

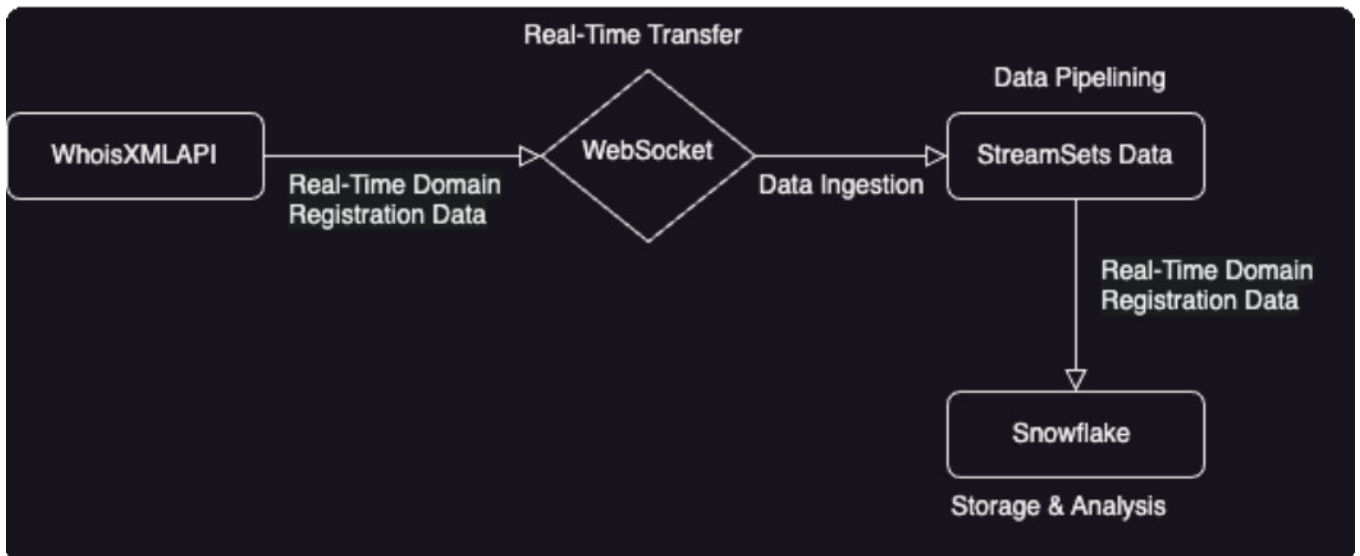
# StreamSetsをSnowflakeに統合する

このガイドでは、StreamSetsをSnowflakeに統合する手順を説明します。

## 前提条件

- StreamSetsのアカウント
- DockerのData Collector（前のガイドに記載の通り）
- Snowflakeのアカウント
- WHOISXMLAPI.COMのReal-Time Domain Registrationのサブスクリプション

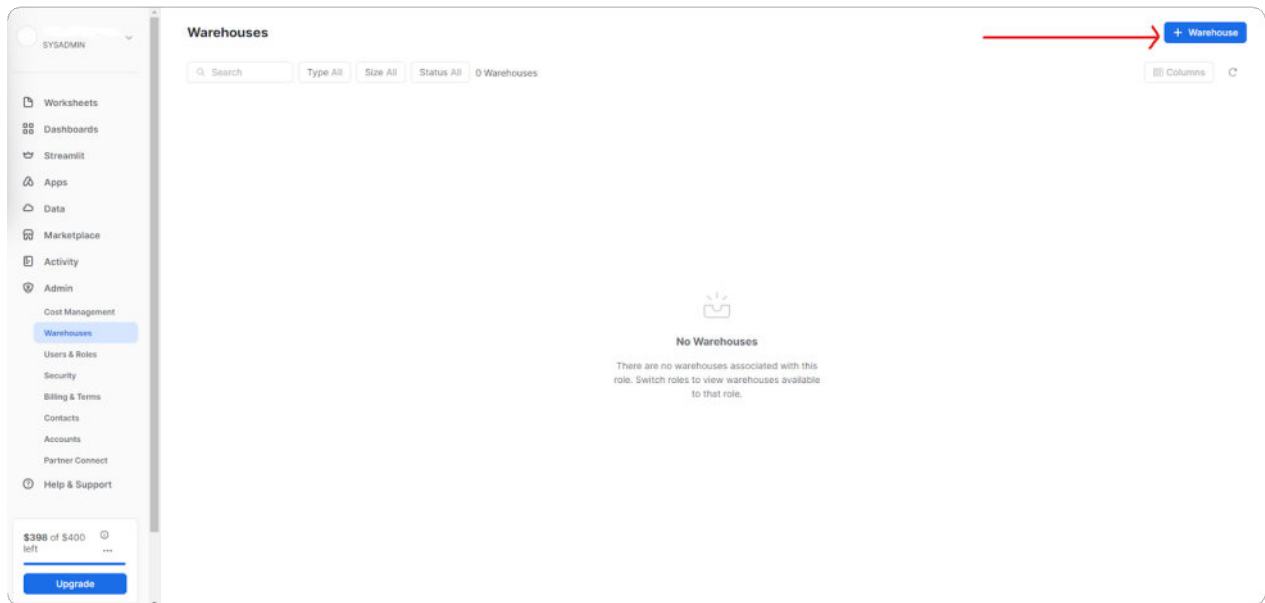
## 設計の概要



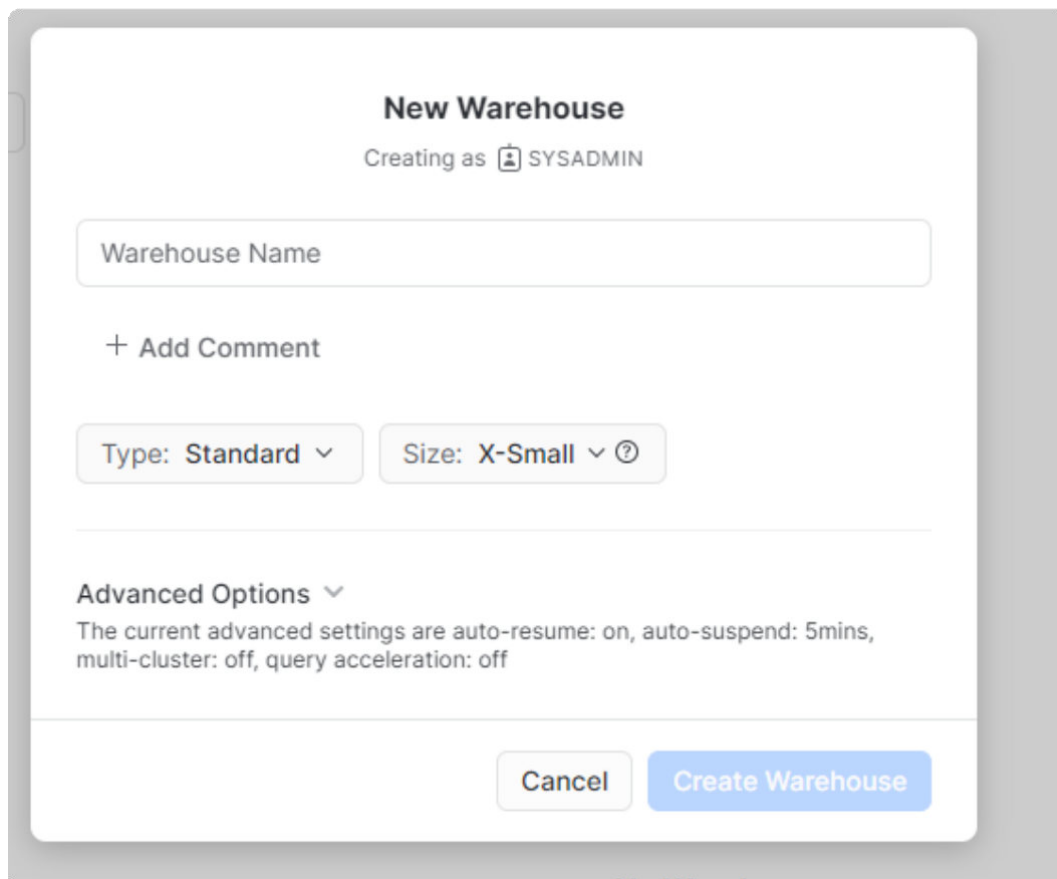
## Snowflakeでデータベースを作成する

### ステップ1: ウェアハウスをセットアップする

- ログイン後、サイドバーの「Admin」から「Warehouses」を選択し、以下の画像のように「Warehouse」ボタンをクリックしてください。

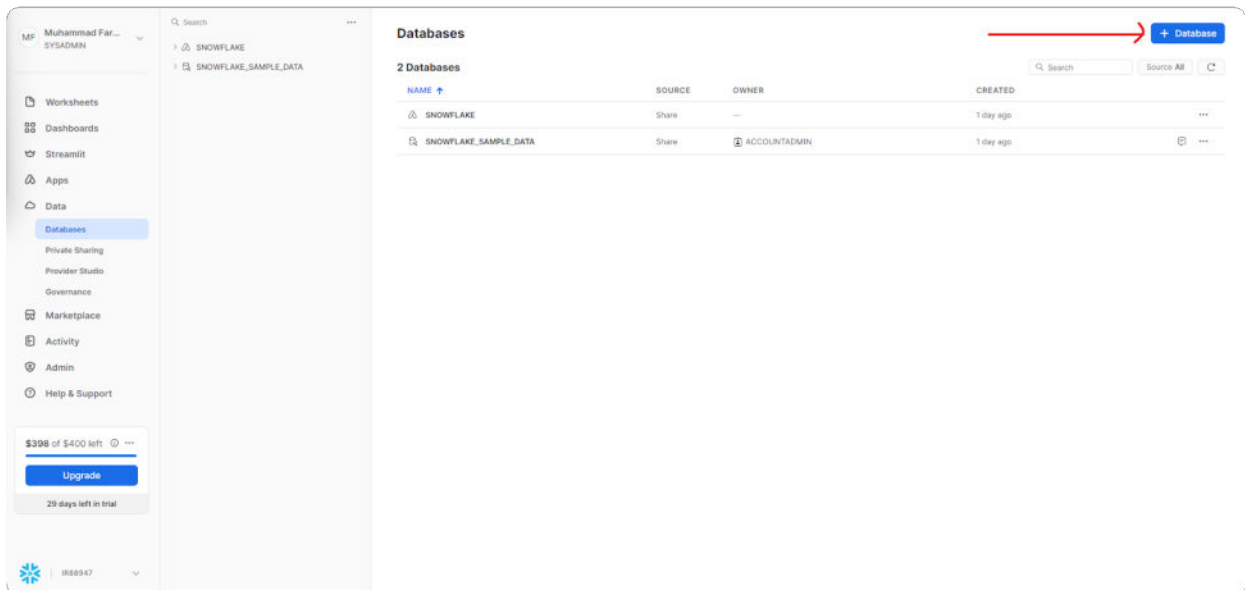


- 適切なウェアハウス名を選択し、「Create Warehouse」をクリックしてください。

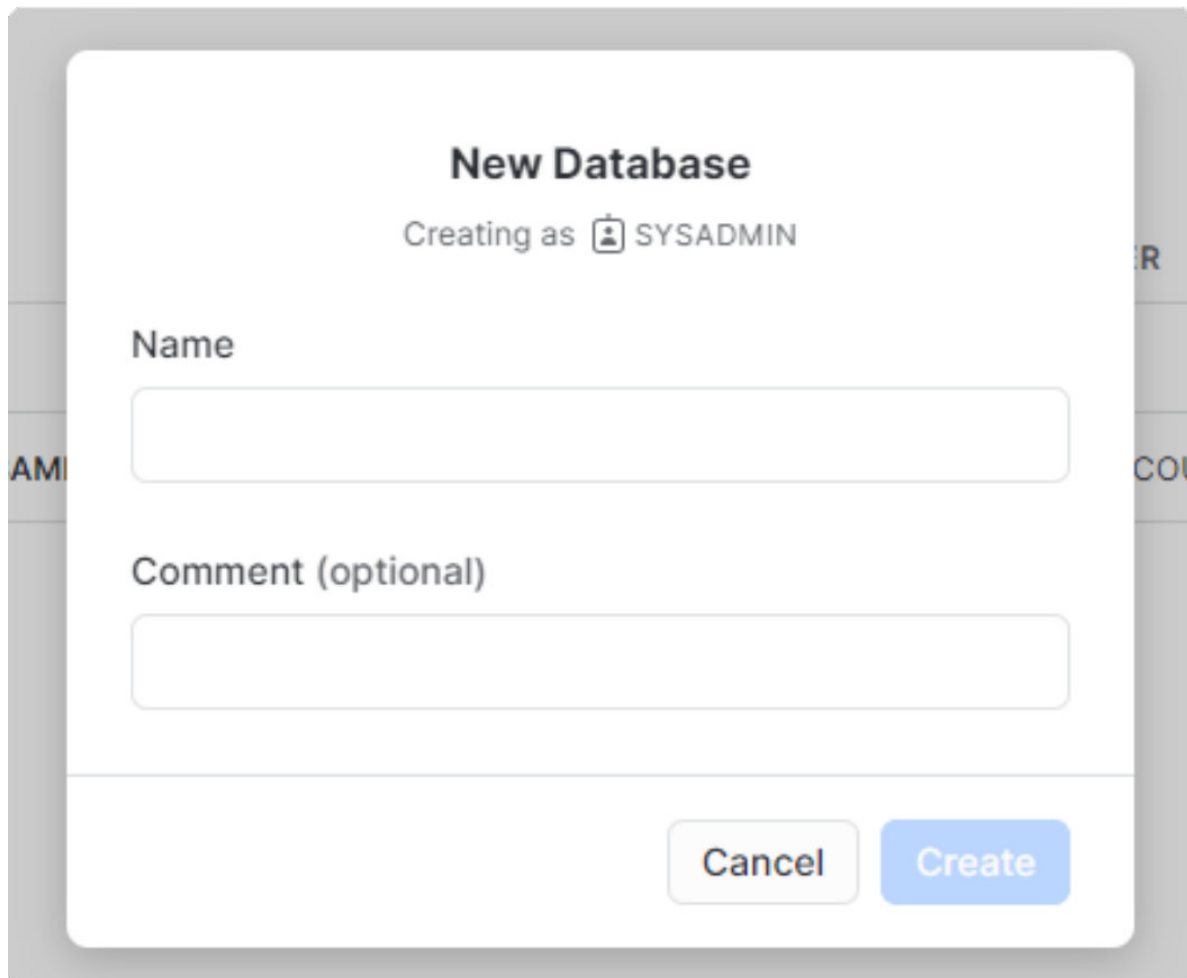



## ステップ2：データベースをセットアップする

- サイドバーの「Data」を選択し、「Databases」を選んでクリックします。



- 適切なデータベース名を選び、「Create」をクリックします。



**New Database**  
Creating as  SYSADMIN

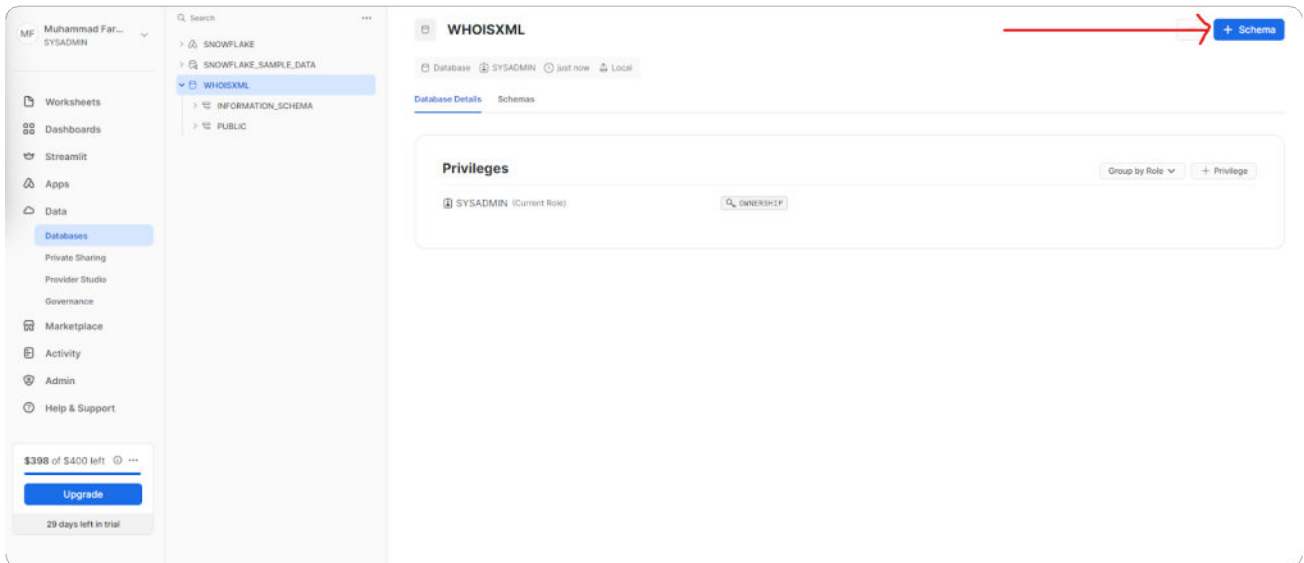
Name

Comment (optional)

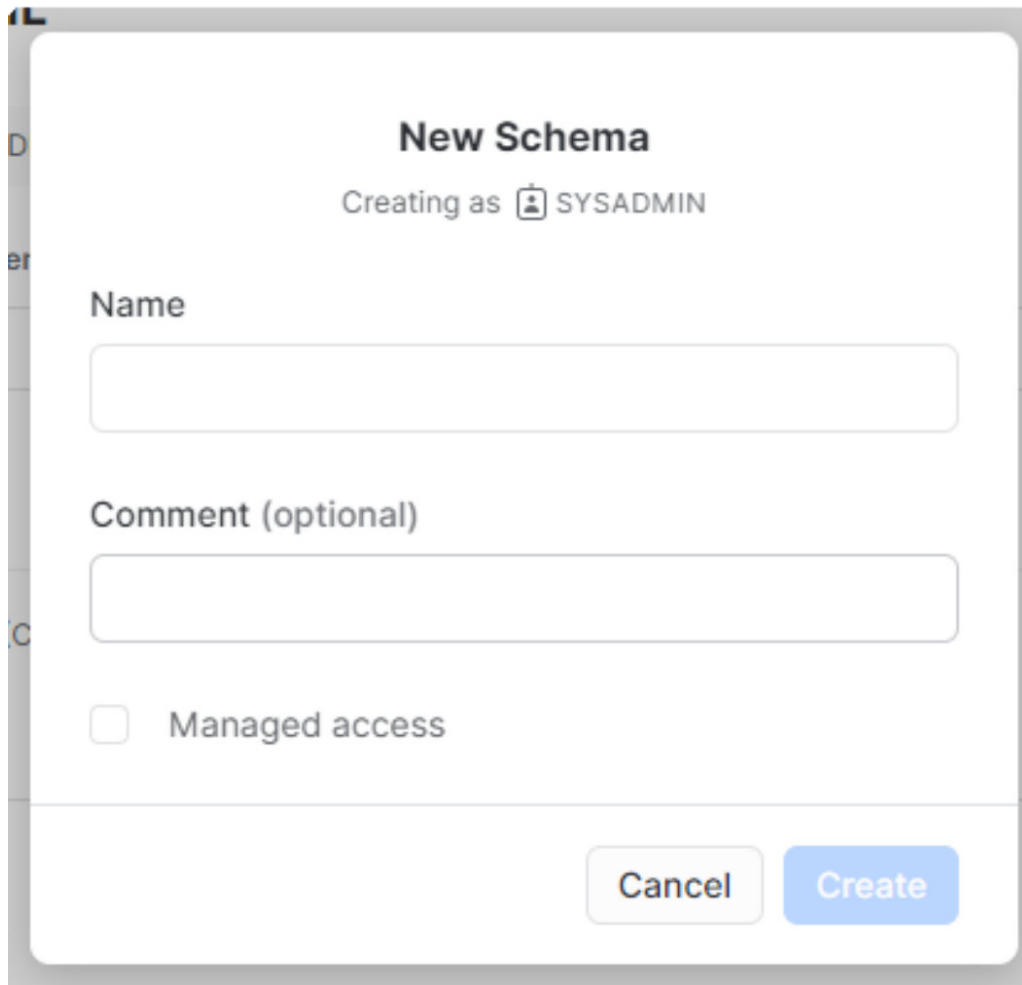
Cancel Create


### ステップ3：スキーマを作成する

- データベースを選択し、「+Schema」をクリックします。



- 適切なスキーマ名を選択し、「Create」をクリックします。



**New Schema**  
Creating as  SYSADMIN

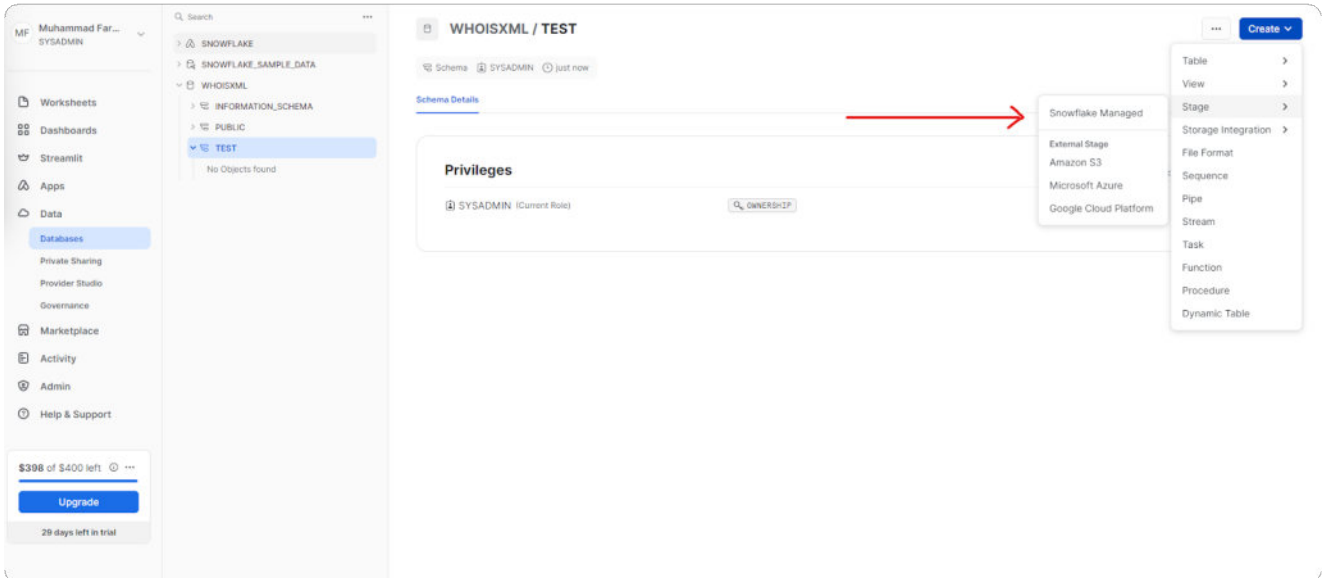
Name

Comment (optional)

Managed access

## ステップ4：ステージを作成する

- 前のステップで作成されたスキーマを選んで「Create」をクリックし、「Stage」を選択します。そして、「Snowflake Managed」をクリックします。



- 適切なステージ名を選び、「Create」をクリックします。



### Create Stage

Creating as SYSADMIN

Snowflake

Stage Name

Comment (optional)

---

**Directory table**

Enabling directory table is required to see the files stored on the stage. Warehouse is required. [Learn more](#)

---

**Encryption**

Select the type of encryption supported for all files stored on the stage. You won't be able to change it later.

Client-side encryption

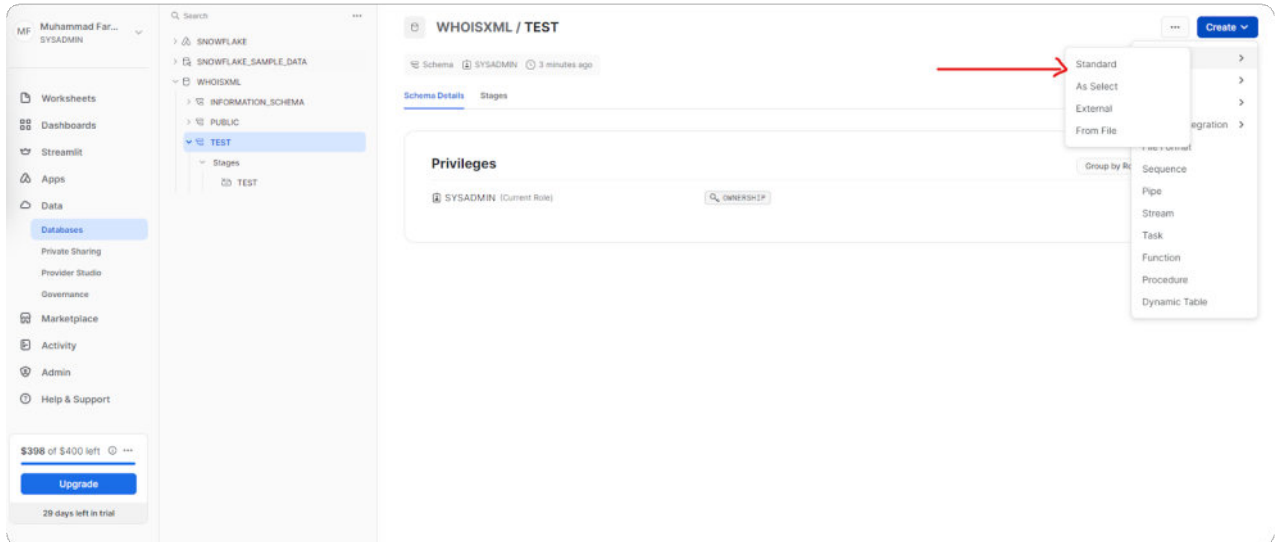
Server-side encryption ⓘ

---

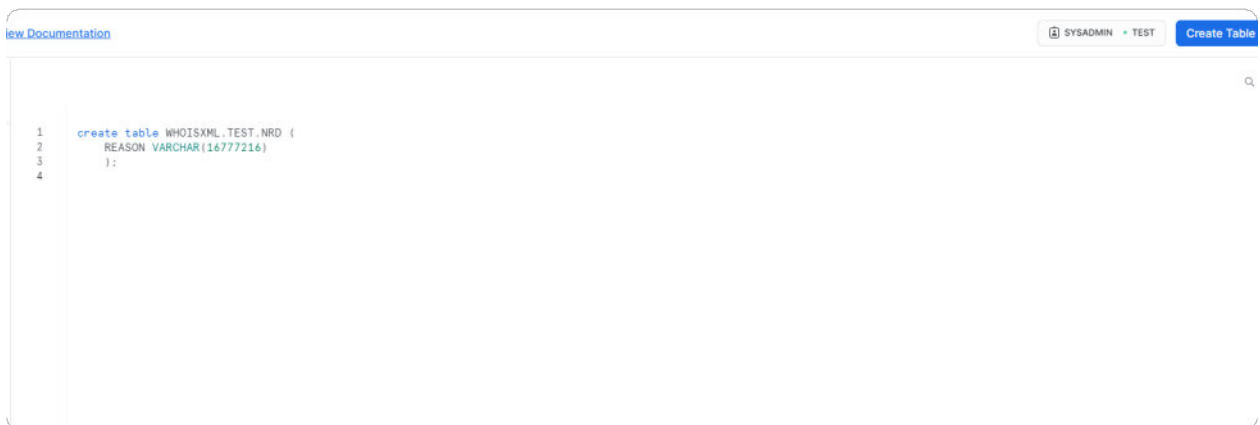
SQL Preview

## ステップ5：テーブルを作成する

- 上の手順で作成されたスキーマを選んで「Create」ボタンをクリックし、「Table」を選択します。そして、「Standard」をクリックします。



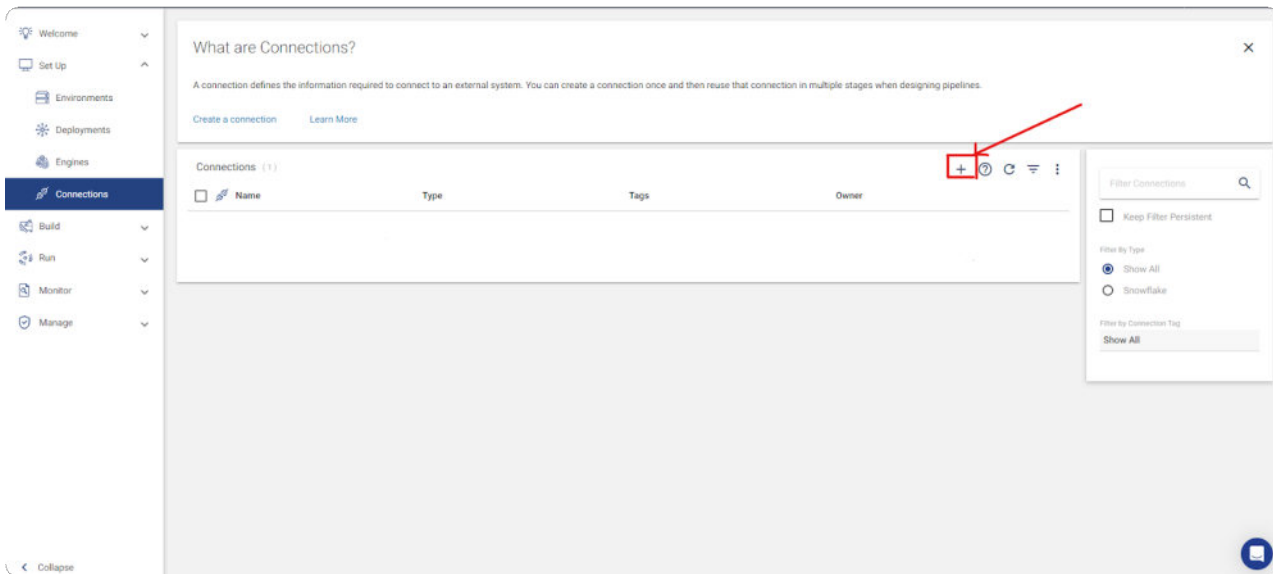
- 下図のようにSQLクエリを記述してテーブルを作成し、「Create Table」をクリックします。



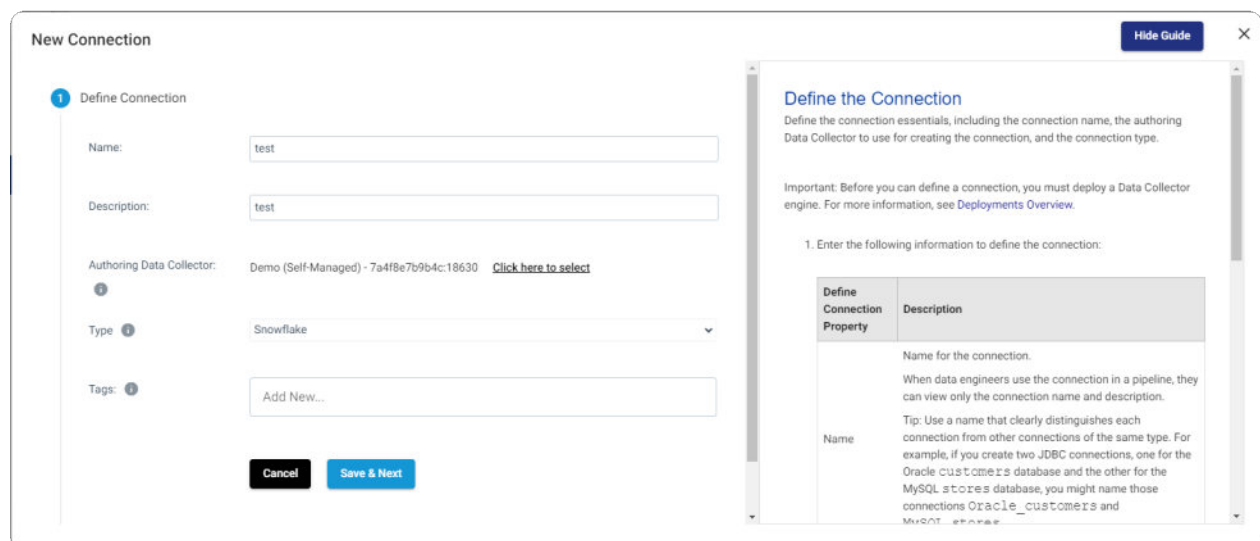
## Snowflakeのコネクションを作成する

### ステップ6：コネクションをセットアップする

- StreamSets UIでサイドバーから「Setup」、「Connections」を選択します。下図のように、「+」ボタンをクリックしてコネクションを作成します。



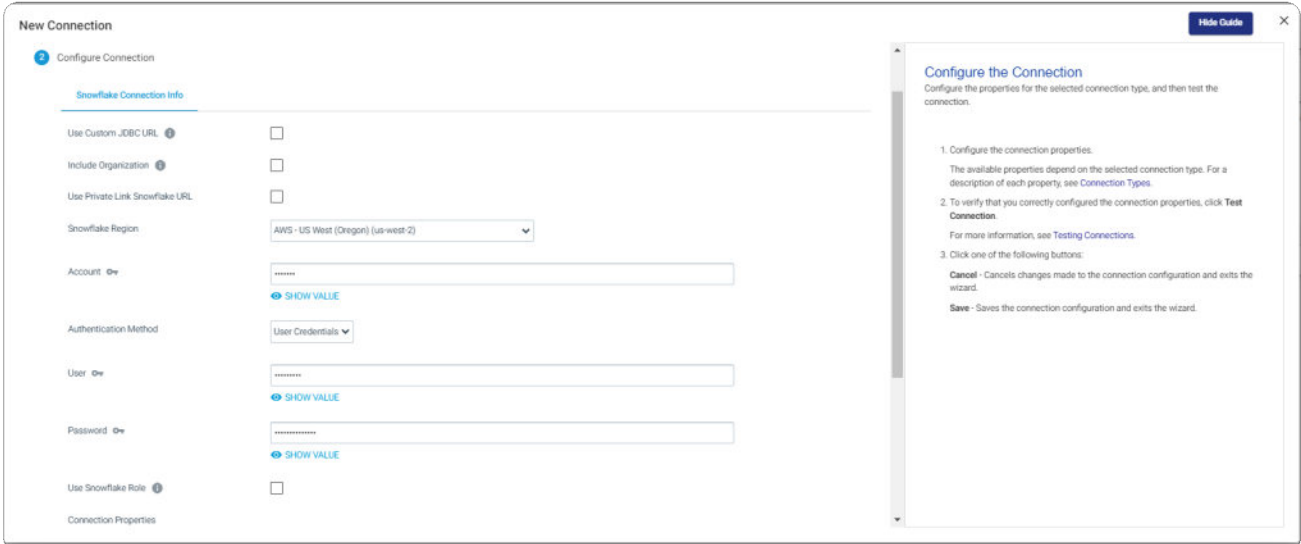
- 新しいコネクションを仕様に従ってカスタマイズし、「Save & Next」をクリックして進みます。データコレクタとタイプを選択してください。



Define Connection Property	Description
Name	Name for the connection. When data engineers use the connection in a pipeline, they can view only the connection name and description. Tip: Use a name that clearly distinguishes each connection from other connections of the same type. For example, if you create two JDBC connections, one for the Oracle customers database and the other for the MySQL stores database, you might name those connections Oracle_customers and MySQL_stores.



- あなたの認証情報に従って新しいコネクションを設定し、「Save」をクリックして進みます。必ずコネクションのテストを行なってください。



**New Connection** Hide Guide ×

1 Configure Connection

**Snowflake Connection Info**

Use Custom JDBC URL

Include Organization

Use Private Link Snowflake URL

Snowflake Region

ACCOUNT  [SHOW VALUE](#)

Authentication Method

User  [SHOW VALUE](#)

Password  [SHOW VALUE](#)

Use Snowflake Role

Connection Properties

**Configure the Connection**

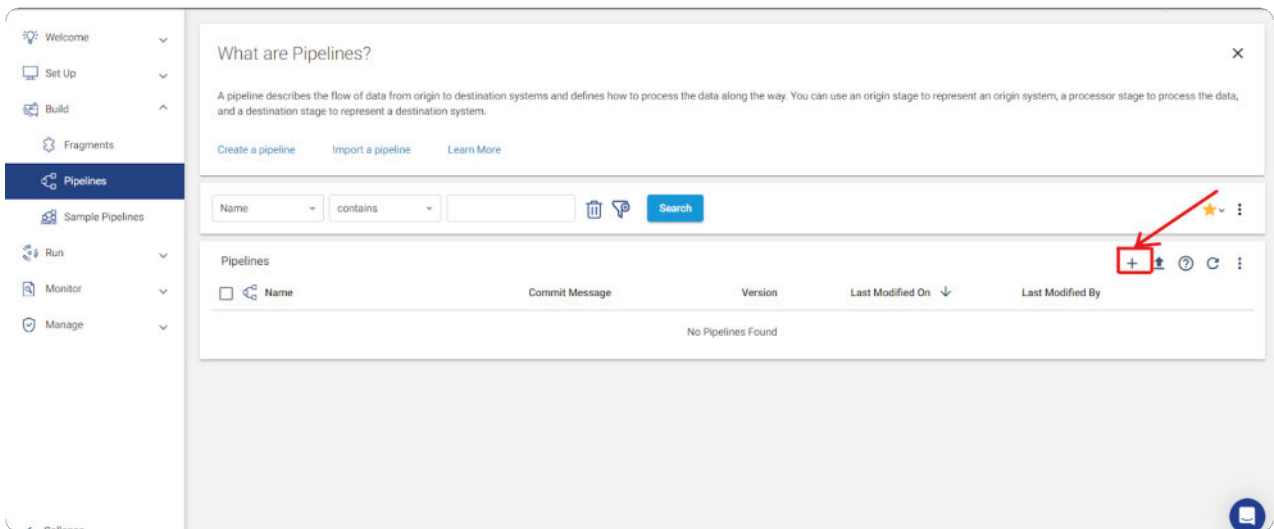
Configure the properties for the selected connection type, and then test the connection.

1. Configure the connection properties.  
The available properties depend on the selected connection type. For a description of each property, see [Connection Types](#).
2. To verify that you correctly configured the connection properties, click **Test Connection**.  
For more information, see [Testing Connections](#).
3. Click one of the following buttons:  
**Cancel** - Cancels changes made to the connection configuration and exits the wizard.  
**Save** - Saves the connection configuration and exits the wizard.

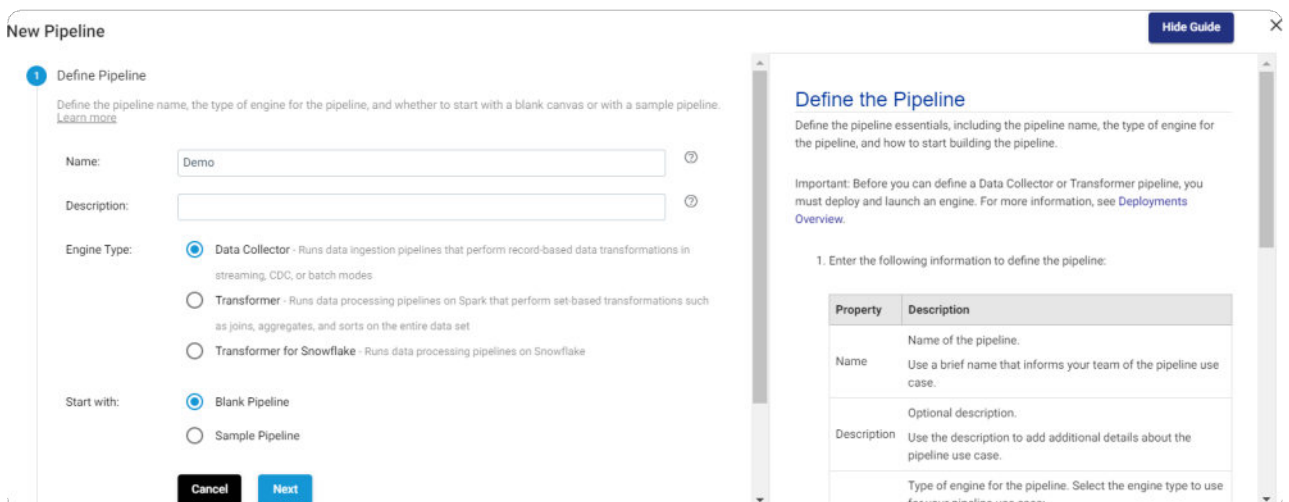
## パイプラインを作成する

### ステップ7：パイプラインをセットアップする

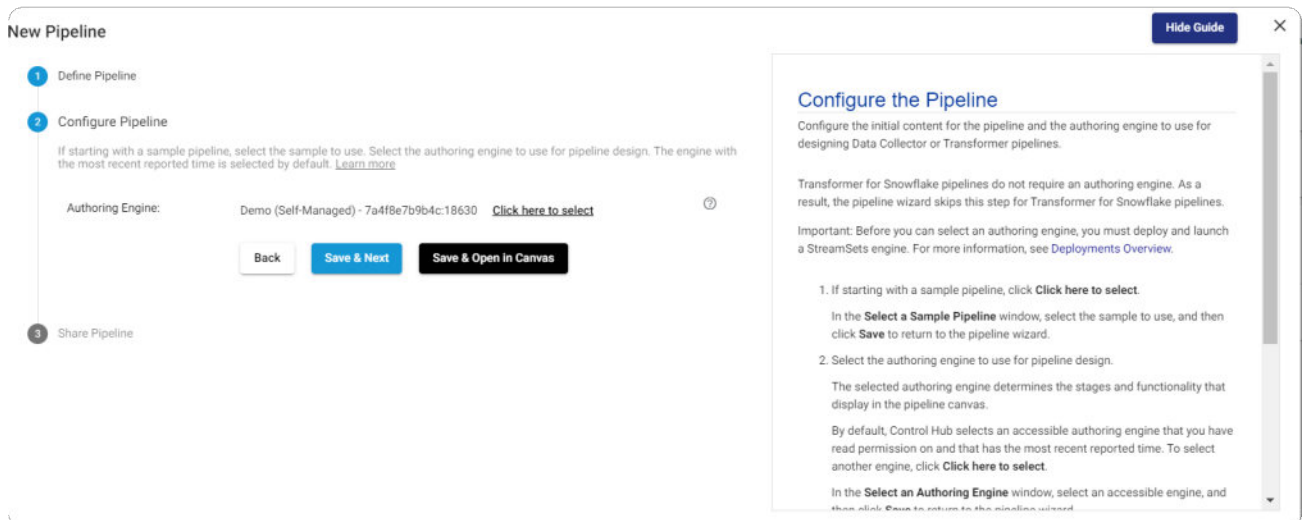
- StreamSets UIでサイドバーから「Build」、「Pipelines」を選択します。下図のように、「+」をクリックしてパイプラインを開始します。



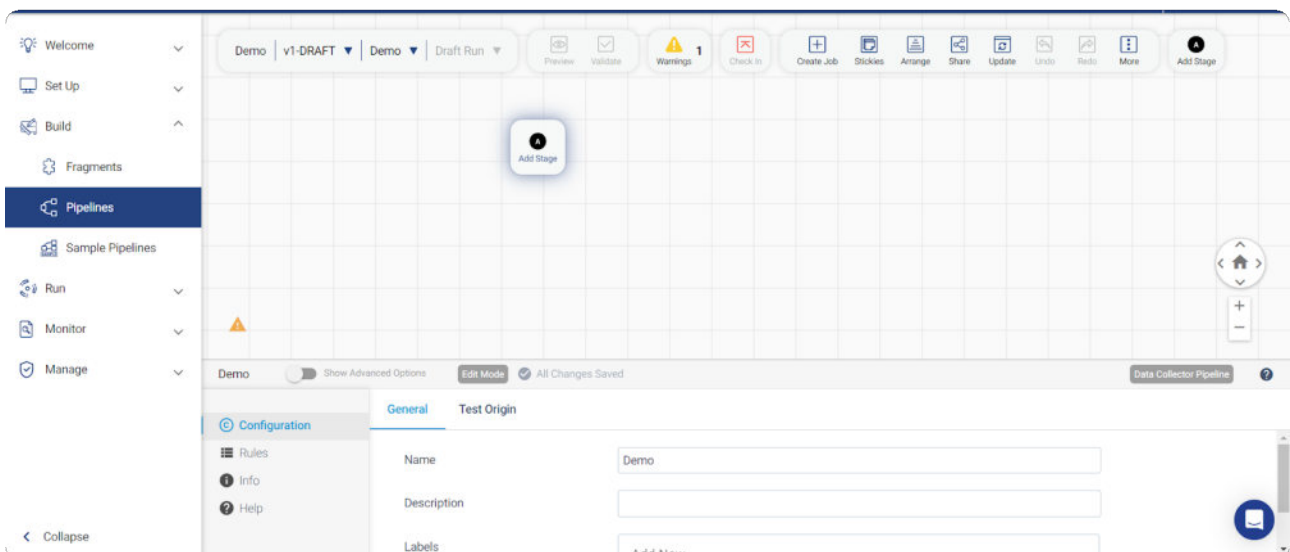
- 新しいパイプラインを仕様に合わせてカスタマイズし、「Next」をクリックして進みます。



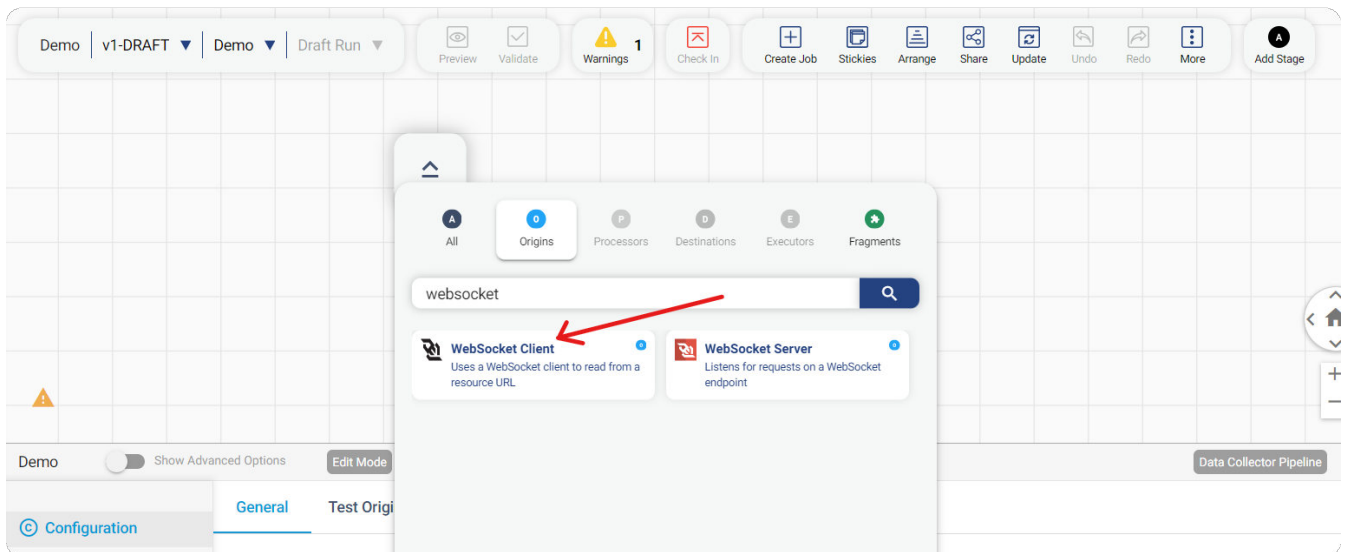
- パイプラインの設定を調整して指定のデータコレクタを選択し、「Save & Open in Canvas」をクリックします。



- 以下のようなユーザーインターフェース (UI) が表示されます。



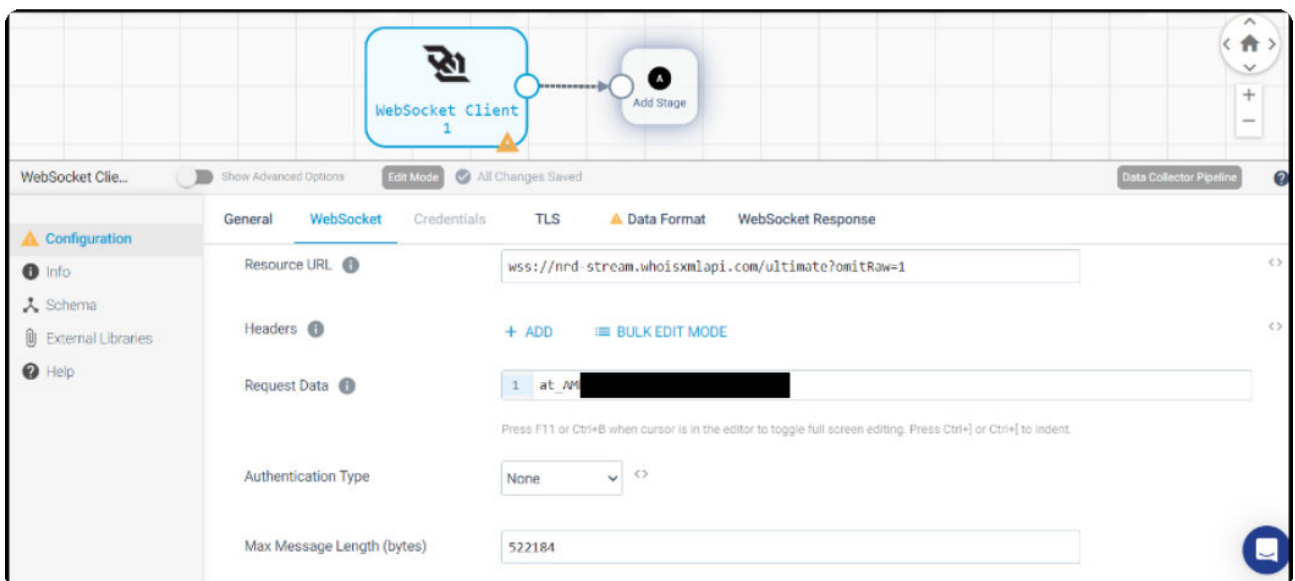
- 「Add Stage」をクリックし、次に「WebSocket」を検索します。そして、「WebSocket Client」を選択します。



WebSocketステージを選択し、必要に応じて設定します。

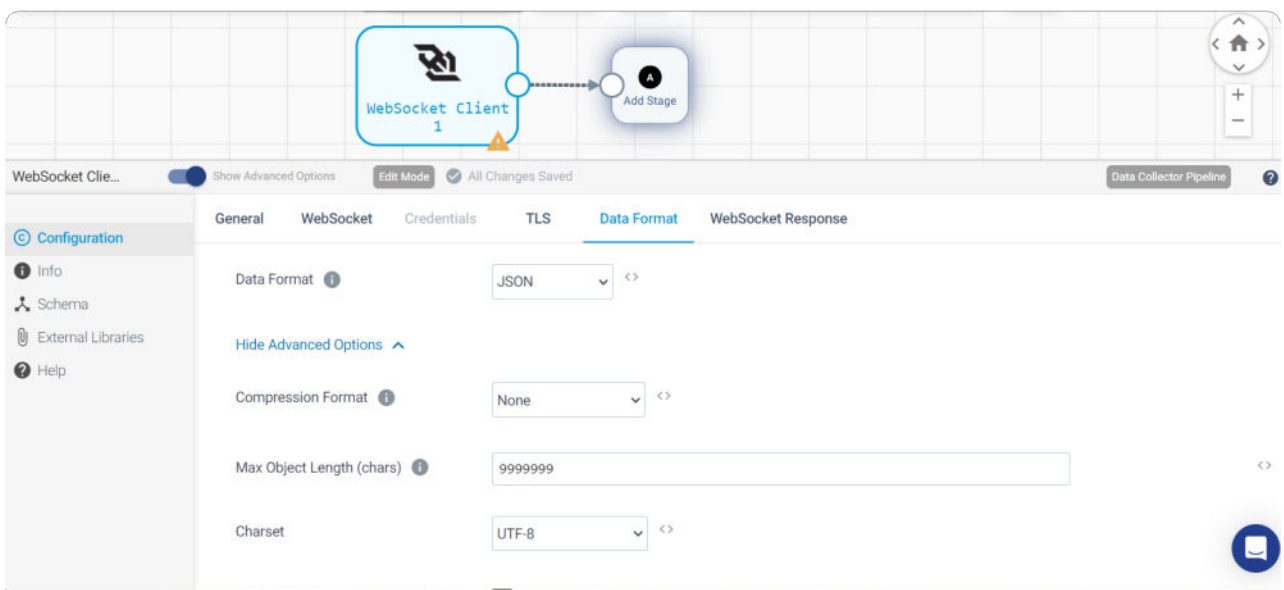
## WebSocketを設定する

- Resource URLを追加します。
- Request Dataを追加します（APIキーが入ります）。
- Max Message Length (bytes)に入力します（最小値：522184）。

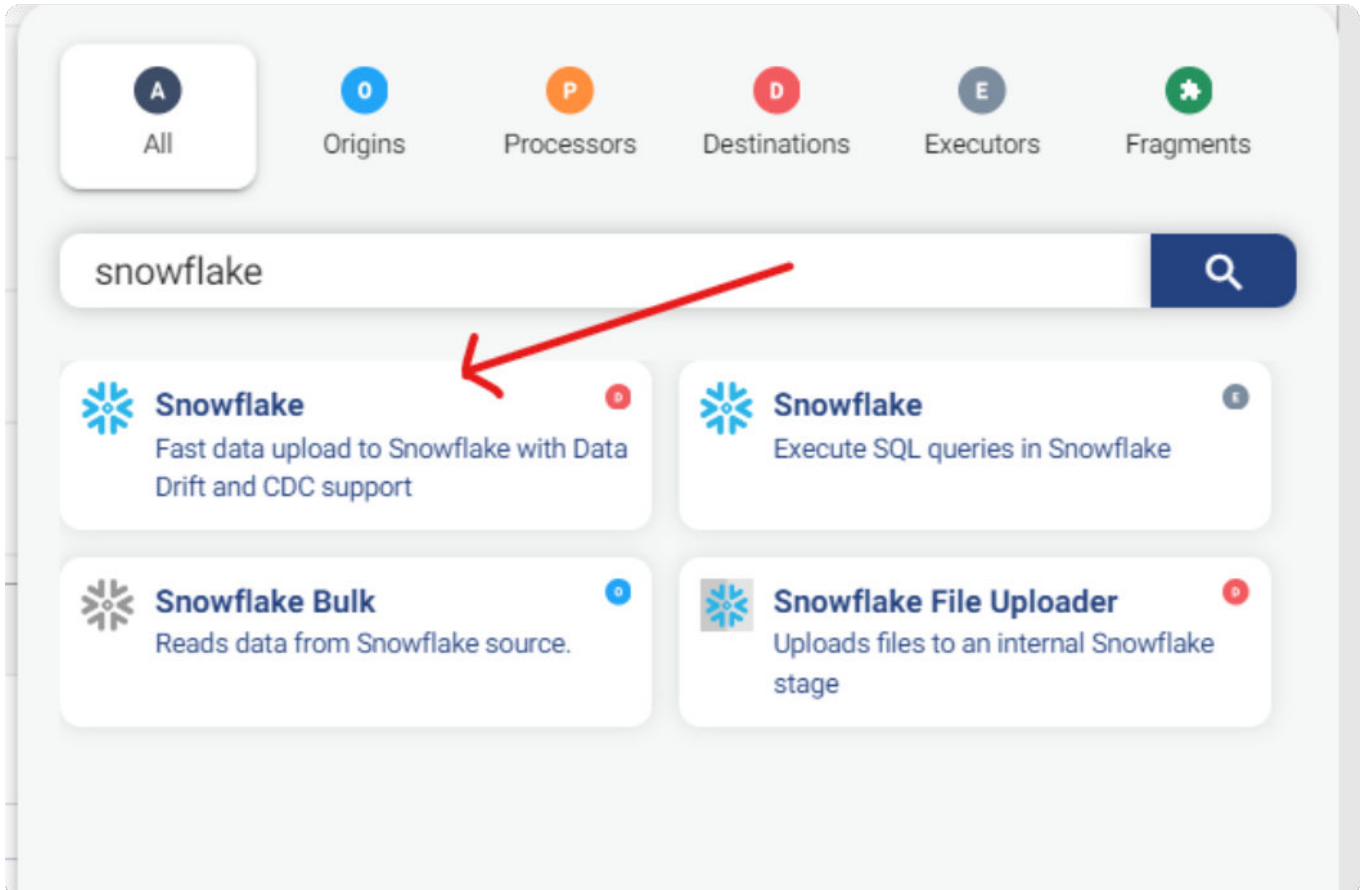


### データフォーマットの設定

- Data FormatでJSONを選択します。
- Max Object Length (chars)に9999999を入力します（必要に応じて変更できます）。

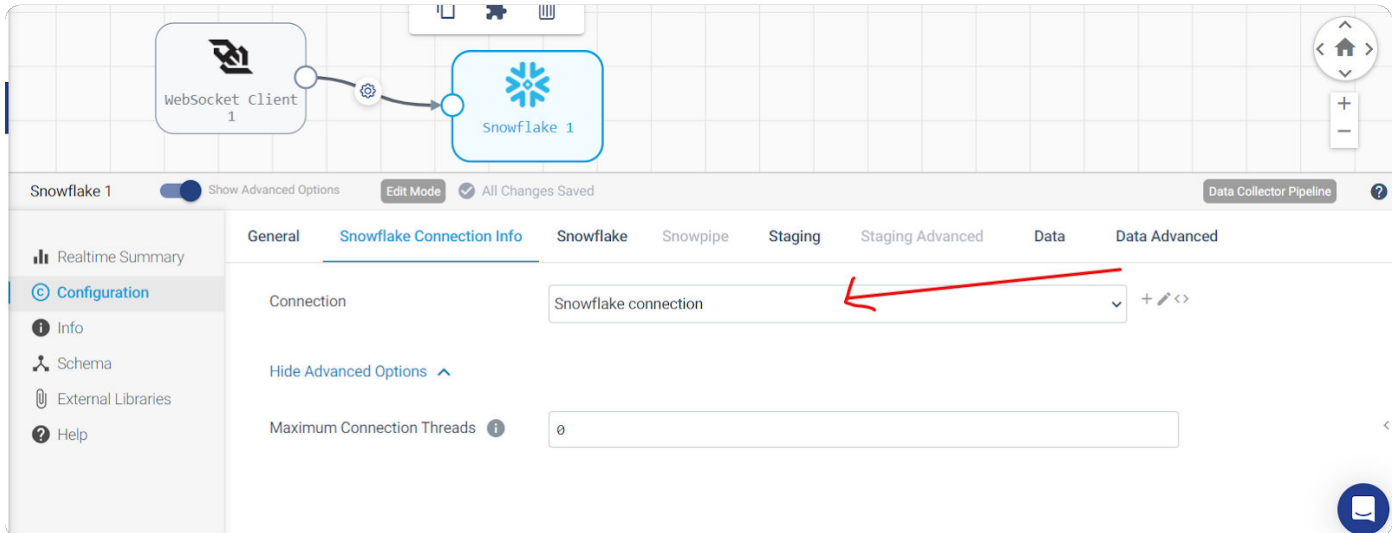


- UIで「Add Stage」を選んでステージを追加し、「Snowflake (destinations)」ステージを選択します。

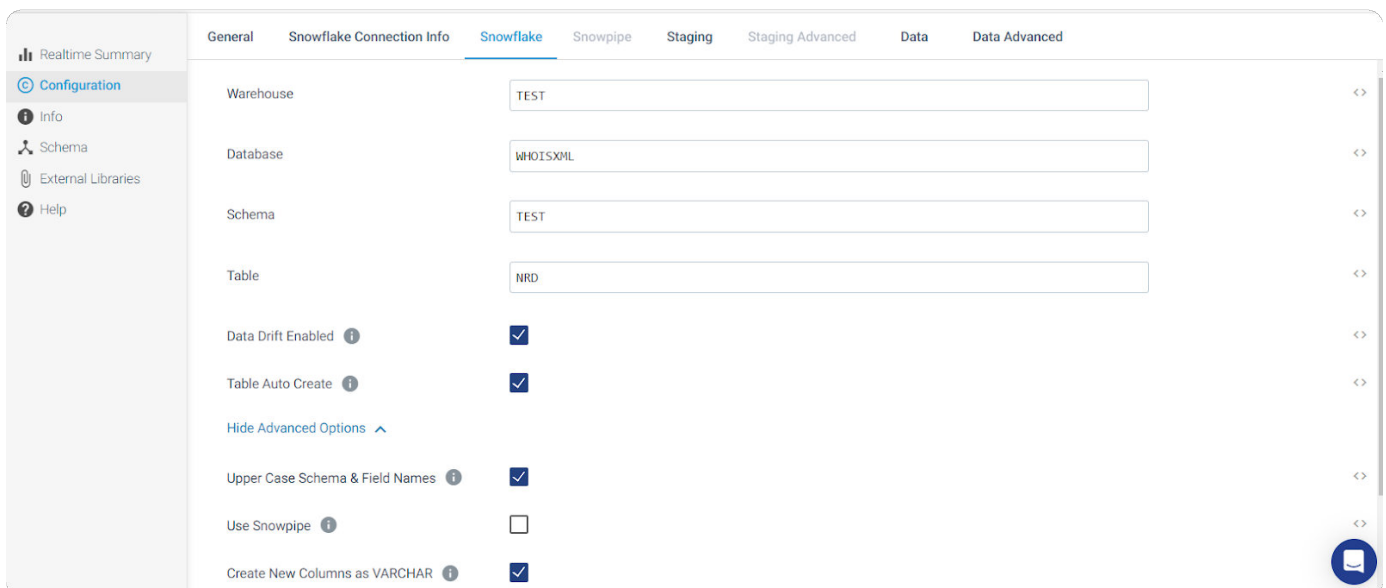


## Snowflakeを設定する

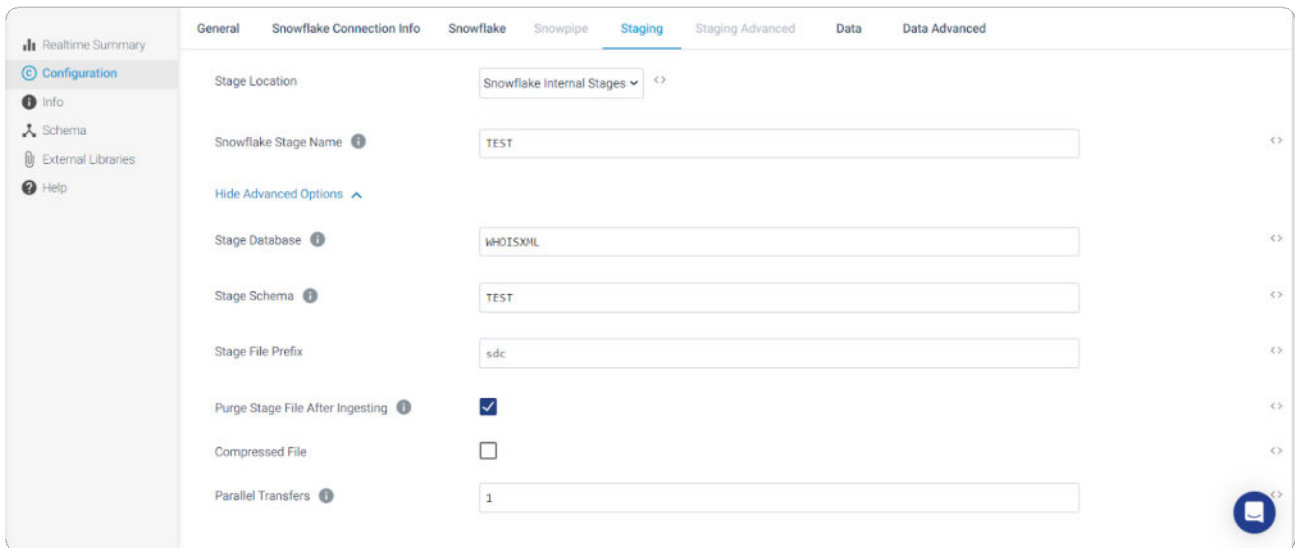
- 前のステップで作成したコネクションを選択します。



- SnowflakeのConfigurationでWarehouse、Database、Schema、Tableを入力します。  
「Create New Columns」にチェックを入れると、自動的にテーブルにカラムが作成されます。



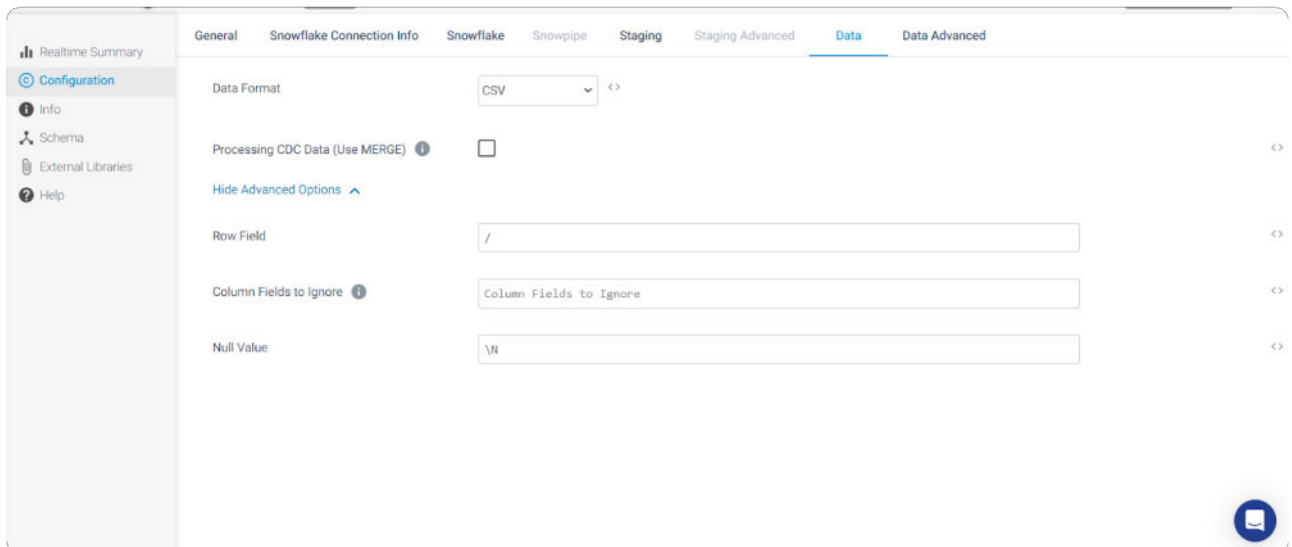
- StagingのConfigurationでStage Name、Stage Database、Stage Schemaに記入します。



The screenshot shows the 'Staging' configuration page. The left sidebar contains 'Configuration', 'Info', 'Schema', 'External Libraries', and 'Help'. The main content area has tabs for 'General', 'Snowflake Connection Info', 'Snowflake', 'Snowpipe', 'Staging', 'Staging Advanced', 'Data', and 'Data Advanced'. The 'Staging' tab is active. The configuration fields are:

Stage Location	Snowflake Internal Stages
Snowflake Stage Name	TEST
Hide Advanced Options	^
Stage Database	WHOISXML
Stage Schema	TEST
Stage File Prefix	sdc
Purge Stage File After Ingesting	<input checked="" type="checkbox"/>
Compressed File	<input type="checkbox"/>
Parallel Transfers	1

- DataのConfigurationに設定を記入します。



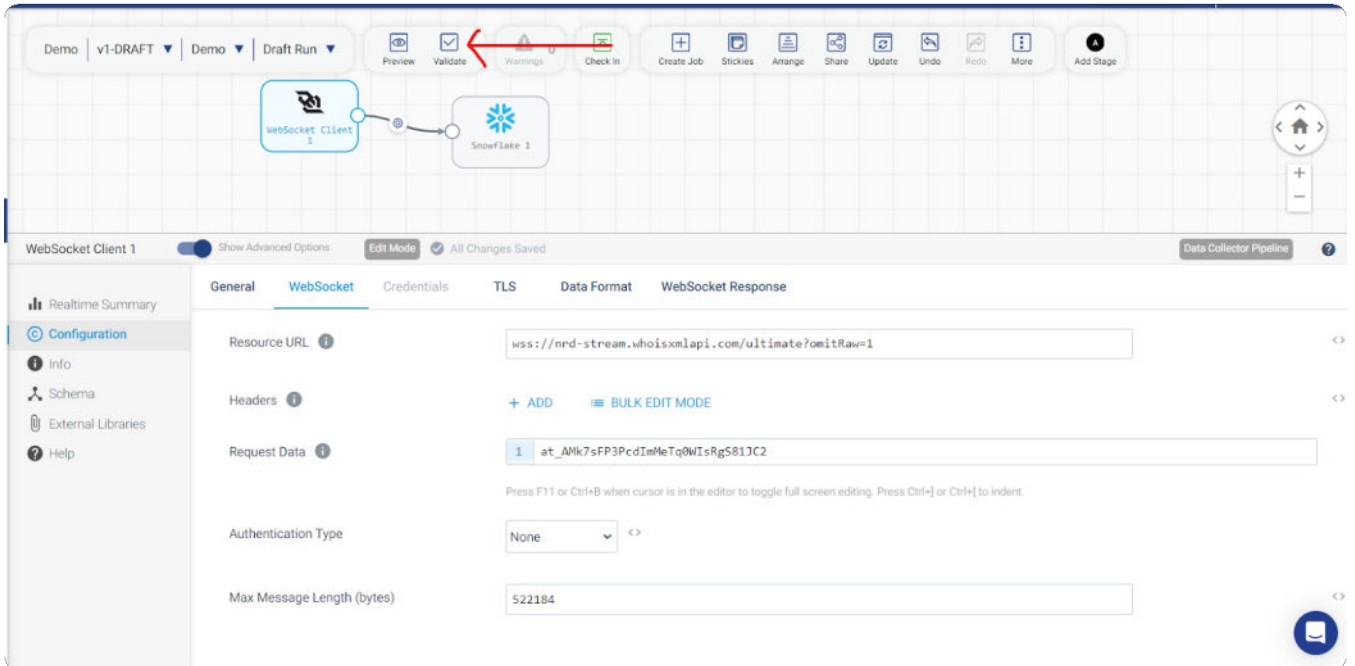
The screenshot shows the 'Data' configuration page. The left sidebar is the same as in the previous screenshot. The main content area has tabs for 'General', 'Snowflake Connection Info', 'Snowflake', 'Snowpipe', 'Staging', 'Staging Advanced', 'Data', and 'Data Advanced'. The 'Data' tab is active. The configuration fields are:

Data Format	CSV
Processing CDC Data (Use MERGE)	<input type="checkbox"/>
Hide Advanced Options	^
Row Field	/
Column Fields to Ignore	Column Fields to Ignore
Null Value	\N



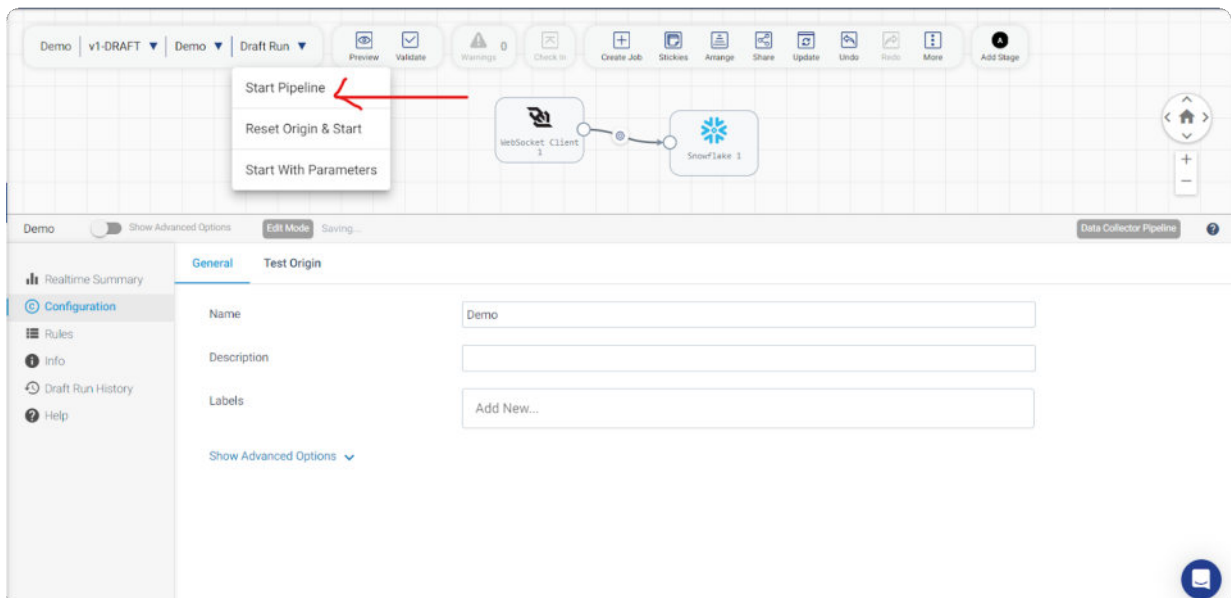
## 最終ステップ

設定後、「Validate」をクリックしてパイプラインを検証し、エラーを修正します。パイプラインの最終状態は、下図のようになっているはずですが。

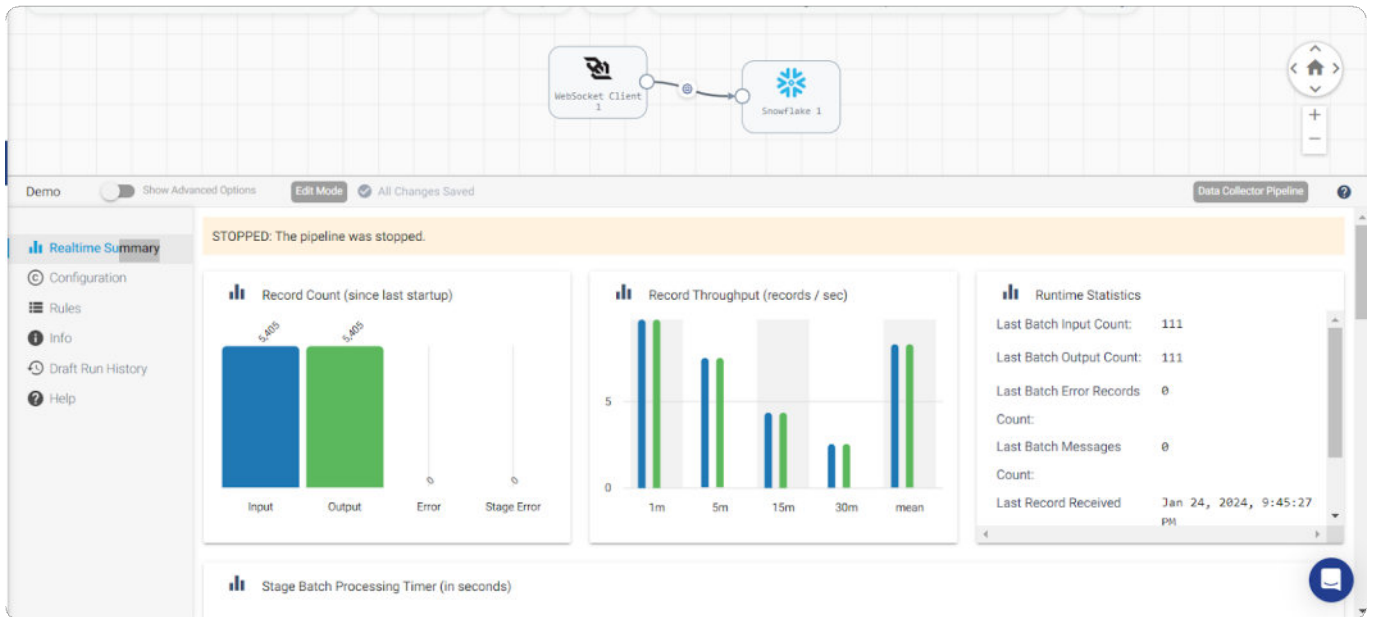


## パイプラインを実行する

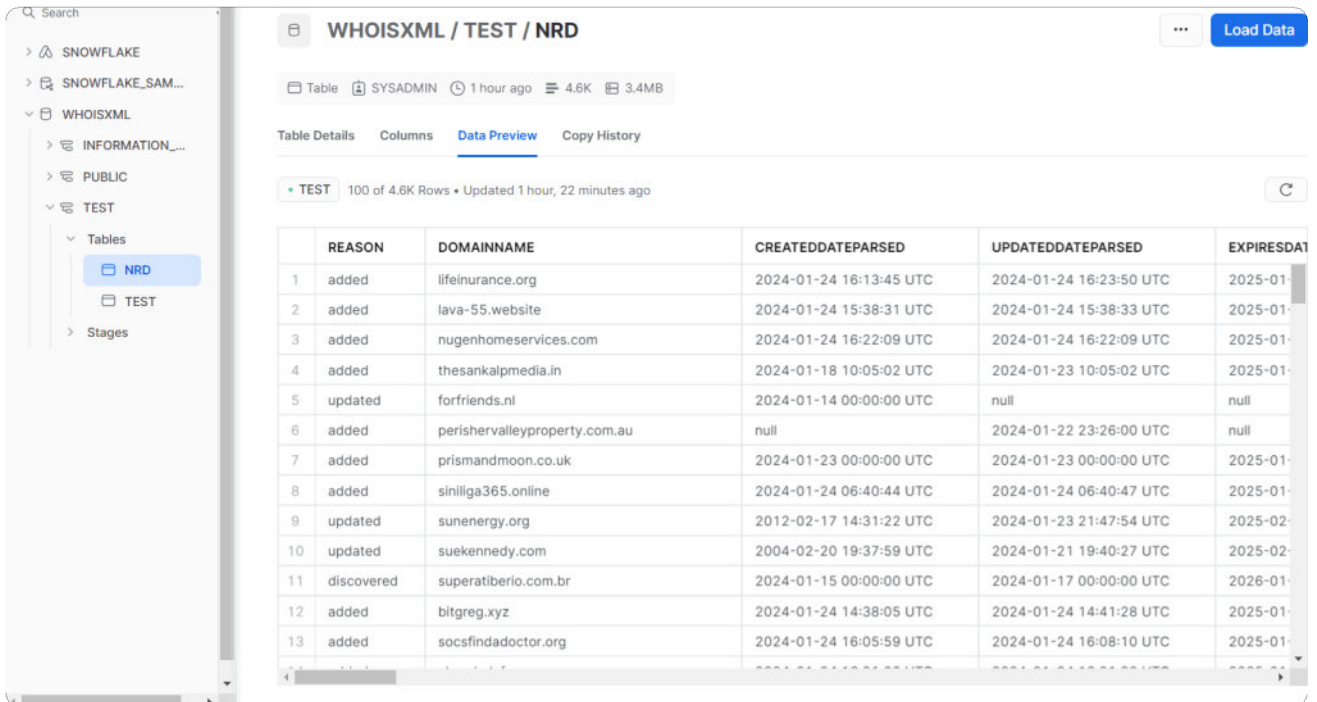
UIで「Draft Run」、「Start Pipeline」を選択してパイプラインを実行します。



前述の手順を完了してパイプラインを起動すると、下図のようなUIが表示されます。



Snowflakeデータベース内で、WebSocketから取得したデータを含むテーブル上のデータを下図のようにプレビューできるはずです。



The screenshot shows the 'Data Preview' tab for a table named 'TEST' in the 'WHOISXML / TEST / NRD' schema. The table contains 100 rows of data. The first 13 rows are visible:

	REASON	DOMAINNAME	CREATEDDATEPARSED	UPDATEDDATEPARSED	EXPIRESDATE
1	added	lifeinsurance.org	2024-01-24 16:13:45 UTC	2024-01-24 16:23:50 UTC	2025-01-
2	added	lava-55.website	2024-01-24 15:38:31 UTC	2024-01-24 15:38:33 UTC	2025-01-
3	added	nugenhomeservices.com	2024-01-24 16:22:09 UTC	2024-01-24 16:22:09 UTC	2025-01-
4	added	thesankalpmedia.in	2024-01-18 10:05:02 UTC	2024-01-23 10:05:02 UTC	2025-01-
5	updated	forfriends.nl	2024-01-14 00:00:00 UTC	null	null
6	added	perishervalleyproperty.com.au	null	2024-01-22 23:26:00 UTC	null
7	added	prismandmoon.co.uk	2024-01-23 00:00:00 UTC	2024-01-23 00:00:00 UTC	2025-01-
8	added	siniliga365.online	2024-01-24 06:40:44 UTC	2024-01-24 06:40:47 UTC	2025-01-
9	updated	sunenergy.org	2012-02-17 14:31:22 UTC	2024-01-23 21:47:54 UTC	2025-02-
10	updated	suekennedy.com	2004-02-20 19:37:59 UTC	2024-01-21 19:40:27 UTC	2025-02-
11	discovered	superatiberio.com.br	2024-01-15 00:00:00 UTC	2024-01-17 00:00:00 UTC	2026-01-
12	added	bitgreg.xyz	2024-01-24 14:38:05 UTC	2024-01-24 14:41:28 UTC	2025-01-
13	added	socsfindadoctor.org	2024-01-24 16:05:59 UTC	2024-01-24 16:08:10 UTC	2025-01-

## まとめ

本ガイドでは、**Docker**を使用して**StreamSets UI**でパイプラインを設定および実行する手順を概説しました。**WebSocket Client**の設定から**Snowflake**ステージの定義までの各ステップが、データ処理のパイプライン構築に貢献しています。検証ステップではパイプラインの整合性を担保し、実行に成功すると**Snowflake**のUIにアウトプットが表示されます。このガイドで説明した手順を踏むことで、合理的なデータ処理パイプラインの作成、設定および実行、ならびにデータの効果的な統合と管理を実現できます。