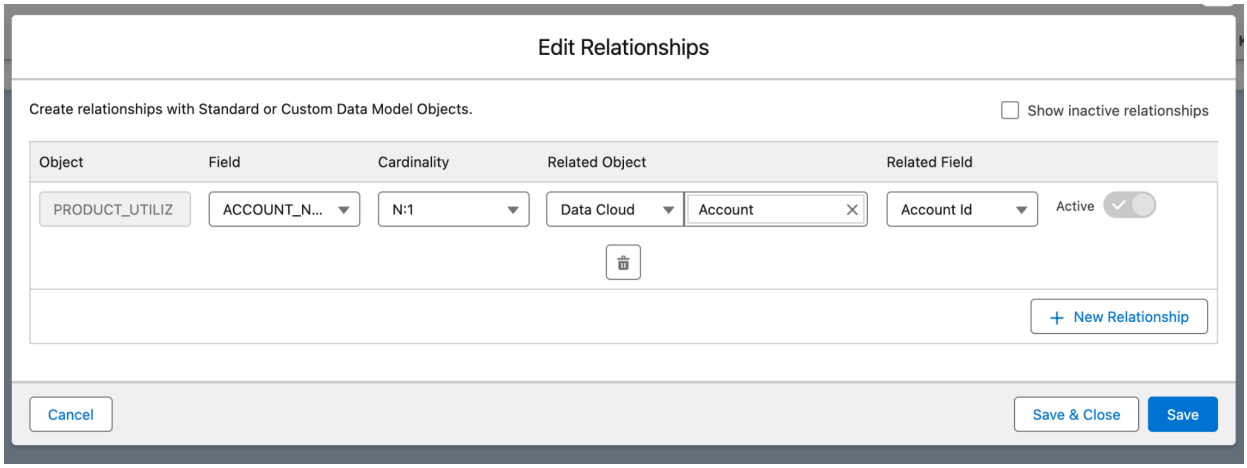
Map The fields of `PRODUCT_UTLIZATION` to a custom object called `PRODUCT_UTILIZATION`, as shown in the following image:



Step 4" Create DMO Relationships

Link the two DMOs so Data Cloud understands the relationship between accounts and their utilization records.

1. Go to "**Data Model"**→ Find the `PRODUCT_UTILIZATION` (the custom DMO) object.
2. Click on the "**Relationships"** tab.
3. Click "**New Relationship"**
4. Configure:
    a. Related Object `Account(the standard DMO)`
    b. Cardinality: Many-to-One (N:1)
        i. Many utilization records can belong to one account
5. **Source Field:** `ACCOUNT_NUMBER` (in PRODUCT_UTILIZATION)
6. **Target Field:** `Account Id` (in Account)



What this enables: You can now create segments like "Accounts with product utilization above 80%" or join CRM account data with Snowflake usage data in calculated insights.

Verify Your Complete Setup

- Connection is active
- Data Lake Objects show records
- Field mappings are saved
- Relationship is defined

Test End-to-End

1. Navigate to **Data Explorer**
2. Select the `PRODUCT_UTILIZATION` Data Lake Object
3. Click into a record

Expected result: You should see all fields populated with values from Snowflake.

Phase 2: Transform

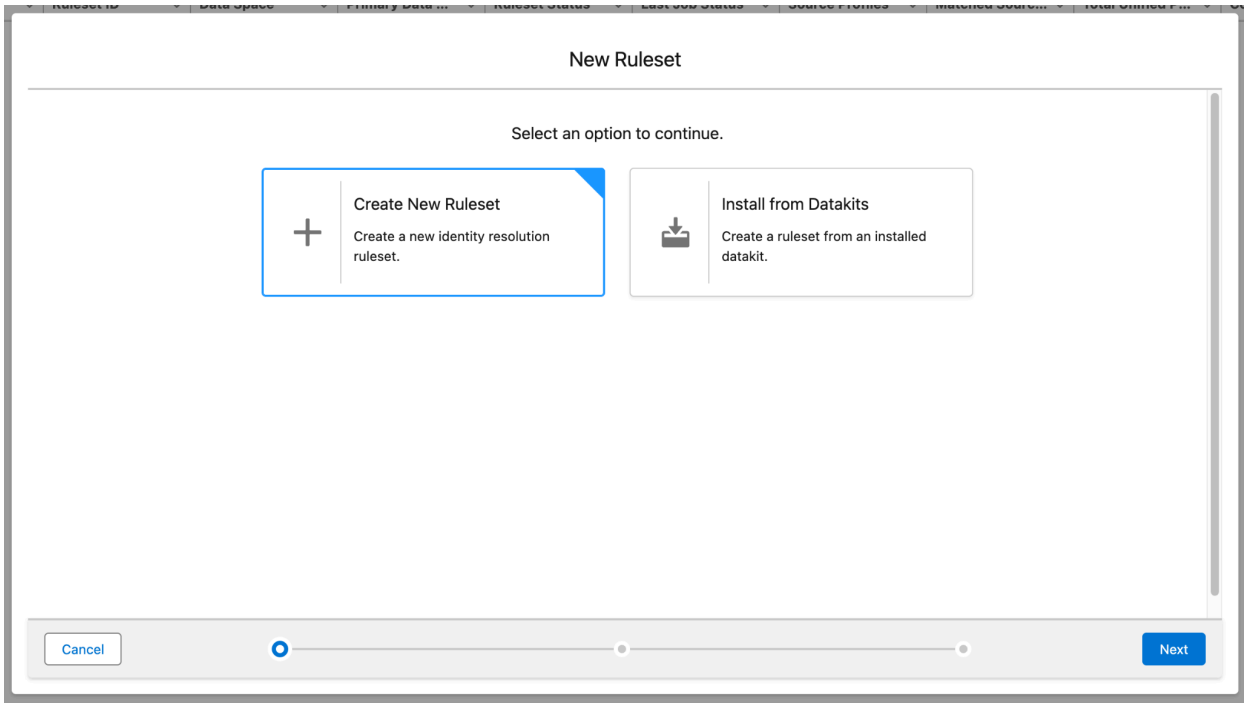
In this section we will review some of the Data Cloud data transforming capabilities, including the crucial Identity Resolution process.

Configure Identity Resolution

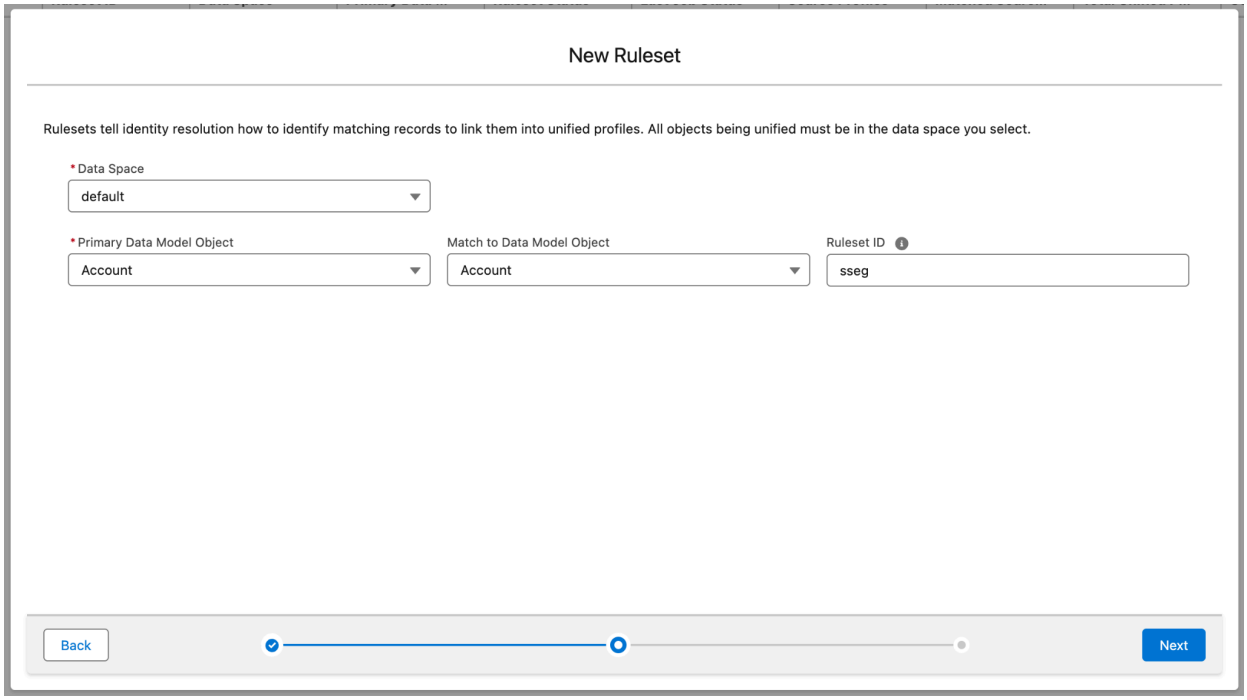
In Data Cloud, **identity resolution** is a two-part process that unifies customer data from diverse sources. It uses **matching rules** to link individual records into a **unified profile** and **reconciliation rules** to summarize attributes. This process creates a **single, comprehensive**

view of each customer, acting as a "key ring" that maintains full data lineage from all contributing systems. It is crucial for enabling accurate segmentation, personalized experiences, and informed business decisions across the organization.

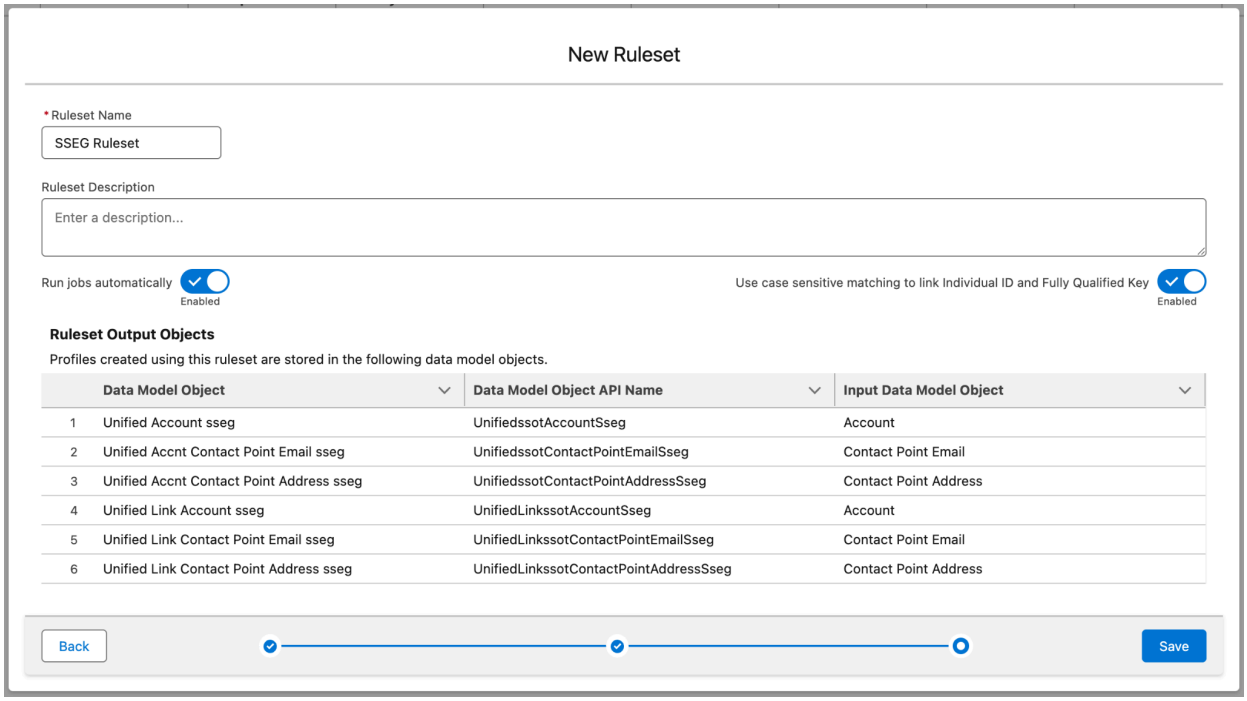
To start an Identity Resolution process in Data Cloud, go to "Identity Resolutions," select "New," and then "Create a New Ruleset" before clicking "Next." This will combine Account data from Salesforce and Snowflake, and the unified identities will be used for customer-facing data activations.



In the next screen, leave the Data Space as “default” and choose the `Account` object as both the Primary and the Match Data Model Object. Choose a four character Ruleset ID. This ID will be used to differentiate the unified objects created by this ruleset. In the example below, we used “sseg”. When you are ready, click “Next”.



In the next screen, choose a name for your Ruleset. The description is optional. The screen shows the names of the objects that will be created as part of the identity resolution process. In the following example, we used the name “SSEG Ruleset” as the Ruleset Name.



Configure a Match Rule

A match rule is a fundamental component of the identity resolution process, designed to determine which individual or account profiles from various data sources should be grouped together to form a unified profile.

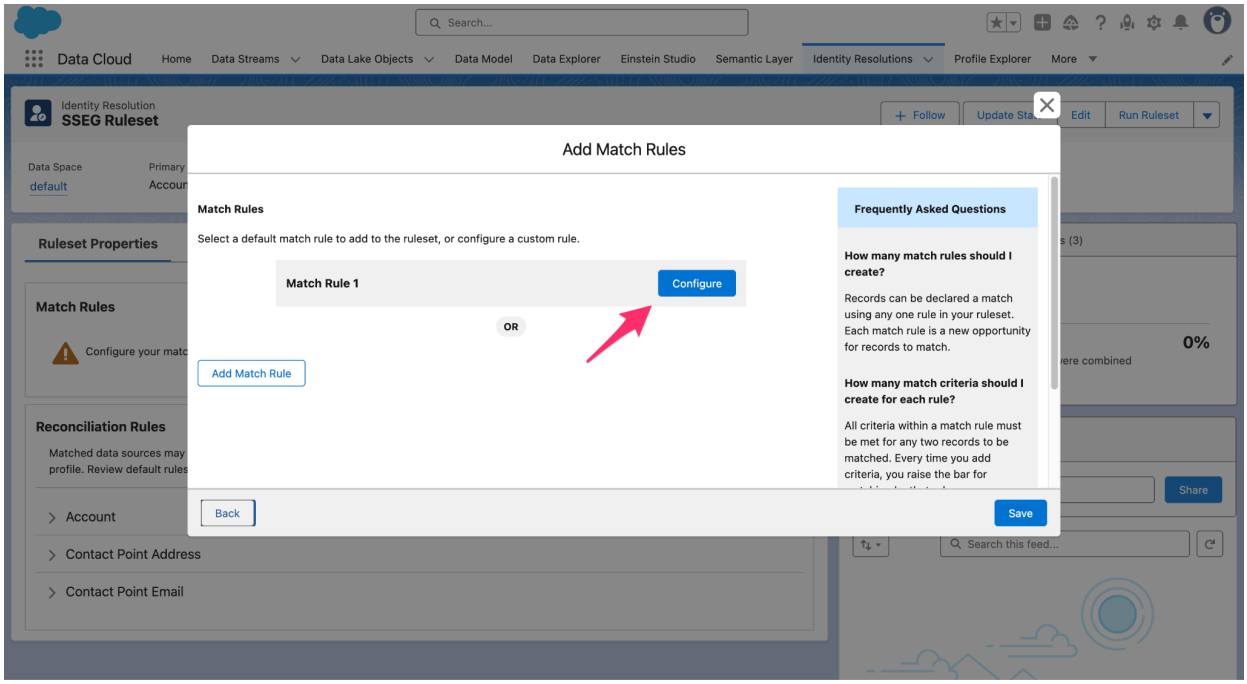
Match rules define the specific criteria that link individual records from different data sources. When at least one match rule within a rule set is satisfied by two or more profiles, those profiles are identified as belonging to the same individual or account and are subsequently linked together. This process is crucial for creating an accurate and cohesive view of your customer.

Data Cloud supports several types of matching criteria to allow for varying degrees of precision and flexibility:

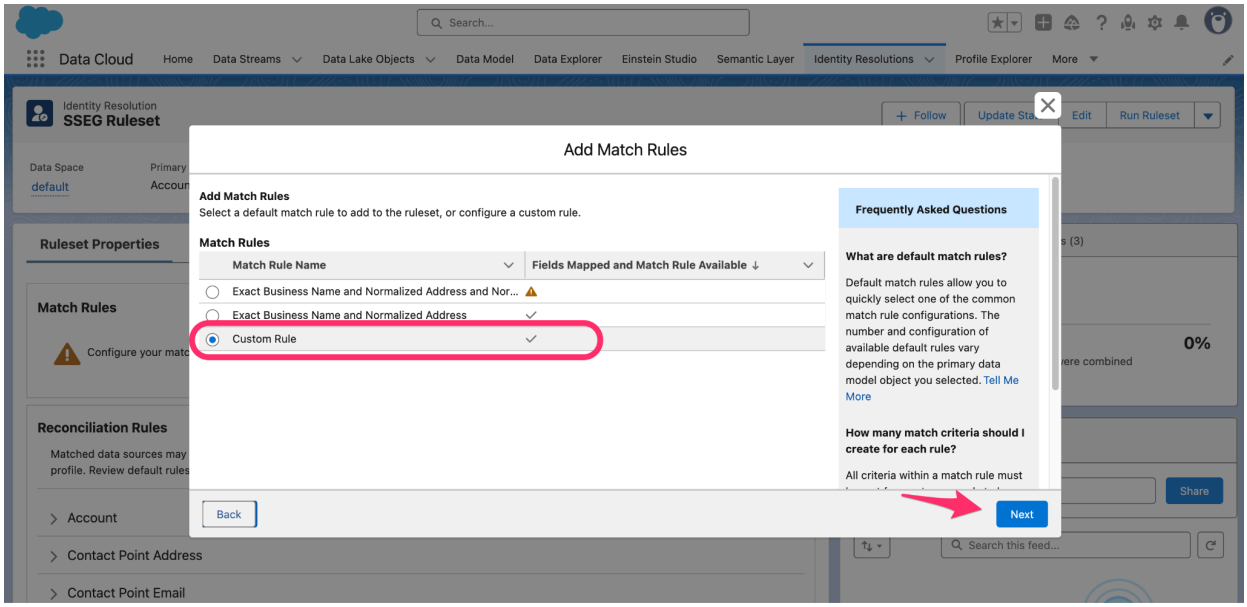
- **Exact Match:** This requires an exact textual match on attributes such as phone numbers, email addresses, names, or mailing addresses.
- **Normalized Match:** This helps to fix certain formatting problems in data input, allowing for more flexibility in match rules. It is supported for basic identifiers like phones, email addresses, and mailing addresses.
- **Fuzzy (Probabilistic) Match:** This type of matching is currently supported on first names and uses machine learning and statistics to identify similar records with a high degree of probability. The lower the precision in fuzzy matching, the lower the statistical similarity will be between two different names.
- **Party Identification:** This is a highly precise method that uses real identifiers (e.g., golden record IDs, ERP IDs, government IDs, customer IDs, subscriber keys) rather than relying solely on basic individual attributes like name and address. It is configured during the harmonization phase by mapping key identifiers to the Party Identification DMO, requiring fields like identification name, type, and number.

In the following example we will use a straightforward match rule that identifies accounts by exact matches on their account name.

In Data Cloud, choose the Identity Resolution Ruleset that we created before, and click “Configure”, and then “Next”. In the following screen, choose the option “Configure”.



In the next screen, choose the option “Custom Rule”, and click “Next”.



Next, choose a name for the rule and choose the following values for the Match Criteria:

- Data Model Object: Account
- Field: Account Name
- Match Method: Exact

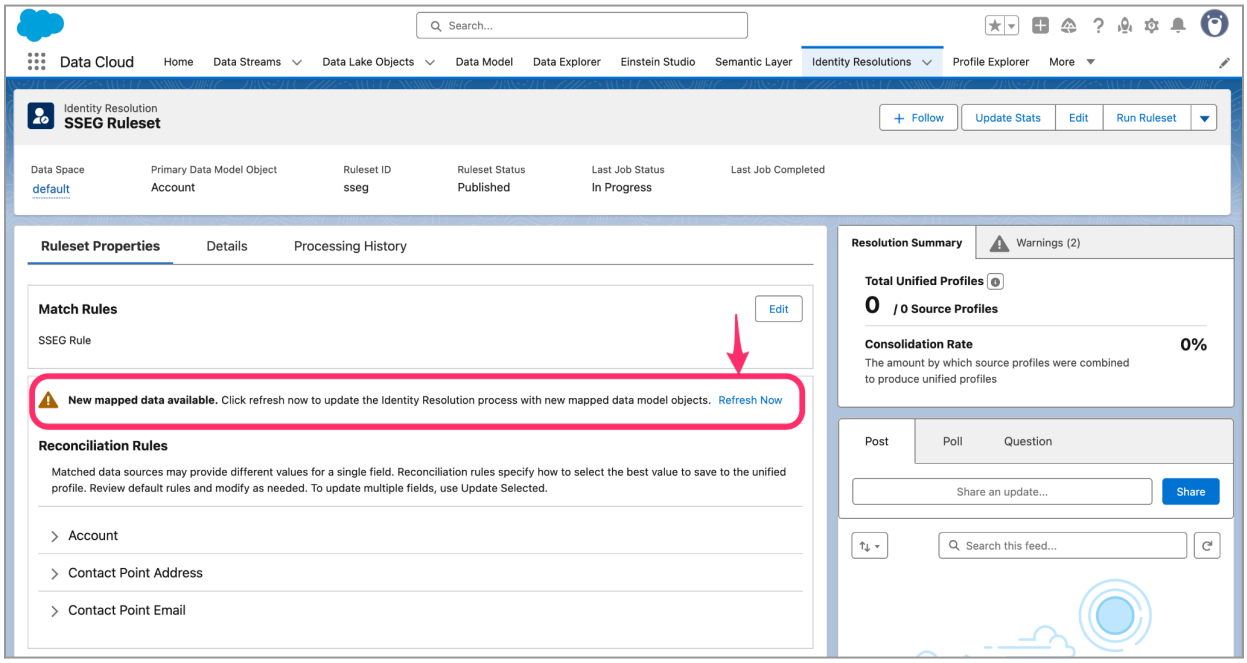
Finally, click “Next” and “Save”. That will start the publishing process, that can take a couple of minutes to complete.

In the meantime, we need to map some additional fields. In order to do that, use the “Data Lake Objects” tab to find the object `Account_Home`, click “Review” in the “Data Mapping” section, and add the following mappings:

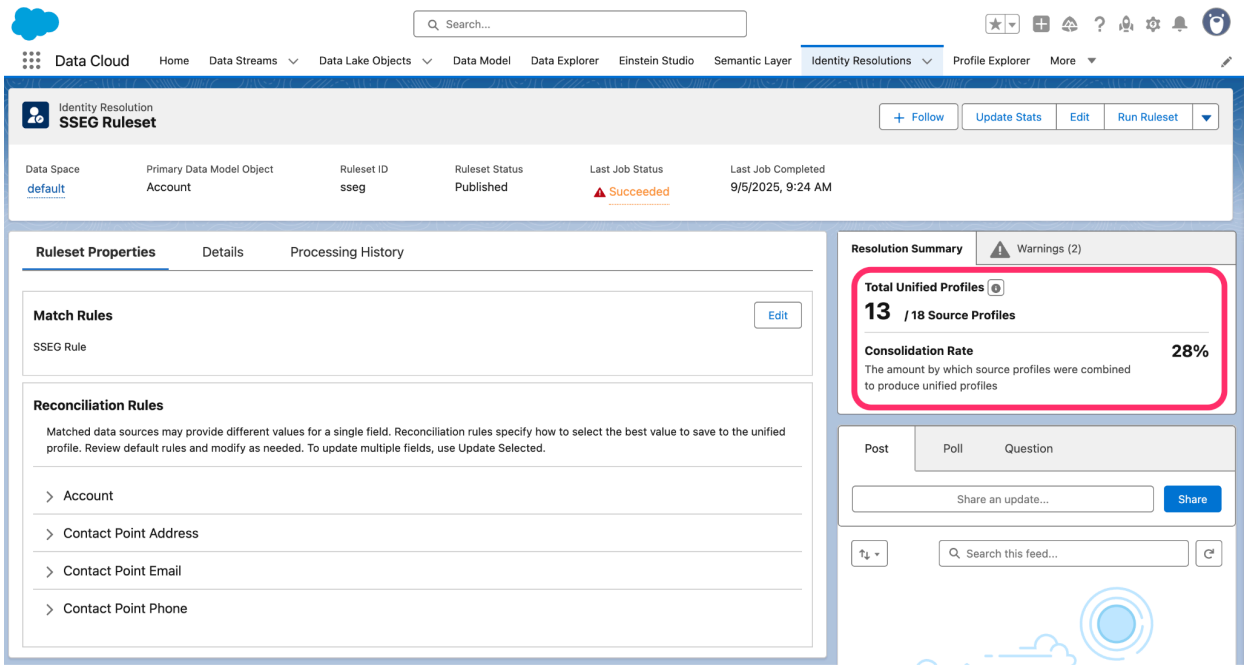
- Map “Account ID” to the field “Party” in “Contact Point Address”.
- Map “Account ID” to the field “Party” in “Contact Point Phone”.
- Map “Account Phone” to “Formatted E164 Phone Number”.

Click “Save and Close”, to store the changes.

We need to apply the changes to the identity resolution process. In order to do that, go back to the Identity Resolution ruleset. You should have an alert asking to refresh the process. Click on “Refresh Now”, and wait until the process is complete. The “Last Job Status” will change from “In Progress” to “Succeeded”



Once the process is complete, you should be able to see the Total Number of Unified Profiles, as shown below. You can ignore any alert message.



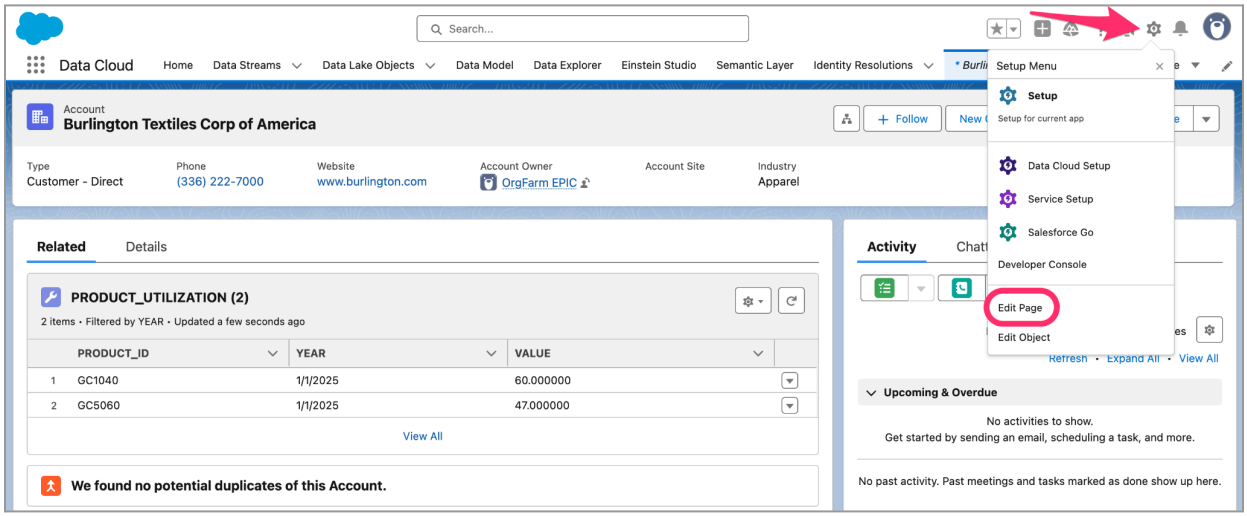
Phase 3: Action

Configure a Data Cloud Related List

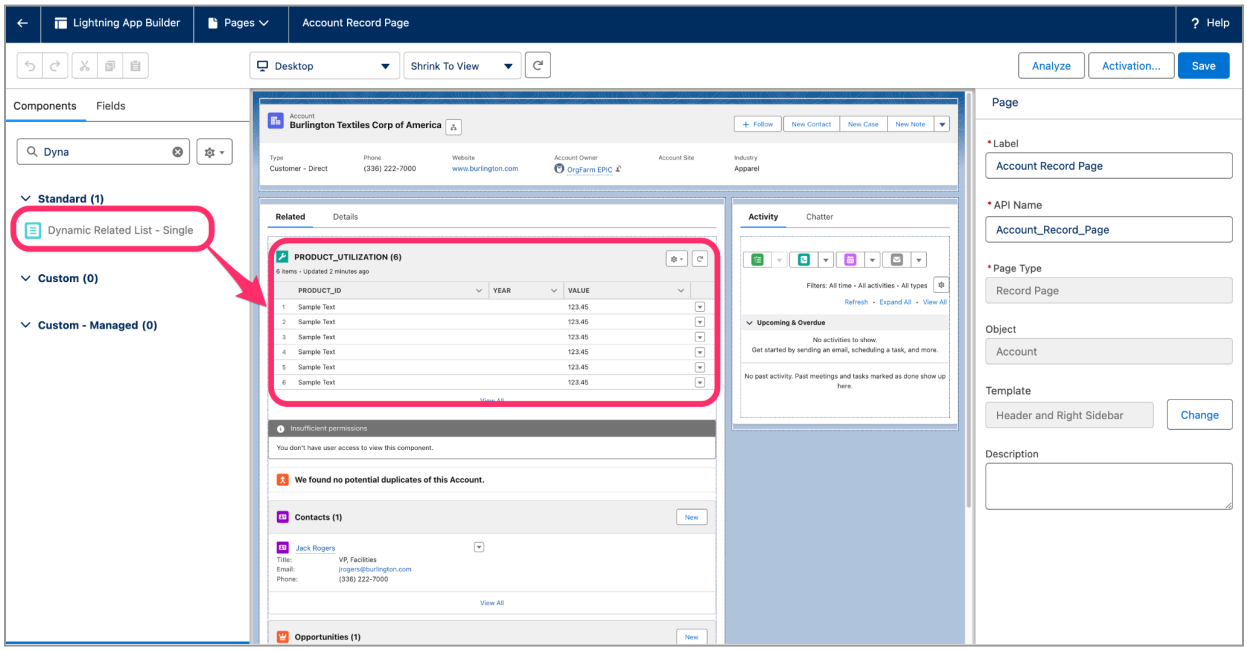
After Identity Resolution, the next step is to enable our initial activation: a Data Cloud related list on the Account page.

In our example we will provide users with immediate access to product utilization, aiding in informed decision-making and understanding customer engagement.

In your Salesforce Organization, open any Account record, and choose **“Edit Page”** from the Setup Menu.



In the Lightning App Builder, drag and drop a **“Dynamic Related List - Single”** component to the top of the components in the **“Related”** tab of the page, as shown below.



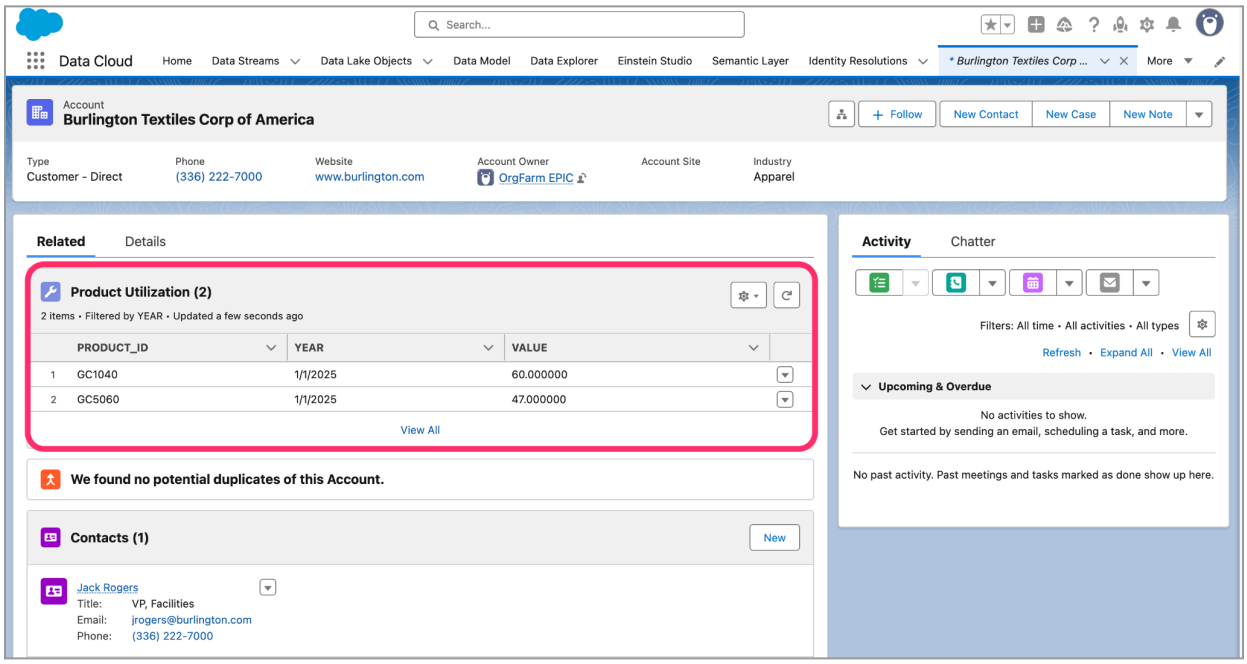
Configure the component with the following parameters:

- Parent Record: Use the default value “Use This Account”.
- Related List: PRODUCT_UTILIZATION
- Related List Label: Product Utilization
- Related List Type: List
- Number of Records to Display: 10
- Related List Fields:

    - PRODUCT_ID
    - YEAR
    - VALUE
- Sort Field: Default
- Related List Filters
    - Field: YEAR
    - Operator: equals
    - Value: THIS YEAR

With this configuration, the component will show the utilization the account has achieved for every product they are using the current year.

Save the page and return to the Account page. The page will show the product utilization records that are actually stored in Snowflake, but they are available in Data Cloud through the zero-copy configuration we completed before.



Respond to changes in the customer behavior

For businesses, maintaining customer satisfaction and minimizing churn are paramount. To achieve this goal, we'll set up an automated system that helps account owners quickly reach out to customers who might be unhappy or aren't using the product much. This system works by automatically creating a high-priority task for the account owner whenever a customer's "Product Utilization" score (which we get from Data Cloud) drops below a

certain level. This way, we can address problems early, improve customer satisfaction, and ultimately keep more customers.

Create a Calculated Insight

We will start by creating a **Calculated Insight**, a custom metric derived from Data Cloud information and computed periodically.

The Calculated Insight, in this example, displays the average utilization score for each account per year.

Step 1: Start a New Calculated Insight

1. From the Data Cloud app, navigate to the **Insights** tab and then select **Calculated Insights**.
2. Click **New**.
3. Choose **Visual Builder** and click **Next**.

Step 2: Add the Source Object

1. Select the **`Product_Utilization`** object. This is your source of raw data.

Step 3: Configure Aggregation

1. This screen is where you define your aggregation, including dimensions and measures.
2. Under **Dimensions**, add your grouping fields. Click the **+** button, select the **`account_number__c`** field, and give it an alias like **`account_number`**. Then, add the **`year__c`** field and give it an alias like **`year`**.
3. Under **Measures**, add your aggregation. Click the **+** button, select the **`value__c`** field, and choose **`Average`** as the function. Give this new measure an alias like **`avg_utilization`**.
4. Click **Next**.

Step 4: Finalize and Deploy

1. Click **Next**.
2. Provide a name for your calculated insight, such as **`Account_Avg_Utilization`**.
3. Set the deployment schedule. The CI will run on this schedule and its updates can now trigger your flow.
4. Click **Save and Deploy**.

Create a Flow to Create a Task in response to low utilization values

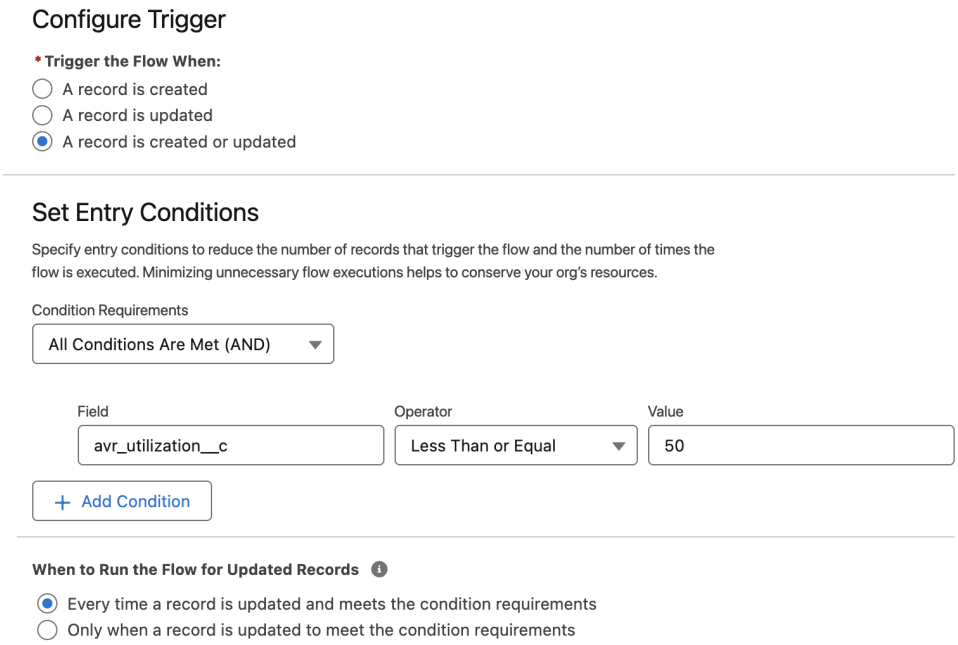
Step 1: Start a New Data Cloud-Triggered Flow

1. From your Salesforce Organization, go to **Setup**.
2. In the Quick Find box, type `Flows` and select it under **Process Automation**.
3. Click the **New Flow** button.
4. Select the **Data Cloud-Triggered Flow** template and click **Create**.

Step 2: Configure the Flow Trigger

This is where you tell the flow what to listen for in Data Cloud.

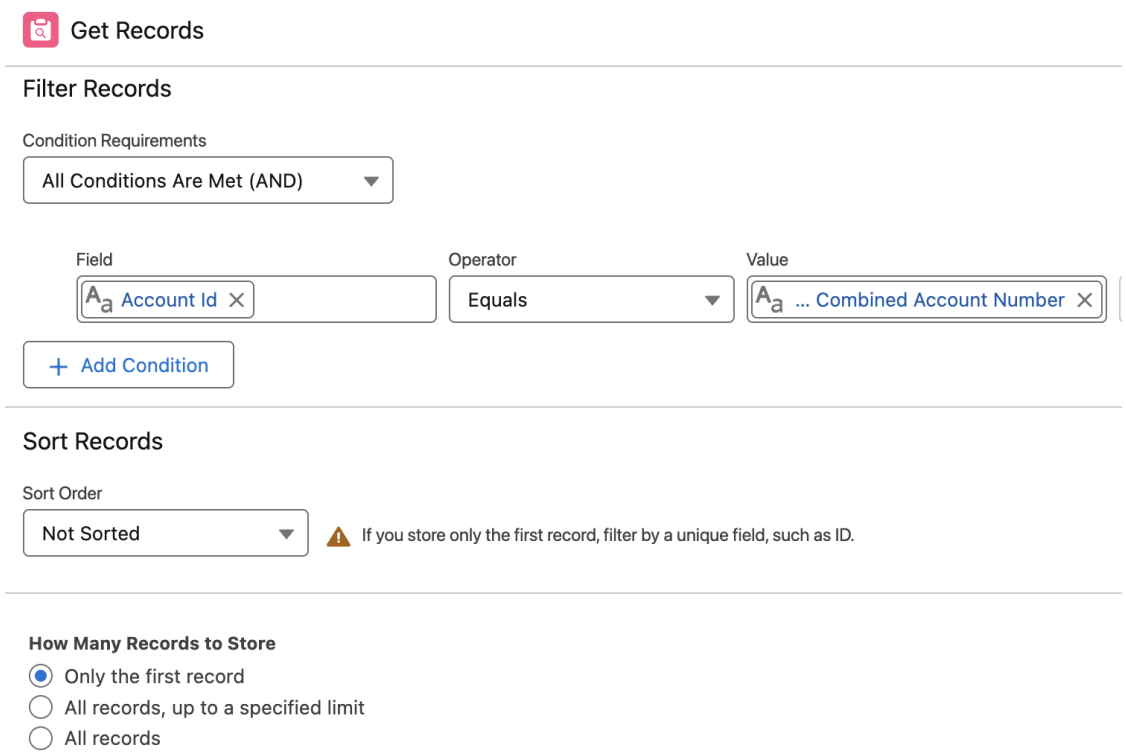
1. **Data Space:** Select the data space that contains your product utilization data. For this example, choose `default`.
2. **Object** Select your calculated insight, in this example Utilization by Product
3. Trigger Configuration:
    - Choose `A record is created or updated`.
    - Set the trigger condition to `All Conditions Are Met (AND)`.
    - Set the Condition:
        - **Field:** Select `avr_utilization__c` (this is the field that stores the utilization score).
        - **Operator:** Select `Less Than or Equal`.
        - **Value:** Enter the utilization threshold, such as `50`.



Step 3: Find the Matching Salesforce Account

Before creating a task, the flow needs to find the corresponding account from the Unified Account object in Datacloud.

1. 1Add a **Get Records** element to the canvas by clicking on the plus button under the Data Cloud Triggered Flow.
2. Give it a label like `Get Unified Link Account`
3. Data Source: Select Data Cloud Object
4. **Object:** Select `Unified Link Account sseg`.
5. **Filter:** Set the condition to find the matching account.
    - **Field:** `Account.AccountNumber`
    - **Operator:** `Equals`
    - **Value:** `{!$Record.combined_account_number__c}` (this is the account number from your Product Utilization DMO).
6. **Sort and Store:** Select `Only the first record` and `Automatically store all fields`.

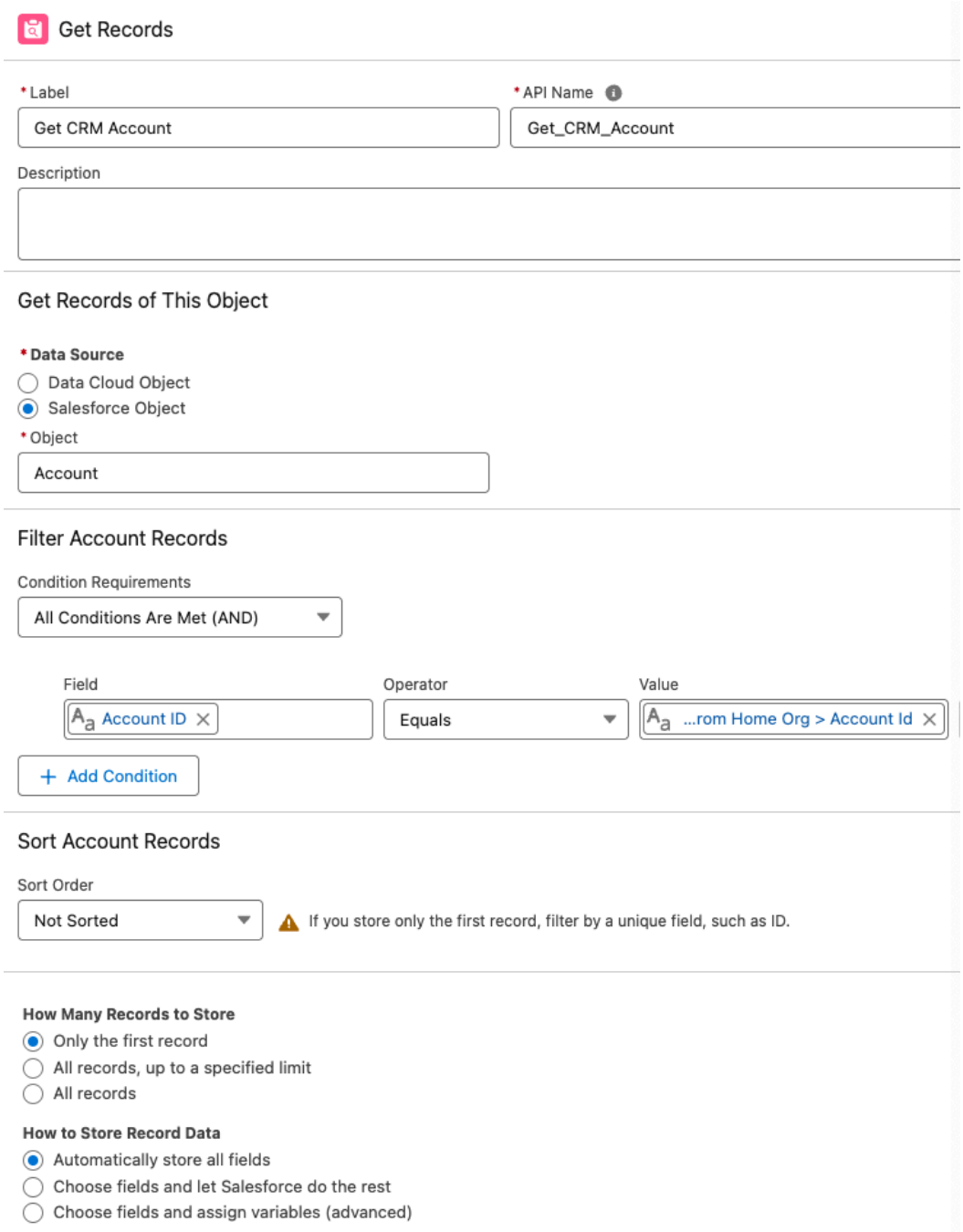


Step 4: Create a Get Record Element to retrieve the Account Record in CRM

The next part is to create a get record task, to find corresponding account that is stored in the Salesforce CRM Account object

1. Add a **Get Records** element to the canvas by clicking on the plus button under the Get Unified Link Acocunt Triggered Flow.
2. Give it a label like `Get CRM Account`
3. Data Source: Select Salesforce Object
4. **Object:** Select `Account`
5. **Filter:** Set the condition to find the matching account.
    - **Field:** `Account.ID`
    - **Operator:** `Equals`
    - **Value:** `{!Get_Unified_Link_Account_from_Home_Org.SourceRecordId__c}` (this is the account number from the previous get Record action).

6. **Sort and Store:** Select `Only the first record` and `Automatically store all fields`.

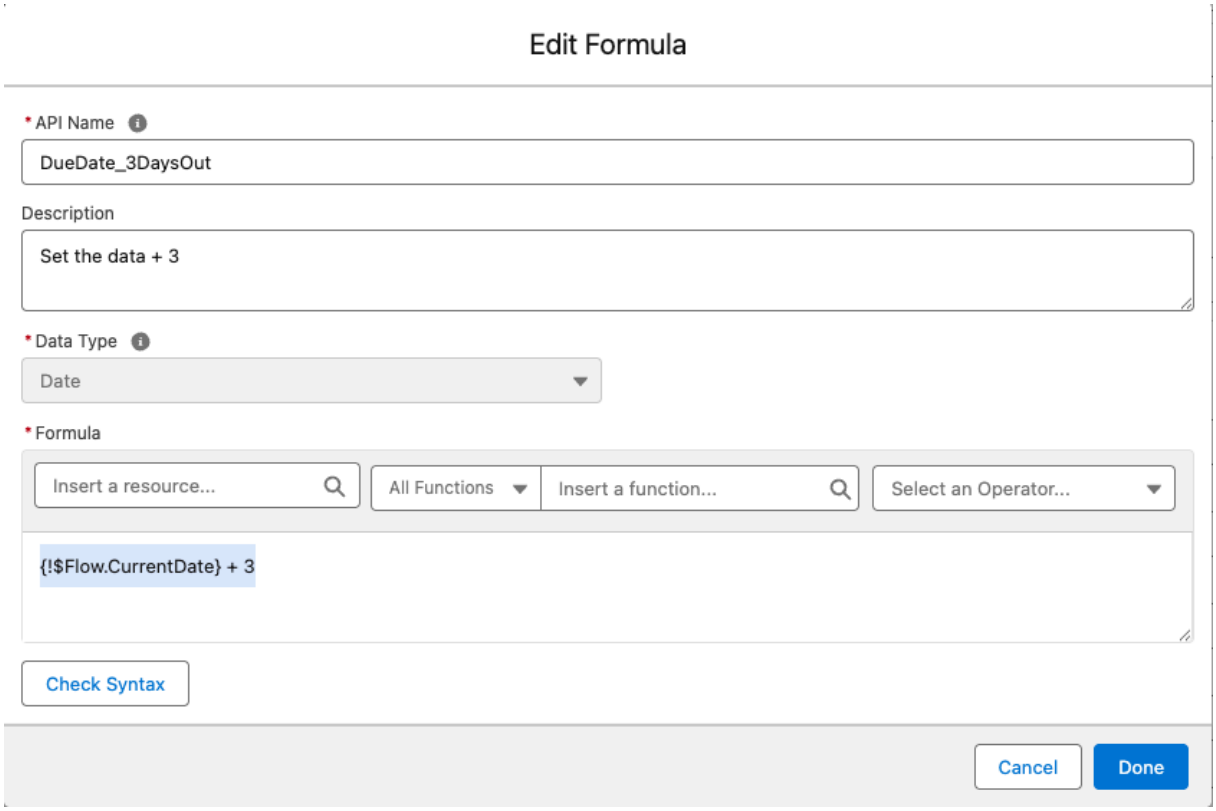


Step 5: Create a Formula Resource for the Due Date

To set a due date for the task, you'll need a simple formula.

1. In the Flow Builder, go to the **Manager** tab on the left and click **New Resource**.
2. **Resource Type:** Select `Formula`.

3. **API Name:** Name it `DueDate_3DaysOut`.
4. **Data Type:** Select `Date`.
5. **Formula:** Enter `{!$Flow.CurrentDate} + 3`. This will set the task due date three days from when the flow runs.

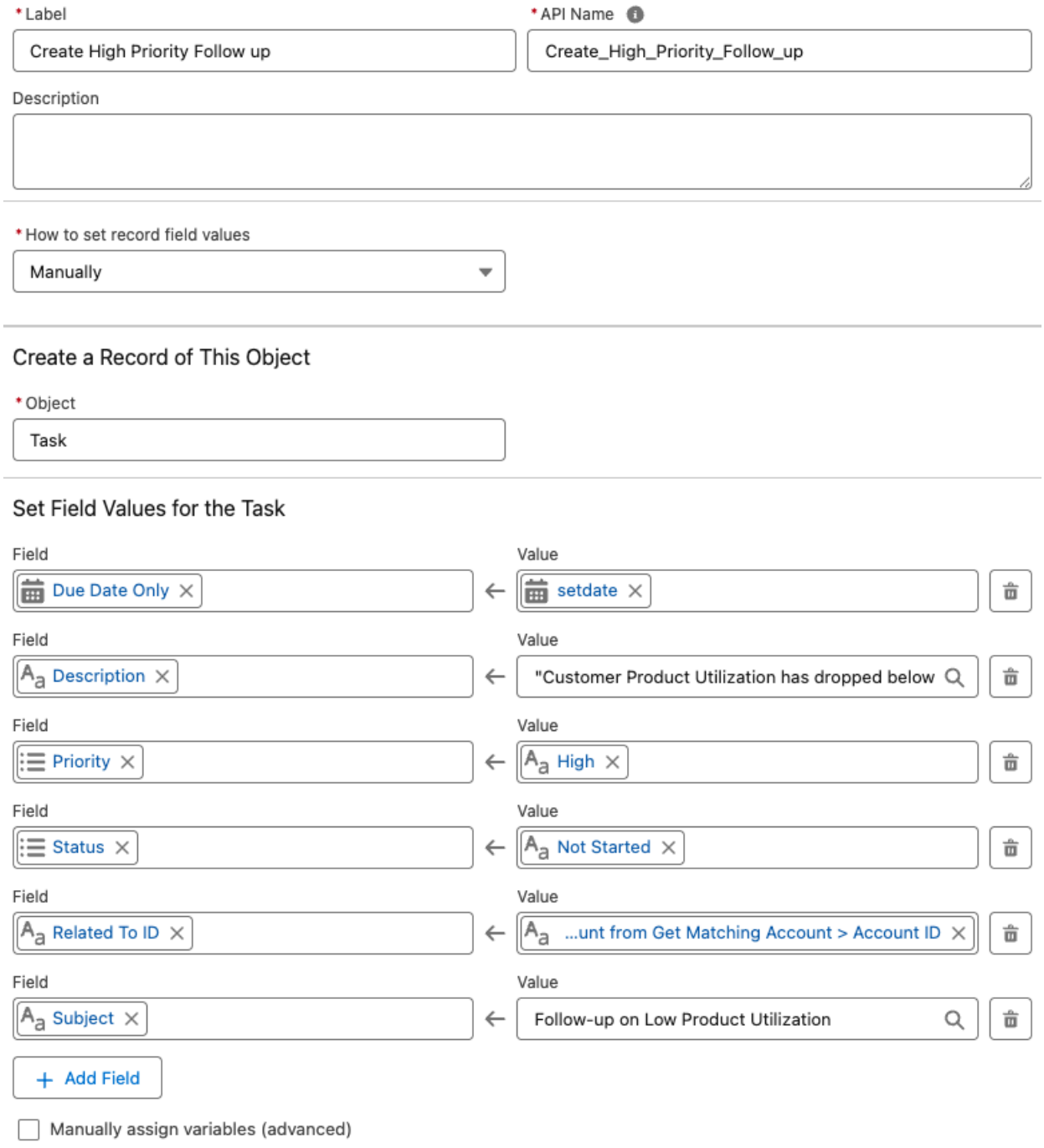


Step 6: Create the Task Record

Now the flow will create the task and assign it to the right person.

1. Add a **Create Records** element to the canvas.
2. Give it a label like `Create High-Priority Follow-up`.
3. **How to set record field values:** Select `Manually`.
4. **Object:** Select `Task`.
5. **Map the Fields:** Set the values for the new task.
    - **Subject:** "Follow-up on Low Product Utilization"
    - **Description:** "Customer Product Utilization has dropped below the threshold. Please follow up ASAP."
    - **ActivityDate (Due Date):** `DueDate_3DaysOut`
    - **Priority:** "High"
    - **Status:** "Not Started"

- **WhatId (Related To ID):** `Get_Matching_Account.Id` (This links the task to the correct account).



Step 7: Save and Activate

1. Click **Save**.

2. Name your flow something clear and descriptive, like `Create Task for Low Product Utilization`.
3. Click **Activate**.

When activated, the flow will automatically trigger a task for users when the calculated insight for product utilization falls below 50%, prompting them to take action.