

HULFT Squareアプリケーション仕様書

マーケティングソリューション  
Snowflake

1.0 版 | 2024 年 6 月 13 日

## Snowflake向けHULFT Squareアプリケーション Overview

### 本アプリケーションについて

本アプリケーションはSnowflakeとのデータ連携を行うためのスクリプトのサンプルをアプリケーション化したものとなる

機能については機能一覧シートを参照

コピー後、状況に合わせてカスタマイズすることを推奨する

コピー後、編集なしでスクリプトを実行する場合は、実行前にスクリプトのビルド（デザイン画面赤枠）を実施する



### INDEX

#### 本スクリプトについて

- ・機能一覧

#### スクリプト利用手順

- ・設定フロー
- ・環境設定\_Snowflake側
- ・コネクション設定\_HULFT Square側
- ・スクリプト利用上注意

#### スクリプト詳細

- ・Snowflake\_from\_SFA [Salesforce情報CSVのSnowflakeへのストア]
- ・Snowflake\_from\_SHANON [MA(SHANON)情報CSVのSnowflakeへのストア]
- ・Snowflake\_from\_Marketo [MA(Marketo)情報CSVのSnowflakeへのストア]
- ・Snowflake\_from\_e-Stat\_0000010103 [政府統計(e-Stat)情報CSVのSnowflakeへのストア]
- ・Snowflake\_from\_CORP\_INFO [企業情報検索サービスCSVファイルのSnowflakeへの更新]
- ・Snowflake\_from\_AOAI [AOAI返答CSVファイルのSnowflakeへの更新]
- ・Snowflake\_to\_CORP\_INFO [企業情報検索サービスへの問い合わせ用CSVファイル作成]
- ・Snowflake\_to\_SFA\_and\_MA [企業情報検索サービスからのCSVファイル作成]
- ・Snowflake\_to\_e-Stat\_for\_AOAI [e-Statトークン精査]

**Snowflake向けHULFT Squareアプリケーション**  
アプリケーション概要

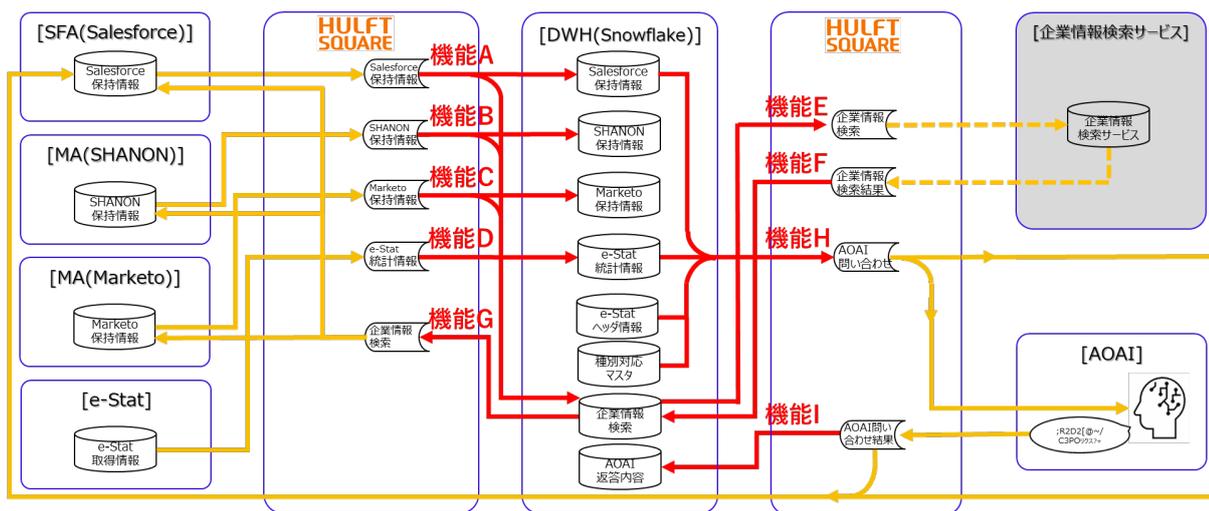
**1.動作イメージ**

本アプリケーションは、下図のようなMAソリューションをイメージしたシステムで、Snowflakeと組み合わせて動作することを想定している

本アプリケーションをインストールすることで、下記機能を持ったスクリプトのサンプルを使用することができる

- A SalesforceからダウンロードしたAccount情報(CSV)をSnowflakeの所定のテーブルにストアするとともに、企業情報検索サービスで企業コードを検索するための情報もストアする
- B SHANONからダウンロードした企業情報(CSV)をSnowflakeの所定のテーブルにストアするとともに、企業情報検索サービスで企業コードを検索するための情報もストアする
- C Marketoからダウンロードした企業情報(CSV)をSnowflakeの所定のテーブルにストアするとともに、企業情報検索サービスで企業コードを検索するための情報もストアする
- D e-Statからダウンロードした政府統計情報(CSV)をSnowflakeの所定のテーブルにストアする
- E Snowflakeに保持した企業情報検索サービスの検索用情報をCSVファイルにダウンロードする
- F 企業情報検索サービスの検索結果(CSV)をSnowflakeの所定のテーブルにストアする
- G Snowflakeに保持した企業情報検索サービスの検索結果をCSVファイルにダウンロードする
- H Snowflakeに保持したe-Stat統計情報をコードから名称変換しAOAIに問い合わせするためのCSVファイルにダウンロードする
- I AOAIからの問い合わせ結果をSnowflakeの所定のテーブルにストアする

本アプリケーションの動作範囲は下図の赤線部分となり、機能A~Iが存在する



2.各機能とスクリプト

機能A：Salesforce情報CSVのSnowflakeへのストア

機能概要：SalesforceからダウンロードしたAccount情報(CSV)をSnowflakeの所定のテーブルにストアするとともに、企業情報検索サービスの企業コードを検索するための情報もストアする  
 使用スクリプト一覧：

ID	スクリプト名	対応機能
A-1	Snowflake_from_SFA	Salesforce情報CSVファイルをSnowflakeにアップロードする

A-1.Snowflake\_from\_SFA



前提条件	Snowflakeはあらかじめログイン登録を行い、所定のテーブル配置しておく事
入力データ	Salesforce_ACCOUNT.csv (Salesforce保持情報CSVファイル)
出力データ	Snowflake.SALESFORCE_ACCOUNT (Salesforce保持情報テーブル) Snowflake.CORP_INFO (企業情報検索テーブル)
エラー処理	例外発生時に処理途中終了

機能B：SHANON情報CSVのSnowflakeへのストア

機能概要：SHANONからダウンロードした企業情報(CSV)をSnowflakeの所定のテーブルにストアするとともに、企業情報検索サービスの企業コードを検索するための情報もストアする  
 使用スクリプト一覧：

ID	スクリプト名	対応機能
B-1	Snowflake_from_SHANON	SHANON情報CSVファイルをSnowflakeにアップロードする

B-1.Snowflake\_from\_SHANON



前提条件	Snowflakeはあらかじめログイン登録を行い、所定のテーブル配置をしておく事
入力データ	SHANON_COMPANY.csv (SHANON保持情報CSVファイル)
出力データ	Snowflake.SHANON_COMPANY (SHANON保持情報テーブル) Snowflake.CORP_INFO (企業情報検索テーブル)
エラー処理	例外発生時に処理途中終了

機能C：Marketo情報CSVのSnowflakeへのストア

機能概要：Marketoからダウンロードした企業情報(CSV)をSnowflakeの所定のテーブルにストアするとともに、企業情報検索サービスの企業コードを検索するための情報もストアする  
 使用スクリプト一覧：

ID	スクリプト名	対応機能
C-1	Snowflake_from_Marketo	Marketo情報CSVファイルをSnowflakeにアップロードする

C-1.Snowflake\_from\_Marketo



前提条件	Snowflakeはあらかじめログイン登録を行い、所定のテーブル配置をしておく事
入力データ	Marketo_COMPANY.csv (Marketo保持情報CSVファイル)
出力データ	Snowflake.MARKETO_COMPANY (Marketo保持情報テーブル) Snowflake.CORP_INFO (企業情報検索テーブル)
エラー処理	例外発生時に処理途中終了

機能D：e-Stat情報CSVのSnowflakeへのストア

機能概要：e-Statから取得した政府統計情報(CSV)をSnowflakeの所定のテーブルにストアする

使用スクリプト一覧：

ID	スクリプト名	対応機能
D-1	Snowflake_from_e-Stat_0000010103	e-Stat情報CSVファイルをSnowflakeにアップロードする

D-1.Snowflake\_from\_e-Stat\_0000010103



前提条件	Snowflakeはあらかじめログイン登録を行い、所定のテーブル配置をしておく事
入力データ	e-Stat_CENSUS_0000010103.csv (e-Stat統計情報CSVファイル)
出力データ	Snowflake.ESTAT_STATISTICS (e-Stat統計情報テーブル)
エラー処理	例外発生時に処理途中終了

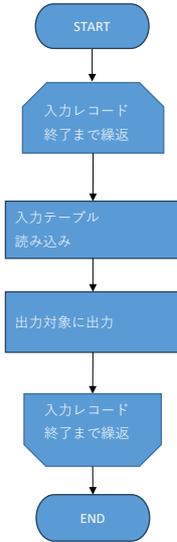
機能E：企業情報検索テーブルからのCSVファイル作成

機能概要：Snowflakeに保持した企業情報検索サービスの検索用情報をCSVファイルにダウンロードする

使用スクリプト一覧：

ID	スクリプト名	対応機能
E-1	Snowflake_to_CORP_INFO	企業情報検索サービス用検索情報をSnowflakeからダウンロードする

E-1.Snowflake\_to\_CORP\_INFO



前提条件	Snowflakeはあらかじめログイン登録を行い、所定のテーブル配置をしておく事
入力データ	Snowflake.CORP_INFO (企業情報検索テーブル)
出力データ	CORP_INFO_INQUIRY.csv (企業情報検索CSVファイル)
エラー処理	例外発生時に処理途中終了

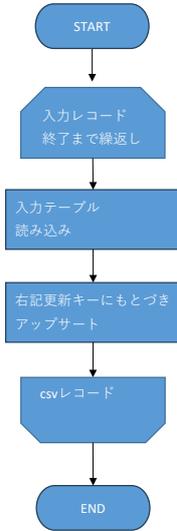
**機能F：企業情報検索結果CSVファイルのSnowflakeへの更新**

機能概要：企業情報検索サービス検索結果(CSV)をSnowflakeの所定のテーブルにストアする

使用スクリプト一覧：

ID	スクリプト名	対応機能
F-1	Snowflake_from_CORP_INFO	企業情報検索サービスからの問い合わせ結果をSnowflakeにアップロードする

F-1.Snowflake\_from\_CORP\_INFO



前提条件	Snowflakeはあらかじめログイン登録を行い、所定のテーブル配置をしておく事
入力データ	CORP_INFO_RESULTS.csv（企業情報検索結果CSVファイル）
出力データ	Snowflake.CORP_INFO（企業情報検索テーブル）
エラー処理	例外発生時に処理途中終了

**更新キー**

次の項目をキーとして既に出力側に存在する場合レコード更新し、ない場合はなにもしない

- ・Salesforce企業コード
- ・SHANON企業コード
- ・Marketo企業コード

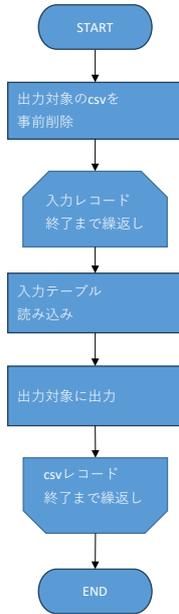
機能G：企業情報検索テーブルからのCSVファイル作成

機能概要：Snowflakeに保持した企業情報検索サービスの検索結果をCSVファイルにダウンロードする

使用スクリプト一覧：

ID	スクリプト名	対応機能
G-1	Snowflake_to_SFA_and_MA	企業情報検索結果をSnowflakeからダウンロードする

G-1.Snowflake\_to\_SFA\_and\_MA



前提条件	Snowflakeはあらかじめログイン登録を行い、所定のテーブル配置をしておく事
入力データ	Snowflake.CORP_INFO（企業情報検索テーブル）
出力データ	CORP_INFO.csv（企業情報検索CSVファイル）
エラー処理	例外発生時に処理途中終了

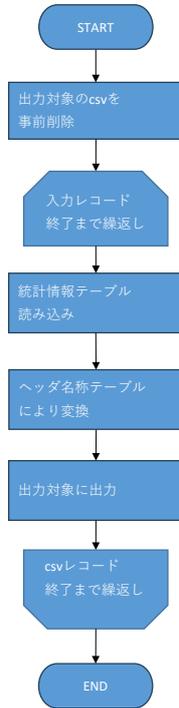
機能H：e-Statトークン精査

機能概要：e-Statの統計情報をヘッダ項目名称変換を行いSnowflakeからCSV形式にてダウンロードする

使用スクリプト一覧：

ID	スクリプト名	対応機能
H-1	Snowflake_to_e-Stat_for_AOAI	e-Statの統計情報をダウンロードする

H-1.Snowflake\_to\_e-Stat\_for\_AOAI



前提条件	Snowflakeはあらかじめログイン登録を行い、所定のテーブル配置をしておく事
入力データ	ESTAT_HEADER (e-Statヘッダ名称テーブル) ESTAT_STATISTICS (e-Stat統計情報テーブル)
出力データ	e-Stat_0000010103_FOR_AOAI.csv (AOAI問い合わせ用e-Stat情報CSVファイル)
エラー処理	例外発生時に処理途中終了

※抽出対象：e-Stat統計情報テーブル,ヘッダ情報3 = '13000' (※都道府県=東京都)  
e-Stat統計情報テーブル,ヘッダ情報4 LIKE '2014%' (※2014年度情報)

※事業所種別変換 (下記条件によりe-Statヘッダ名称テーブル,ヘッダ情報名称を得る)  
e-Statヘッダ名称テーブル,統計情報番号 = '0000010103'、 e-Statヘッダ名称テーブル,ヘッダ情報順 = '2'  
e-Statヘッダ名称テーブル,ヘッダ情報コード = e-Stat統計情報テーブル,ヘッダ情報2

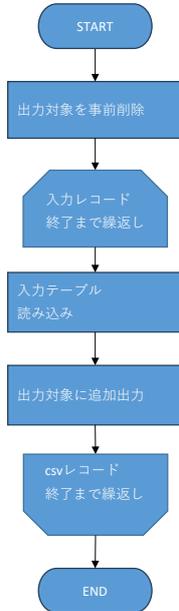
機能I：AOAI返答CSVファイルのSnowflakeへの更新

機能概要：AOAIからの返答内容CSVをSnowflakeにストアする

使用スクリプト一覧：

ID	スクリプト名	対応機能
I-1	Snowflake_from_AOAI	AOAI返答内容のストア

I-1.Snowflake\_from\_AOAI



前提条件	Snowflakeはあらかじめログイン登録を行い、所定のテーブル配置をしておく事
入力データ	AOAI_e-Stat_0000010103_RESULTS.csv (AOAI問い合わせ結果統計情報)
出力データ	Snowflake.AOAI_REPLY (AOAI返答内容テーブル)
エラー処理	Exception発生時には処理は異常終了となり、出力テーブルはロールバックされる

Snowflake向けHULFT Squareアプリケーション  
設定フロー

本アプリケーションの設定フロー

1. [環境設定(Snowflake側)]に沿って設定を行う
  - ・Snowflakeのデータベース設定構築を行い、Snowflakeへのアクセスを利用可能にする
2. [コネクション設定(HULFT Square側)]に沿って設定を行う
  - ・HULFT SquareからSnowflakeへ接続するための設定を作成する

## Snowflake連携サンプルスクリプト 環境設定(Snowflake側)

### コネクション設定

HULFT SquareとSnowflakeの接続に必要な設定について記述する

#### Snowflake側設定(前提条件)

前提条件として、Snowflake側でデータベース構築設定が完了していること

0. まずはSnowflakeの無料アカウント登録（もしくはサブスクリプション契約）をSnowflakeサイトにてサインアップし、アカウントを作成する  
※この手順はSnowflakeにサインアップした事がない場合、もしくは既存のSnowflakeアカウント以外で当スクリプトを実行したい場合に行う  
サインアップ画面上に表記されるエディション、クラウドプロバイダー、リージョン等は適宜選択する

#### 1. Snowflake上にてDBを構築する

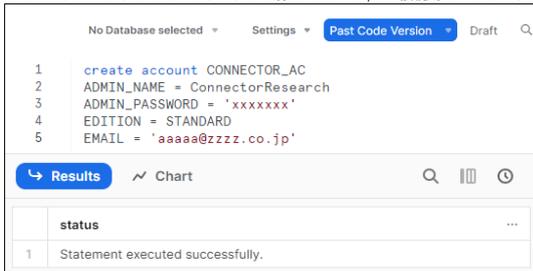
SnowflakeのGUIのworksheetにてSQLを投入する

- 1-1.HULFT Squareとの接続に使用するアカウントの作成
- 1-2.作成したアカウントに権限付与
- 1-3.ロケータ確認
- 1-4.新規ロケータアクセス
- 1-5.仮想ウェアハウス作成
- 1-6.データベース作成
- 1-7.スキーマ作成
- 1-8.各テーブル作成

※1-1～7について

既にSnowflakeの構成を行っており、構成した既存の環境を使用する場合は行う必要はない

1-1. ここではサインアップしたアカウントとは別にHULFT Square接続用のアカウントを作成している

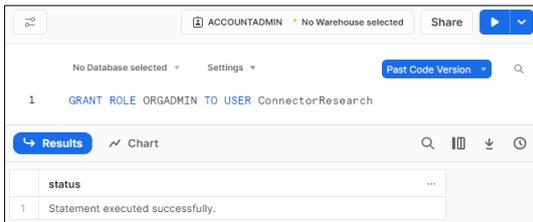


```

CREATE ACCOUNT CONNECTOR_AC ADMIN_NAME = ConnectorResearch
ADMIN_PASSWORD = 'xxxxxxx'
EDITION = STANDARD
EMAIL = 'aaaaa@zzzz.co.jp'
    
```

※ADMIN\_NAMEはサインアップした際のアカウント名  
 ※EDITIONは適宜選択

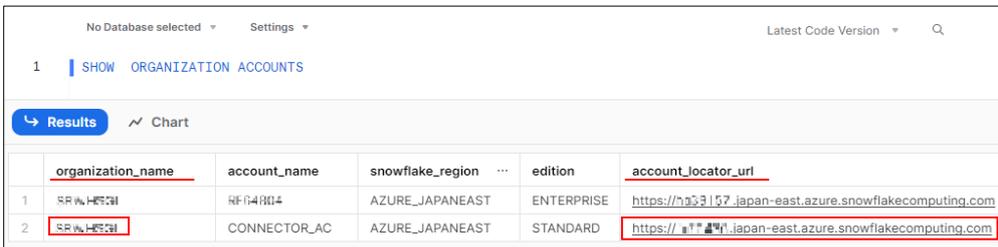
1-2. 作成したアカウントに対して権限付与



```
GRANT ROLE ORGADMIN TO USER ConnectorResearch
```

※ここでは1-3の「SHOW ORGANIZATION ACCOUNTS」を実行するための権限を付与している

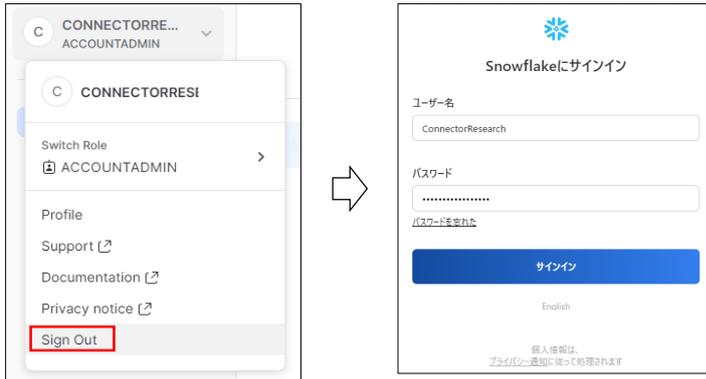
1-3. アカウント環境の表示を行い、ロケータを確認する



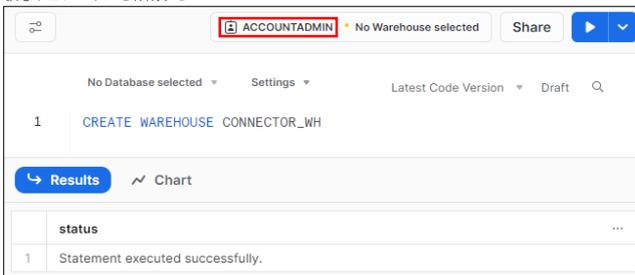
SHOW ORGANIZATION ACCOUNTS

※1-1で作成したアカウントのロケータを確認する  
 ※organization\_nameは「コネクション設定(HULFT Square側)」において設定項目となるので確認しておく

1-4. Snowflakeから一旦サインアウトし、1-1で作成したアカウントで再ログインするため、1-3で確認したロケータURL(account\_locator\_url)にアクセスする



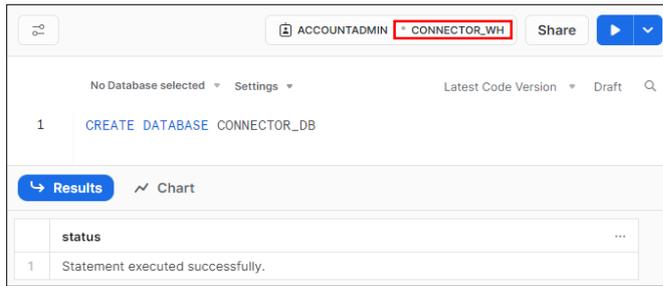
1-5. 仮想ウェアハウスを作成する



```
CREATE WAREHOUSE CONNECTOR_WH
```

※ACCOUNTADMINにて実行する

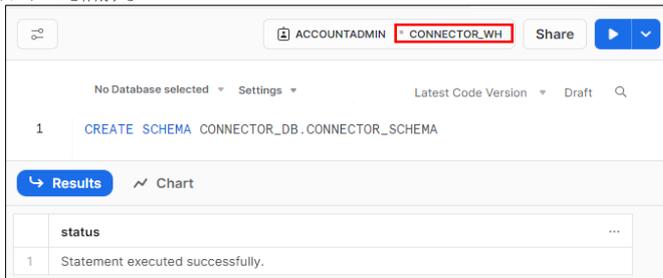
1-6. データベースを作成する



CREATE DATABASE CONNECTOR\_DB

※1-5で作成したウェアハウスを指定して実行する

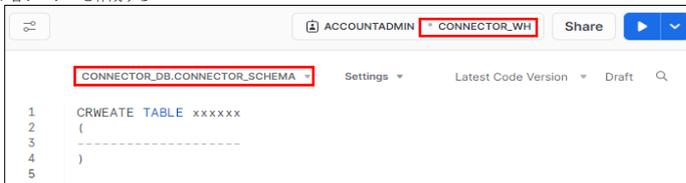
1-7. スキーマを作成する



CREATE SCHEMA CONNECT\_DB.CONNECT\_SCHEMA

※1-5で作成したウェアハウスを指定して実行する

1-8. 各テーブルを作成する



※1-7で作成したスキーマを指定して実行する  
各テーブル作成SQLは下記の通り

1-8. 各テーブル作成SQL

```
AOAI返答内容テーブル
CREATE TABLE AOAI_REPLY
(
  QUESTION VARCHAR,
  REPLY_ITEM VARCHAR,
  ALERT VARCHAR
)
```

```
e-Statヘッダ名称テーブル
CREATE TABLE ESTAT_HEADER
(
  STATS_DATA_ID VARCHAR NOT NULL,
  HEADER_SEQ VARCHAR NOT NULL,
  CODE VARCHAR NOT NULL,
  NAME VARCHAR NOT NULL
)
```

```
e-Stat統計情報テーブル
CREATE TABLE ESTAT_STATISTICS
(
  STATS_DATA_ID VARCHAR NOT NULL,
  HEADER_01 VARCHAR,
  HEADER_02 VARCHAR,
  HEADER_03 VARCHAR,
  HEADER_04 VARCHAR,
  HEADER_05 VARCHAR,
  STATS_VALUE NUMBER
)
```

```
Salesforce_e-Stat種別対応マスタ
CREATE TABLE SF_ESTAT_TYPES (
  SF_TYPES VARCHAR,
  ESTAT_TYPES_CODE VARCHAR,
  ESTAT_TYPES_NAME VARCHAR
);
```

```
企業情報検索テーブル
CREATE TABLE CORP_INFO (
  COMPANY_CODE_SF VARCHAR,
  COMPANY_CODE_SHANON VARCHAR,
  COMPANY_CODE_MARKETO VARCHAR,
  CORP_INFO_CODE VARCHAR,
  COMPANY_NAME VARCHAR,
  TEL VARCHAR,
  SEC_CODE VARCHAR,
  COMPANY_STAT VARCHAR,
  OFFICE_STAT VARCHAR,
  CAPITAL NUMBER,
  FIRSTTERM_AMOUNT NUMBER,
  REPRESENTATIVE VARCHAR
);
```

```
Salesforce保持情報テーブル
CREATE TABLE SALESFORCE_ACCOUNT
(
  CORP_INFO_CODE VARCHAR NOT NULL,
  ID VARCHAR NOT NULL,
  IS_DELETED BOOLEAN,
  MASTER_RECORD_ID VARCHAR,
  NAME VARCHAR,
  TYPE VARCHAR,
  PARENT_ID VARCHAR,
  BILLING_STREET VARCHAR,
  BILLING_CITY VARCHAR,
  BILLING_STATE VARCHAR,
  BILLING_POATAL_CODE VARCHAR,
  BILLING_COUNTRY VARCHAR,
  BILLING_LATITUDE NUMBER(38, 4),
  BILLING_LONGITUDE NUMBER(38, 4),
  BILLING_GEOCODE_ACCURACY VARCHAR,
  BILLING_ADDRESS VARCHAR,
  SHIPPING_STREET VARCHAR,
  SHIPPING_CITY VARCHAR,
  SHIPPING_STATE VARCHAR,
  SHIPPING_POATAL_CODE VARCHAR,
  SHIPPING_COUNTRY VARCHAR,
  SHIPPING_LATITUDE NUMBER(38, 4),
  SHIPPING_LONGITUDE NUMBER(38, 4),
  SHIPPING_GEOCODE_ACCURACY VARCHAR,
  SHIPPING_ADDRESS VARCHAR,
  PHONE VARCHAR,
  FAX VARCHAR,
  ACCOUNT_NUMBER VARCHAR,
  WEBSITE VARCHAR,
  PHOTO_URL VARCHAR,
  SIC VARCHAR,
  INDUSTRY VARCHAR,
  ANNUAL_REVENUE NUMBER,
  NUMBER_OF_EMPLOYEES VARCHAR,
  OWNERSHIP VARCHAR,
  TICKER_SYMBOL VARCHAR,
  DESCRIPTION VARCHAR,
  RATING VARCHAR,
  SITE VARCHAR,
  OWNER_ID VARCHAR,
  CREATED_DATE TIMESTAMP_NTZ,
  CREATED_BY_ID VARCHAR,
  LAST_MODIFIED_DATE TIMESTAMP_NTZ,
  LAST_MODIFIED_BY_ID VARCHAR,
  SYSTEM_MODSTAMP TIMESTAMP_NTZ,
  LAST_ACTIVITY_DATE TIMESTAMP_NTZ,
  LAST_VIEWED_DATE TIMESTAMP_NTZ,
  LAST_REFERENCED_DATE TIMESTAMP_NTZ,
  JIGSAW VARCHAR,
  JIGSAW_COMPANY_ID VARCHAR,
  CLEAN_STATUS VARCHAR,
  ACCOUNT_SOURCE VARCHAR,
  DUNS_NUMBER VARCHAR,
  TRADE_STYLE VARCHAR,
  NAICS_CODE VARCHAR,
  NAICS_DESC VARCHAR,
  YEAR_STARTED NUMBER,
  SIC_DESC VARCHAR,
  DANDB_COMPANY_ID VARCHAR,
  OPERATING_HOURS_ID VARCHAR
)
```

```
SHANON保持情報テーブル
CREATE TABLE SHANON_COMPANY
(
  CORP_INFO_CODE VARCHAR NOT NULL,
  ID VARCHAR NOT NULL,
  DATE_REGIST TIMESTAMP_NTZ,
  DATE_UPDATE TIMESTAMP_NTZ,
  NAME VARCHAR,
  NAME_KA VARCHAR,
  ZIP1 VARCHAR,
  ZIP2 VARCHAR,
  ZIP3 VARCHAR,
  COUNTRY VARCHAR,
  PREFECTURE VARCHAR,
  ADDRESS1 VARCHAR,
  ADDRESS2 VARCHAR,
  ADDRESS3 VARCHAR,
  ADDRESS4 VARCHAR,
  ADDRESS5 VARCHAR,
  TEL VARCHAR,
  SUB_TEL VARCHAR,
  FAX VARCHAR,
  EMAIL_DOMAIN VARCHAR,
  DATE_OF_ESTABLISHMENT VARCHAR,
  URL VARCHAR
)
```

```
Marketo企業情報テーブル
CREATE TABLE MARKETO_COMPANY (
  BILLINGCOUNTRY VARCHAR,
  BILLINGPOSTALCODE VARCHAR,
  WENSITE VARCHAR,
  COMPANYNOTES VARCHAR,
  INDUSTRY VARCHAR,
  COMPANY VARCHAR,
  ANNUALREVENUE NUMBER,
  NUMBEROFEMPLOYEES VARCHAR,
  SITE VARCHAR,
  EXTERNALCOMPANTID VARCHAR,
  MAINPHONE VARCHAR,
  BILLINGCITY VARCHAR,
  SICCODE VARCHAR,
  BILLINGSTATE VARCHAR,
  BILLINGSTREET VARCHAR
);
```

```
SHANON企業情報テーブル
CREATE TABLE SHANON_COMPANY (
  CORP_INFO_CODE VARCHAR NOT NULL,
  ID VARCHAR NOT NULL,
  DATE_REGIST TIMESTAMP,
  DATE_UPDATE TIMESTAMP,
  NAME VARCHAR,
  NAME_KA VARCHAR,
  ZIP1 VARCHAR,
  ZIP2 VARCHAR,
  ZIP3 VARCHAR,
  COUNTRY VARCHAR,
  PREFECTURE VARCHAR,
  ADDRESS1 VARCHAR,
  ADDRESS2 VARCHAR,
  ADDRESS3 VARCHAR,
  ADDRESS4 VARCHAR,
  ADDRESS5 VARCHAR,
  TEL VARCHAR,
  SUB_TEL VARCHAR,
  FAX VARCHAR,
  EMAIL_DOMAIN VARCHAR,
  DATE_OF_ESTABLISHMENT VARCHAR,
  URL VARCHAR
);
```

※Snowflakeではテーブル名、項目名が全角の場合はダブルクォーテーションで括る

## Snowflake連携サンプルスクリプト コネクション設定(HULFT Square側)

### コネクション設定

HULFT SquareとSnowflakeの接続に必要な設定に関して記述する

### HULFT Square側設定

Snowflakeに接続するための設定を作成する

「コネクション>新規作成」設定にて下記を選択する

- ・コネクタ：Snowflake
- ・種類：Snowflake接続設定

接続用コネクション設定

アカウント識別子	「環境設定 (Snowflake側)」 [1-3]organization_name + "-" + [1-1]アカウント名
ウェアハウス名	「環境設定 (Snowflake側)」 [1-5]で作成したウェアハウス名
データベース名	「環境設定 (Snowflake側)」 [1-6]で作成したデータベース名
スキーマ名	「環境設定 (Snowflake側)」 [1-7]で作成したスキーマ名
Enter URL directly	随意
URLを直接入力する	Enter URL directlyをチェックした場合、URLを指定 (指定した際はアカウント識別子～スキーマ名の設定の必要なし)
ユーザ名	「環境設定 (Snowflake側)」 [1-1]で作成したユーザ名
パスワード	「環境設定 (Snowflake側)」 [1-1]で指定したパスワード

## Snowflake向けHULFT Squareアプリケーション スクリプト利用上注意

### 利用について

- ・本スクリプトはHULFT SquareのSnowflakeコネクタの使用を前提とする
- ・想定と異なる出力結果、もしくは異常終了した場合、HULFT Squareの実行ログを参照しエラーを確認する必要がある  
(後述の「エラーログの収集について」を参照)

### エラーのログ収集について

HULFT Squareの実行ログは以下から取得できる

[モニタリング]>[サービス]>[HULFT Integrate]より、該当ログのメニューから[エクスポート]

サービス>

### HULFT Integrate

+ フィルタを追加

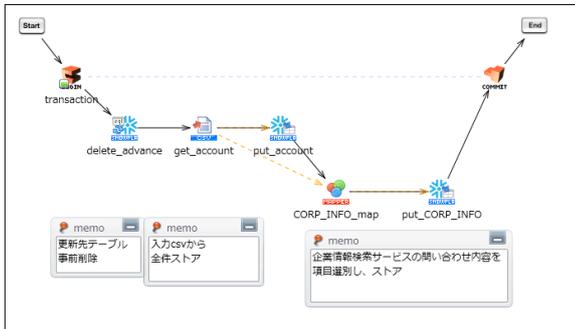
🔄 設定 🗑️

ステータス	プロジェクト名	バージョン	スクリプト名	プロファイル名
📄 ✓	██████████	204	get_token	Profil: ██████████
📄 ✓	██████████			Profil: ██████████

🔍 詳細  
📄 エクスポート

**Snowflake向けHULFT Squareアプリケーション  
スクリプト設定(from\_SFA)**

スクリプト全体図



スクリプト動作概要

Salesforceからの企業情報CSVをSnowflakeにストアする

\*以下設定のオレンジ色の設定箇所は、スクリプトをインストールするだけでは設定されないため、スクリプトをインストール後に手動で入力する必要がある

スクリプト基本設定

HULFT Square URL	
ワークスペース	
格納プロジェクト名	Snowflake
格納フォルダ名	
スクリプト名	Snowflake_from_SFA

外部システムから取り込む値/ファイル

ファイル [Salesforce_ACCOUNT.csv]	項目構成については該当コンポーネントの列一覧を参照のこと
-------------------------------	------------------------------

スクリプト変数

_InputDataPathAndName	入力元となるCSVファイルのパスとファイル名(/<ワークスペース名>/Salesforce/Salesforce_ACCOUNT.csv)
-----------------------	---

スクリプト詳細設定

コンポーネント名	transaction
コンポーネント概要	処理単位開始 通常概念のトランザクションと同義、COMMIT発生により当トランザクション配下の処理は実更新される

コンポーネント名	delete_advance
コンポーネント概要	「更新系SQL実行処理」コンポーネントにより、出力テーブルレコードを事前削除する
必須設定	
接続先	{APIリクエスト用コネクション} *[コネクション設定(HULFT Square側)]参照
SQL文	DELETE FROM SALESFORCE_ACCOUNT
スキーマ定義	設定なし

コンポーネント名	
get_account	
	
コンポーネント概要	
「CSVファイル読み込み処理」コンポーネントにより、Salesforceからの企業情報CSVを読み込む	
ファイル	右記をスクリプト変数に格納 /{利用するワークスペース名}/Salesforce/Salesforce_ACCOUNT.csv
デリミタ文字	半角カンマ
列一覧	CORP_INFO_CODE ID IS_DELETEDBOOLEAN MASTER_RECORD_ID NAME TYPE PARENT_ID BILLING_STREET BILLING_CITY BILLING_STATE BILLING_POATAL_CODE BILLING_COUNTRY BILLING_LATITUDE BILLING_LONGITUDE BILLING_GEOCODE_ACCURACY BILLING_ADDRESS SHIPPING_STREET SHIPPING_CITY SHIPPING_STATE SHIPPING_POATAL_CODE SHIPPING_COUNTRY SHIPPING_LATITUDE SHIPPING_LONGITUDE SHIPPING_GEOCODE_ACCURACY SHIPPING_ADDRESS PHONE FAX ACCOUNT_NUMBER WEBSITE PHOTO_URL SIC INDUSTRY ANNUAL_REVENUENUMBER NUMBER_OF_EMPLOYEES OWNERSHIP TICKER_SYMBOL DESCRIPTION RATING SITE OWNER_ID CREATED_DATE CREATED_BY_ID LAST_MODIFIED_DATE LAST_MODIFIED_BY_ID SYSTEM_MODSTAMP LAST_ACTIVITY_DATE LAST_VIEWED_DATE LAST_REFERENCED_DATE JIGSAW JIGSAW_COMPANY_ID CLEAN_STATUS ACCOUNT_SOURCE DUNS_NUMBER TRADE_STYLE NAICS_CODE NAICS_DESC YEAR_STARTEDNUMBER SIC_DESC DANDB_COMPANY_ID OPERATING_HOURS_ID
読み取り設定	
エンコード	UTF-8
読み取り開始行	1
最初の行は値として取得しない	有効

コンポーネント名	
put_account 	
「テーブル書き込み処理」コンポーネントにより、読み込んだCSVデータセットをデータベースに一括更新する（バルク更新）	
必須設定	
接続先	{接続用コネクション} *{コネクション設定(HULFT Square側)}参照
テーブル名	SALESFORCE_ACCOUNT
テーブル名を引用符で囲う	指定なし（※Snowflakeで日本語テーブル名を表記する際は引用符で囲う必要がある）
キーの値と一致する行は更新	指定なし（※このチェックによりUPSERT処理が可能となる）
列一覧	上記「get_account」コンポーネント項目と同様
書き込み設定	
インサートモード	バッチインサート（※この指定により一括更新を行うバルク処理となる）
バッチサイズ	1000(※デフォルト値：コミット行数単位)
クエリタイムアウト(秒)	指定なし

コンポーネント名											
CORP_INFO_map 											
「企業情報検索」テーブルに出力するための項目をマッピングする											
必須設定											
マッピング列一覧	<table border="1"> <tr> <td>get_account</td> <td>put_CORP_INFO</td> </tr> <tr> <td>企業情報検索サービスの企業コード</td> <td>CORP_INFO_CODE</td> </tr> <tr> <td>Salesforce取引先ID</td> <td>COMPANY_CODE_SF</td> </tr> <tr> <td>取引先名</td> <td>COMPANY_NAME</td> </tr> <tr> <td>取引先電話</td> <td>TEL</td> </tr> </table>	get_account	put_CORP_INFO	企業情報検索サービスの企業コード	CORP_INFO_CODE	Salesforce取引先ID	COMPANY_CODE_SF	取引先名	COMPANY_NAME	取引先電話	TEL
get_account	put_CORP_INFO										
企業情報検索サービスの企業コード	CORP_INFO_CODE										
Salesforce取引先ID	COMPANY_CODE_SF										
取引先名	COMPANY_NAME										
取引先電話	TEL										

コンポーネント名											
put_CORP_INFO 											
マッピングされたデータセットをデータベースに一括更新する（バルク更新）											
必須設定											
接続先	{接続用コネクション} *{コネクション設定(HULFT Square側)}参照										
テーブル名	CORP_INFO										
テーブル名を引用符で囲う	指定なし（※Snowflakeで日本語テーブル名を表記する際は引用符で囲う必要がある）										
キーの値と一致する行は更新	COMPANY_CODE_SF										
列一覧	<table border="1"> <tr><td>COMPANY_CODE_SF</td></tr> <tr><td>CORP_INFO_CODE</td></tr> <tr><td>COMPANY_NAME</td></tr> <tr><td>TEL</td></tr> <tr><td>SEC_CODE</td></tr> <tr><td>COMPANY_STAT</td></tr> <tr><td>OFFICE_STAT</td></tr> <tr><td>CAPTAL</td></tr> <tr><td>FIRSTTERM_AMOUNT</td></tr> <tr><td>REPRESENTATIVE</td></tr> </table>	COMPANY_CODE_SF	CORP_INFO_CODE	COMPANY_NAME	TEL	SEC_CODE	COMPANY_STAT	OFFICE_STAT	CAPTAL	FIRSTTERM_AMOUNT	REPRESENTATIVE
COMPANY_CODE_SF											
CORP_INFO_CODE											
COMPANY_NAME											
TEL											
SEC_CODE											
COMPANY_STAT											
OFFICE_STAT											
CAPTAL											
FIRSTTERM_AMOUNT											
REPRESENTATIVE											
書き込み設定											
インサートモード	バッチインサート（※この指定により一括更新を行うバルク処理となる）										
バッチサイズ	1000(※デフォルト値：コミット行数単位)										
クエリタイムアウト(秒)	指定なし										

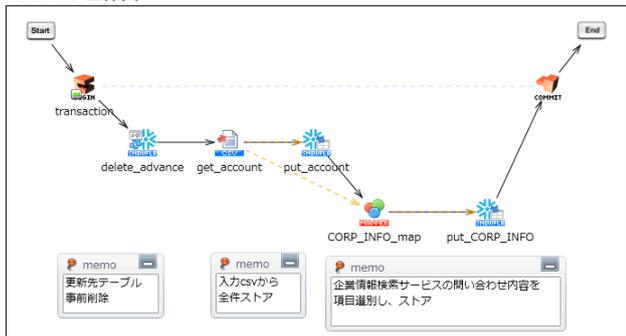
コンポーネント名	
commit 	
コンポーネント概要	
処理単位開始 通常概念のトランザクションのコミットと同義、COMMIT発生により当トランザクション配下の処理は変更される	

スクリプト利用手順

1. 「コネクション設定(Snowflake側)」を設定する
2. 「コネクション設定(HULFT Square側)」を設定する
3. Snowflakeにアクセスするコンポーネントの接続先に「コネクション設定\_HULFT Square側」で設定した接続先を指定する
4. [/{利用するワークスペース}/Salesforce]ディレクトリに[Salesforce\_ACCOUNT.csv]を配置する
5. スクリプトを実行する

**Snowflake向けHULFT Squareアプリケーション  
スクリプト設定(from\_SHANON)**

スクリプト全体図



スクリプト動作概要

SHANONからの企業情報CSVをSnowflakeにストアする

\*以下設定のオレンジ色の設定箇所は、スクリプトをインストールするだけでは設定されないため、スクリプトをインストール後に手動で入力する必要がある

スクリプト基本設定

HULFT Square URL	
ワークスペース	
格納プロジェクト名	Snowflake
格納フォルダ名	
スクリプト名	Snowflake_from_SHANON

外部システムから取り込む値/ファイル

ファイル [SHANON_COMPANY.csv]	項目構成については該当コンポーネントの列一覧を参照のこと
---------------------------	------------------------------

スクリプト変数

_InputDataPathAndName	入力元となるCSVファイルのパスとファイル名(/<ワークスペース名>/SHANON/SHANON_COMPANY.csv)
-----------------------	---

スクリプト詳細設定

コンポーネント名	transaction
コンポーネント概要	処理単位開始 通常概念のトランザクションと同義、COMMIT発生により当トランザクション配下の処理は実更新される

コンポーネント名	delete_advance
コンポーネント概要	「更新系SQL実行処理」コンポーネントにより、出力テーブルレコードを事前削除する
必須設定	
接続先	[APIリクエスト用コネクション] *[コネクション設定(HULFT Square側)]参照
SQL文	DELETE FROM SHANON_COMPANY
スキーマ定義	設定なし

コンポーネント名	
get_account 	
コンポーネント概要	
「CSVファイル読み込み処理」コンポーネントにより、SHANONからの企業情報CSVを読み込む	
ファイル	\$(InputDataPathAndName)
デリミタ文字	半角カンマ
列一覧	CORP_INFO_CODE ID DATE_REGIST DATE_UPDATE NAME NAME_KA ZIP1 ZIP2 ZIP3 COUNTRY PREFECTURE ADDRESS1 ADDRESS2 ADDRESS3 ADDRESS4 ADDRESS5 TEL SUB_TEL FAX EMAIL_DOMAIN DATE_OF_ESTABLISHMENT URL
読み取り設定	
エンコード	UTF-8
読み取り開始行	1
最初の行は値として取得しない	有効

コンポーネント名	
put_account 	
「テーブル書き込み処理」コンポーネントにより、読み込んだCSVデータセットをデータベースに一括更新する（バルク更新）	
必須設定	
接続先	{接続用コネクション} *{コネクション設定(HULFT Square側)}参照
テーブル名	SHANON_COMPANY
テーブル名を引用符で囲う	指定なし（※Snowflakeで日本語テーブル名を表記する際は引用符で囲う必要がある）
キーの値と一致する行は更新	指定なし（※このチェックによりUPSERT処理が可能となる）
列一覧	上記「get_account」コンポーネント項目と同様
書き込み設定	
インサートモード	バッチインサート（※この指定により一括更新を行うバルク処理となる）
バッチサイズ	1000(※デフォルト値：コミット行数単位)
クエリタイムアウト(秒)	指定なし

コンポーネント名											
CORP_INFO_map 											
「企業情報検索」テーブルに出力するための項目をマッピングする											
必須設定											
マッピング列一覧	<table border="1"> <tr> <td>get_account</td> <td>put_CORP_INFO</td> </tr> <tr> <td>CORP_INFO_CODE</td> <td>CORP_INFO_CODE</td> </tr> <tr> <td>ID</td> <td>COMPANY_CODE_SHANON</td> </tr> <tr> <td>NAME</td> <td>COMPANY_NAME</td> </tr> <tr> <td>TEL</td> <td>TEL</td> </tr> </table>	get_account	put_CORP_INFO	CORP_INFO_CODE	CORP_INFO_CODE	ID	COMPANY_CODE_SHANON	NAME	COMPANY_NAME	TEL	TEL
get_account	put_CORP_INFO										
CORP_INFO_CODE	CORP_INFO_CODE										
ID	COMPANY_CODE_SHANON										
NAME	COMPANY_NAME										
TEL	TEL										

コンポーネント名	
put_CORP_INFO	
	
マッピングされたデータセットをデータベースに一括更新する (バルク更新)	
必須設定	
接続先	{接続用コネクション} *{コネクション設定(HULFT Square側)}参照
テーブル名	CORP_INFO
テーブル名を引用符で囲う	指定なし (※Snowflakeで日本語テーブル名を表記する際は引用符で囲う必要がある)
キーの値と一致する行は更新	COMPANY_CODE_SHANON
列一覧	CORP_INFO_CODE ID DATE_REGIST DATE_UPDATE NAME NAME_KA ZIP1 ZIP2 ZIP3 COUNTRY PREFECTURE ADDRESS1 ADDRESS2 ADDRESS3 ADDRESS4 ADDRESS5 TEL SUB_TEL FAX EMAIL_DOMAIN DATE_OF_ESTABLISHMENT URL
書き込み設定	
インサートモード	バッチインサート (※この指定により一括更新を行うバルク処理となる)
バッチサイズ	1000(※デフォルト値:コミット行数単位)
クエリタイムアウト(秒)	指定なし

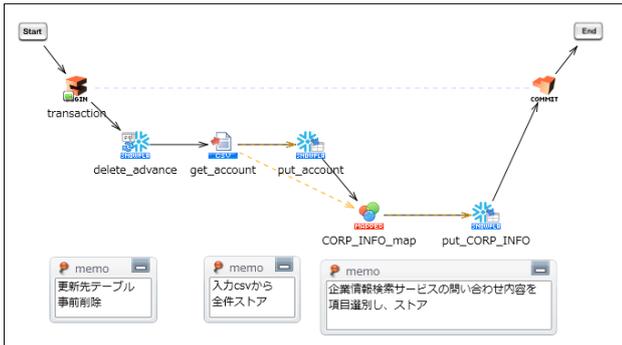
コンポーネント名	
commit	
	
コンポーネント概要	
処理単位開始 通常概念のトランザクションのコミットと同義、COMMIT発生により当トランザクション配下の処理は実更新される	

スクリプト利用手順

1. 「コネクション設定Snowflake側」を設定する
2. 「コネクション設定(HULFT Square側)」を設定する
3. Snowflakeにアクセスするコンポーネントの接続先に「コネクション設定\_HULFT Square側」で設定した接続先を指定する
4. [/{利用するワークスペース}/SHANON]ディレクトリに[SHANON\_COMPANY.csv]を配置する
5. スクリプトを実行する

**Snowflake向けHULFT Squareアプリケーション  
スクリプト設定(from\_Marketo)**

**スクリプト全体図**



**スクリプト動作概要**

Marketoからの企業情報CSVをSnowflakeにストアする

\*以下設定のオレンジ色の設定箇所は、スクリプトをインストールするだけでは設定されないため、スクリプトをインストール後に手動で入力する必要がある

**スクリプト基本設定**

HULFT Square URL	
ワークスペース	
格納プロジェクト名	Snowflake
格納フォルダ名	
スクリプト名	Snowflake_from_Marketo

**外部システムから取り込む値/ファイル**

ファイル [Marketo_COMPANY.csv]	項目構成については該当コンポーネントの列一覧を参照のこと
----------------------------	------------------------------

**スクリプト変数**

_InputDataPathAndName	入力元となるCSVファイルのパスとファイル名(/<ワークスペース名>/Marketo/Marketo_COMPANY.csv)
-----------------------	---

**スクリプト詳細設定**

<b>コンポーネント名</b>	transaction
<b>コンポーネント概要</b>	処理単位開始 通常概念のトランザクションと同義、COMMIT発生により当トランザクション配下の処理は実更新される

<b>コンポーネント名</b>	delete_advance
<b>コンポーネント概要</b>	「更新系SQL実行処理」コンポーネントにより、出力テーブルレコードを事前削除する
<b>必須設定</b>	
接続先	[APIリクエスト用コネクション] *[コネクション設定(HULFT Square側)]参照
SQL文	DELETE FROM MARKETO_COMPANY
スキーマ定義	設定なし

コンポーネント名	
get_account 	
コンポーネント概要	
「CSVファイル読み込み処理」コンポーネントにより、Marketoからの企業情報CSVを読み込む	
ファイル	\$(L_InputDataPathAndFileName)
デリミタ文字	半角カンマ
列一覧	CORP_INFO_CODE BILLINGCOUNTRY BILLINGPOSTALCODE WENSITE COMPANYNOTES INDUSTRY COMPANY ANNUALREVENUE NUMBEROFEMPLOYEES SITE EXTERNALCOMPANTID MAINPHONE BILLINGCITY SICCODE BILLINGSTATE BILLINGSTREET
読み取り設定	
エンコード	UTF-8
読み取り開始行	1
最初の行は値として取得しない	有効

コンポーネント名	
put_account 	
「テーブル書き込み処理」コンポーネントにより、読み込んだCSVデータセットをデータベースに一括更新する（バルク更新）	
必須設定	
接続先	{接続用コネクション} *[コネクション設定(HULFT Square側)]参照
テーブル名	MARKETO_COMPANY
テーブル名を引用符で囲う	指定なし（※Snowflakeで日本語テーブル名を表記する際は引用符で囲う必要がある）
キーの値と一致する行は更新	指定なし（※このチェックによりUPSERT処理が可能となる）
列一覧	上記「get_account」コンポーネント項目と同様
書き込み設定	
インサートモード	バッチインサート（※この指定により一括更新を行うバルク処理となる）
バッチサイズ	1000(※デフォルト値：コミット行数単位)
クエリタイムアウト(秒)	指定なし

コンポーネント名	
CORP_INFO_map 	
「企業情報検索」テーブルに出力するための項目をマッピングする	
必須設定	
マッピング列一覧	get_account put_CORP_INFO
	COMPANYNOTES CORP_INFO_CODE
	EXTERNALCOMPANYID COMPANY_CODE_MARKETO
	COMPANY COMPANY_NAME
	MAINPHONE TEL

コンポーネント名	
put_CORP_INFO	
	
マッピングされたデータセットをデータベースに一括更新する（バルク更新）	
必須設定	
接続先	[接続用コネクション] * [コネクション設定(HULFT Square側)]参照
テーブル名	CORP_INFO
テーブル名を引用符で囲う	指定なし（※Snowflakeで日本語テーブル名を表記する際は引用符で囲う必要がある）
キーの値と一致する行は更新	COMPANY_CODE_MARKETO
列一覧	CORP_INFO_CODE BILLINGCOUNTRY BILLINGPOSTALCODE WENSITE COMPANYNOTES INDUSTRY COMPANY ANNUALREVENUE NUMBEROFEMPLOYEES SITE EXTERNALCOMPANTID MAINPHONE BILLINGCITY SICCODE BILLINGSTATE BILLINGSTREET
書き込み設定	
インサートモード	バッチインサート（※この指定により一括更新を行うバルク処理となる）
バッチサイズ	1000(※デフォルト値：コミット行数単位)
クエリタイムアウト(秒)	指定なし

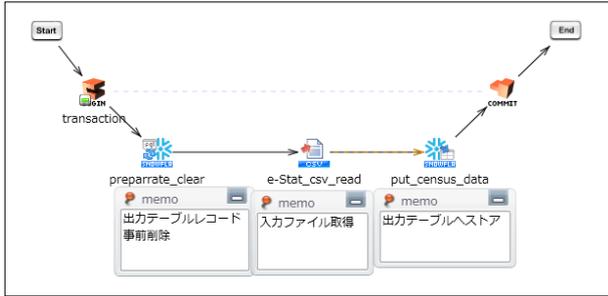
コンポーネント名	
commit	
	
コンポーネント概要	
処理単位開始 通常概念のトランザクションのコミットと同義、COMMIT発生により当トランザクション配下の処理は実更新される	

スクリプト利用手順

1. 「コネクション設定Snowflake側」を設定する
2. 「コネクション設定(HULFT Square側)」を設定する
3. Snowflakeにアクセスするコンポーネントの接続先に「コネクション設定\_HULFT Square側」で設定した接続先を指定する
4. [/利用するワークスペース]/Marketo)ディレクトリに[Marketo\_COMPANY.csv]を配置する
5. スクリプトを実行する

Snowflake向けHULFT Squareアプリケーション  
 スクリプト設定(from\_e-Stat\_0000010103)

スクリプト全体図



スクリプト動作概要

e-Stat政府統計情報からのCSVファイルを読み込み、Snowflakeテーブルへアップロードする

\*以下設定のオレンジ色の設定箇所は、スクリプトをインストールするだけでは設定されないため、スクリプトをインストール後に手動で入力する必要がある

スクリプト基本設定

HULFT Square URL	
ワークスペース	
格納プロジェクト名	Snowflake
格納フォルダ名	
スクリプト名	Snowflake_from_e-Stat_0000010103

外部システムから取り込む値/ファイル

ファイル [e-Stat_CENSUS_0000010103.csv]	項目構成については該当コンポーネントの列一覧を参照のこと
-------------------------------------	------------------------------

スクリプト変数

L_InputDataPathAndFileName	入力元となるCSVファイルのパスとファイル名(/<ワークスペース名>/e-Stat/e-Stat_CENSUS_0000010103.csv)
----------------------------	---

スクリプト詳細設定

コンポーネント名	transaction
コンポーネント概要	処理単位開始 通常概念のトランザクションと同義、COMMIT発生により当トランザクション配下の処理は実更新される

コンポーネント名	preparrate_clear
コンポーネント概要	「更新系SQL実行処理」コンポーネントにより、出力テーブルレコードを事前削除する
必須設定	
接続先	{APIリクエスト用コネクション} *{コネクション設定(HULFT Square側)}参照
SQL文	DELETE FROM ESTAT_STATISTICS
スキーマ定義	設定なし

コンポーネント名	
e-Stat_csv_read  e-Stat_csv_read	
コンポーネント概要	
「CSVファイル読み込み処理」コンポーネントにより、e-Statからの統計情報CSVを読み込む	
ファイル	\$(L_InputDataPathAndFileName)
デリミタ文字	半角カンマ
列一覧	STATS_DATA_ID HEADER_01 HEADER_02 HEADER_03 HEADER_04 HEADER_05 STATS_VAUE
読み取り設定	
エンコード	UTF-8
読み取り開始行	1
最初の行は値として取得しない	有効

コンポーネント名	
put_census_data  put_census_data	
「テーブル書き込み処理」コンポーネントにより、読み込んだCSVデータセットをデータベースに一括更新する（バルク更新）	
必須設定	
接続先	{接続用コネクション} *[コネクション設定(HULFT Square側)]参照
テーブル名	ESTAT_STATISTICS
テーブル名を引用符で囲う	指定なし（※Snowflakeで日本語テーブル名を表記する際は引用符で囲う必要がある）
キーの値と一致する行は更新	指定なし（※このチェックによりUPSERT処理が可能となる）
列一覧	STATS_DATA_ID      VARCHAR HEADER_01            VARCHAR HEADER_02            VARCHAR HEADER_03            VARCHAR HEADER_04            VARCHAR HEADER_05            VARCHAR STATS_VAUE            NUMBER
書き込み設定	
インサートモード	バッチインサート（※この指定により一括更新を行うバルク処理となる）
バッチサイズ	1000(※デフォルト値：コミット行数単位)
クエリタイムアウト(秒)	指定なし

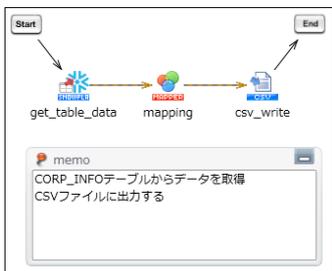
コンポーネント名	
commit  commit	
コンポーネント概要	
処理単位開始 通常概念のトランザクションのコミットと同義、COMMIT発生により当トランザクション配下の処理は実更新される	

スクリプト利用手順

1. 「コネクション設定Snowflake側」を設定する
2. 「コネクション設定(HULFT Square側)」を設定する
3. Snowflakeにアクセスするコンポーネントの接続先に「コネクション設定\_HULFT Square側」で設定した接続先を指定する
4. [/[利用するワークスペース]/e-Stat]ディレクトリに[e-Stat\_CENSUS\_0000010103.csv]を配置する
5. 変数「L\_InputDataPathAndFileName」に上記の入力元のパスとファイル名を設定
6. スクリプトを実行する

Snowflake向けHULFT Squareアプリケーション  
スクリプト設定(to\_CORP\_INFO)

スクリプト全体図



スクリプト動作概要

Snowflakeテーブルから企業情報検索サービスでの検索に利用するデータを取得し、CSVファイルに出力する

\*以下設定のオレンジ色の設定箇所は、スクリプトをインストールするだけでは設定されないため、スクリプトをインストール後に手動で入力する必要がある

スクリプト基本設定

HULFT Square URL	
ワークスペース	
格納プロジェクト名	Snowflake
格納フォルダ名	
スクリプト名	Snowflake_to_CORP_INFO

外部システムから取り込む値/ファイル

なし	
----	--

スクリプト変数

_OutputDataPathAndFileName	出力先CSVファイルのフォルダパス&ファイル名(/<ワークスペース名>/CORP_INFO/CORP_INFO_INQUIRY.csv)
----------------------------	--

スクリプト詳細設定

コンポーネント名			
get_table_data			
コンポーネント概要			
企業情報検索サービス検索用データを取得する			
必須設定			
接続先	[APIリクエスト用コネクション] *[コネクション設定(HULFT Square側)]参照		
テーブル名	CORP_INFO		
テーブル名を引用符で囲う	無効		
スキーマ定義	読み取り対象	列名	型
	有効	COMPANY_CODE_SF	VARCHAR (String)
	有効	COMPANY_CODE_SHANON	VARCHAR (String)
	有効	COMPANY_CODE_Marketo	VARCHAR (String)
	有効	CORP_INFO_CODE	VARCHAR (String)
	有効	COMPANY_NAME	VARCHAR (String)
	有効	TEL	VARCHAR (String)
	有効	SEC_CODE	VARCHAR (String)
	有効	COMPANY_STAT	VARCHAR (String)
	有効	OFFICE_STAT	VARCHAR (String)
	有効	CAPITAL	NUMBER (long)
	有効	FIRSTTERM_AMOUNT	NUMBER (long)
有効	REPRESENTATIVE	VARCHAR (String)	

コンポーネント名		
mapping  mapping		
コンポーネント概要		
データをマッピングする		
HULFT Square mapper エディタ		
入力元 (入力データ)	マッピングツール(値)	出力先 (出力データ)
get_table_data/table/row	単純な繰り返し	csv_write/table/row
get_table_data/table/row/column[1](='COMPANY_CODE_SF')	単純移送	csv_write/table/row/column(='COMPANY_CODE_SF')
get_table_data/table/row/column[2](='COMPANY_CODE_SHANON')	単純移送	csv_write/table/row/column(='COMPANY_CODE_SHANON')
get_table_data/table/row/column[3](='COMPANY_CODE_Marketo')	単純移送	csv_write/table/row/column(='COMPANY_CODE_Marketo')
get_table_data/table/row/column[4](='CORP_INFO_CODE')	単純移送	csv_write/table/row/column(='COMPANY_CORP_INFO_CODE')
get_table_data/table/row/column[5](='COMPANY_NAME')	単純移送	csv_write/table/row/column(='COMPANY_NAME')
get_table_data/table/row/column[6](='TEL')	単純移送	csv_write/table/row/column(='TEL')

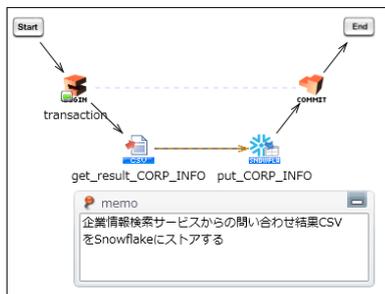
コンポーネント名	
csv_write  csv_write	
コンポーネント概要	
取得した検索用データをCSVファイルに出力する	
必須設定	
ファイル	\${[OutputDataPathAndFileName]}
デリミタ文字	半角カンマ
列一覧	COMPANY_CODE_SF COMPANY_CODE_SHANON COMPANY_CODE_Marketo COMPANY_CORP_INFO_CODE COMPANY_NAME TEL
書き込み設定	
エンコード	UTF-8
改行コード	CR+LF
上書き	有効
追加書き込み	無効
1行目に列名を挿入	有効
ファイルが存在する場合は列名を挿入しない	無効

スクリプト利用手順

1. 「コネクション設定Snowflake側」を設定する
2. 「コネクション設定(HULFT Square側)」を設定する
3. Snowflakeにアクセスするコンポーネントの接続先に「コネクション設定\_HULFT Square側」で設定した接続先を指定する
4. csv\_writeコンポーネントのファイルパスを設定する
5. スクリプトを実行する

Snowflake向けHULFT Squareアプリケーション  
スクリプト設定(from\_CORP\_INFO)

スクリプト全体図



スクリプト動作概要

企業情報検索サービスから取得した企業情報をSnowflakeにストアする

\*以下設定のオレンジ色の設定箇所は、スクリプトをインストールするだけでは設定されないため、スクリプトをインストール後に手動で入力する必要がある

スクリプト基本設定

HULFT Square URL	
ワークスペース	
格納プロジェクト名	Snowflake
格納フォルダ名	
スクリプト名	Snowflake_from_CORP_INFO

外部システムから取り込む値/ファイル

ファイル [CORP_INFO_RESULTS.csv]	項目構成については該当コンポーネントの列一覧を参照のこと
------------------------------	------------------------------

スクリプト変数

_InputDataPathAndFileName	入力元CSVファイルのフォルダパス&ファイル名(/<ワークスペース名>/CORP_INFO/CORP_INFO_RESULTS.csv)
---------------------------	--

スクリプト詳細設定

コンポーネント名	transaction
コンポーネント概要	処理単位開始 通常概念のトランザクションと同義、COMMIT発生により当トランザクション配下の処理は実更新される

コンポーネント名	
get_result_CORP_INFO 	
コンポーネント概要	
「CSVファイル読み込み処理」コンポーネントにより、企業情報検索サービスからの企業情報CSVを読み込む	
ファイル	\$(!_InputDataPathAndFileName)
デリミタ文字	半角カンマ
列一覧	COMPANY_CODE_SF COMPANY_CODE_SHANON COMPANY_CODE_Marketo COMPANY_CORP_INFO_CODE COMPANY_NAME TEL SEC_CODE COMPANY_STAT OFFICE_STAT CAPITAL FIRSTTERM_AMOUNT REPRESENTATIVE
読み取り設定	
エンコード	UTF-8
読み取り開始行	1
最初の行は値として取得しない	有効

コンポーネント名	
put_CORP_INFO 	
「テーブル書き込み処理」コンポーネントにより、読み込んだCSVデータセットをデータベースに一括更新する（バルク更新）	
必須設定	
接続先	[接続用コネクション] *[コネクション設定(HULFT Square側)]参照
テーブル名	CORP_INFO
テーブル名を引用符で囲う	指定なし（※Snowflakeで日本語テーブル名を表記する際は引用符で囲う必要がある）
キーの値と一致する行は更新	COMPANY_CODE_SF、COMPANY_CODE_SHANON、COMPANY_CODE_Marketo
列一覧	COMPANY_CODE_SF COMPANY_CODE_SHANON COMPANY_CODE_Marketo CORP_INFO_CODE COMPANY_NAME TEL SEC_CODE COMPANY_STAT OFFICE_STAT CAPITAL FIRSTTERM_AMOUNT REPRESENTATIVE
書き込み設定	
インサートモード	アップデート・インサート
バッチサイズ	指定なし
クエリタイムアウト(秒)	指定なし

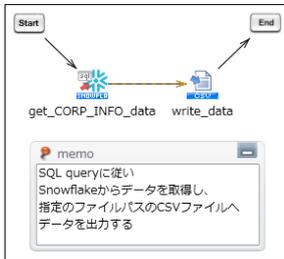
コンポーネント名	
commit 	
コンポーネント概要	
処理単位開始 通常概念のトランザクションのコミットと同義、COMMIT発生により当トランザクション配下の処理は実更新される	

### スクリプト利用手順

1. 「コネクション設定Snowflake側」を設定する
2. 「コネクション設定(HULFT Square側)」を設定する
3. Snowflakeにアクセスするコンポーネントの接続先に「コネクション設定\_HULFT Square側」で設定した接続先を指定する
4. [/利用するワークスペース]/CORP\_INFOディレクトリに[CORP\_INFO\_RESULTS.csv]を配置する
5. スクリプトを実行する

**Snowflake向けHULFT Squareアプリケーション  
スクリプト設定(to\_SFA\_and\_MA)**

**スクリプト全体図**



**スクリプト動作概要**

企業情報検索サービスから得た検索結果をSnowflakeの企業検索テーブルからCSV形式にてダウンロードする

\*以下設定のオレンジ色の設定箇所は、スクリプトをインストールするだけでは設定されないため、スクリプトをインストール後に手動で入力する必要がある

**スクリプト基本設定**

HULFT Square URL	
ワークスペース	
格納プロジェクト名	Snowflake
格納フォルダ名	
スクリプト名	Snowflake_to_SFA_and_MA

**外部システムから取り込む値/ファイル**

なし	
----	--

**スクリプト変数**

_OutputDataPathAndFileName	出力先CSVファイルのフォルダパス & ファイル名(/<ワークスペース名>/CORP_INFO/CORP_INFO.csv)
----------------------------	--

**スクリプト詳細設定**

<b>コンポーネント名</b>	
get_CORP_INFO_data	get_CORP_INFO_data
<b>コンポーネント概要</b>	
「検索系SQL実行処理」コンポーネントにより、入力テーブルレコードを取得する	
<b>必須設定</b>	
接続先	[APIリクエスト用コネクション] *[コネクション設定(HULFT Square側)]参照
SQL文	SELECT * FROM CORP_INFO WHERE CORP_INFO_CODE != 'unset'
入力スキーマ定義	設定なし
出力スキーマ定義	COMPANY_CODE_SF COMPANY_CODE_SHANON COMPANY_CODE_Marketo CORP_INFO_CODE COMPANY_NAME TEL SEC_CODE COMPANY_STAT OFFICE_STAT CAPITAL FIRSTTERM_AMOUNT REPRESENTATIVE

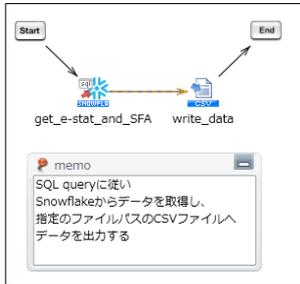
コンポーネント名	
write_data	
	
コンポーネント概要	
「CSVファイル書き込み処理」コンポーネントにより、企業情報検索サービスからの企業付加情報を書き込む	
ファイル	`\${L_OutputDataPathAndFileName}`
デリミタ文字	半角カンマ
列一覧	Salesforce企業コード
	SHANON企業コード
	Marketo企業コード
	企業情報検索サービスの企業コード
	企業名称
	電話番号
	証券コード
	会社状況
	事業所状況
	資本金
	前売上高
代表者名	
読み取り設定	
エンコード	UTF-8
改行コード	CR+LF
上書き	有効
追加書き込み	無効
1行目に列名を挿入	有効
ファイルが存在する場合は列名を挿入しない	無効

### スクリプト利用手順

1. 「コネクション設定Snowflake側」を設定する
2. 「コネクション設定(HULFT Square側)」を設定する
3. Snowflakeにアクセスするコンポーネントの接続先に「コネクション設定\_HULFT Square側」で設定した接続先を指定する
4. 変数「L\_OutputDataPathAndFileName」に出力先のパスとファイル名を設定
5. スクリプトを実行する

Snowflake向けHULFT Squareアプリケーション  
 スクリプト設定(to\_e-Stat\_for\_AOAI)

スクリプト全体図



スクリプト動作概要

e-Stat政府統計情報からのCSVファイルを読み込み、Snowflakeテーブルへアップロードする

\*以下設定のオレンジ色の設定箇所は、スクリプトをインストールするだけでは設定されないため、  
 スクリプトをインストール後に手動で入力する必要がある

スクリプト基本設定

HULFT Square URL	
ワークスペース	
格納プロジェクト名	Snowflake
格納フォルダ名	
スクリプト名	Snowflake_to_e-Stat_for_AOAI

外部システムから取り込む値/ファイル

なし	
----	--

スクリプト変数

_OutputDataPathAndFileName	出力先CSVファイルのフォルダパス & ファイル名(/<ワークスペース名>/e-Stat/e-Stat_0000010103_FOR_AOAI.csv)
----------------------------	--

スクリプト詳細設定

コンポーネント名									
get_e-Stat_and_SFA									
									
コンポーネント概要									
「検索系SQL実行処理」コンポーネントにより、入力テーブルレコードを取得する									
必須設定									
接続先	[APIリクエスト用コネクション] *[コネクション設定(HULFT Square側)]参照								
SQL文	<pre> SELECT BILLING_STATE, ESTAT_TYPES_NAME, SUM(ALL_COMPANIES) ALL_COMPANIES, SUM(MY_CLIENT) MY_CLIENT FROM ( SELECT BILLING_STATE, ESTAT_TYPES_NAME, ESTAT_ALL_COMPANIES, SF.MY_CLIENT FROM ( SELECT SA.INDUSTRY INDUSTRY, SE.ESTAT_TYPES_NAME ESTAT_TYPES_NAME, SABILLING_STATE BILLING_STATE, COUNT(*) AS MY_CLIENT FROM SALESFORCE_ACCOUNT SA INNER JOIN SF.ESTAT_TYPES SE ON SA.INDUSTRY = SE.SF_TYPES WHERE SA.INDUSTRY IS NOT NULL AND SABILLING_STATE IS NOT NULL GROUP BY SA.INDUSTRY, SE.ESTAT_TYPES_NAME, SABILLING_STATE ORDER BY SA.INDUSTRY, SABILLING_STATE ) SF INNER JOIN ( SELECT ES.HEADER_02, EH02.NAME ESTAT_INDUSTRY, ES.HEADER_03, EH03.NAME ESTAT_STATE, ES.STATS_VALUE ALL_COMPANIES FROM ESTAT_STATISTICS ES INNER JOIN ESTAT_HEADER EH03 ON EH03.CODE = ES.HEADER_03 AND EH03.HEADER_SEQ = 3 INNER JOIN ESTAT_HEADER EH02 ON EH02.CODE = ES.HEADER_02 AND EH02.HEADER_SEQ = 2 WHERE HEADER_04 like '2014%' ) ESTAT ON SF.ESTAT_TYPES_NAME = ESTAT.ESTAT_INDUSTRY AND SF.BILLING_STATE = ESTAT.ESTAT_STATE ) WHERE BILLING_STATE IN('茨城','栃木','群馬','埼玉','千葉','東京','神奈川') GROUP BY BILLING_STATE, ESTAT_TYPES_NAME ORDER BY BILLING_STATE ASC, ESTAT_TYPES_NAME ASC </pre>								
入力スキーマ定義	設定なし								
出力スキーマ定義	<table border="1"> <tr> <td>都道府県名称</td> <td>( BILLING_STATE )</td> </tr> <tr> <td>事業所種別</td> <td>( ESTAT_TYPES_NAME )</td> </tr> <tr> <td>事業所数</td> <td>( ESTAT.ALL_COMPANIES )</td> </tr> <tr> <td>自社取引先事業所数</td> <td>( SF.MY_CLIENT )</td> </tr> </table>	都道府県名称	( BILLING_STATE )	事業所種別	( ESTAT_TYPES_NAME )	事業所数	( ESTAT.ALL_COMPANIES )	自社取引先事業所数	( SF.MY_CLIENT )
都道府県名称	( BILLING_STATE )								
事業所種別	( ESTAT_TYPES_NAME )								
事業所数	( ESTAT.ALL_COMPANIES )								
自社取引先事業所数	( SF.MY_CLIENT )								

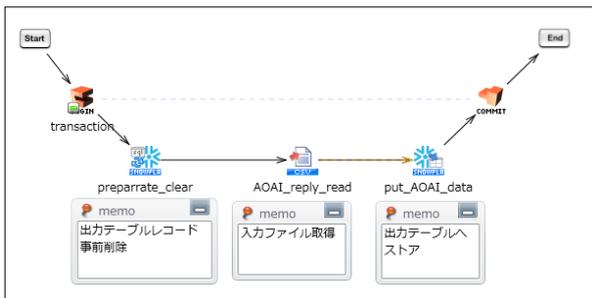
コンポーネント名	
write_data	
 write_data	
コンポーネント概要	
「CSVファイル書き込み処理」コンポーネントにより、e-Statからの統計情報CSVを読み込む	
ファイル	`\${L_OutputDataPathAndFileName}`
デリミタ文字	半角カンマ
列一覧	都道府県名称 ( BILLING_STATE )
	事業所種別 ( ESTAT_TYPES_NAME )
	事業所数 ( ESTAT_ALL_COMPANIES )
	自社取引先事業所数 ( SF.MY_CLIENT )
読み取り設定	
エンコード	UTF-8
改行コード	CR+LF
上書き	yes
追加書き込み	no
1行目に列名を挿入	yes
ファイルが存在する場合は列名を挿入しない	no

### スクリプト利用手順

1. 「コネクション設定(Snowflake側)」を設定する
2. 「コネクション設定(HULFT Square側)」を設定する
3. Snowflakeにアクセスするコンポーネントの接続先に「コネクション設定\_HULFT Square側」で設定した接続先を指定する
4. [/{利用するワークスペース}/e-Stat]ディレクトリに[Snowflake\_to\_e-Stat\_for\_AOAI]を配置する
5. 変数 「L\_OutputDataPathAndFileName」 に上記の入力元のパスとファイル名を設定
6. スクリプトを実行する

**Snowflake向けHULFT Squareアプリケーション  
スクリプト設定(From\_AOAI)**

スクリプト全体図



スクリプト動作概要

AOAIからの返答内容CSVファイルを読み込み、Snowflakeテーブルへアップロードする

\*以下設定のオレンジ色の設定箇所は、スクリプトをインストールするだけでは設定されないため、スクリプトをインストール後に手動で入力する必要がある

スクリプト基本設定

HULFT Square URL	
ワークスペース	
格納プロジェクト名	Snowflake
格納フォルダ名	
スクリプト名	Snowflake_from_AOAI

外部システムから取り込む値/ファイル

ファイル [AOAI_e-Stat_0000010103_RESULTS.csv]	項目構成については各「AOAI_reply_read」コンポーネントの列一覧を参照のこと
---	--

スクリプト変数

_InputDataPathAndFileName	入力元CSVファイルのフォルダパス&ファイル名(/<ワークスペース名>/AOAI/AOAI_e-Stat_0000010103_RESULTS.csv)
---------------------------	--

スクリプト詳細設定

<b>コンポーネント名</b>	
transaction	
<b>コンポーネント概要</b>	
処理単位開始 通常概念のトランザクションと同義、COMMIT発生により当トランザクション配下の処理は実更新される	

<b>コンポーネント名</b>	
preparrate_clear	
<b>コンポーネント概要</b>	
「更新系SQL実行処理」コンポーネントにより、出力テーブルレコードを事前削除する	
<b>必須設定</b>	
接続先	[APIリクエスト用コネクション] * [コネクション設定(HULFT Square側)]参照
SQL文	DELETE FROM AOAI_REPLY
スキーマ定義	設定なし

<b>コンポーネント名</b>	
AOAI_reply_read 	
<b>コンポーネント概要</b>	
「CSVファイル読み込み処理」コンポーネントにより、AOAIからの返答内容CSVを読み込む	
ファイル	\$(L_InputDataPathAndFileName)
デリミタ文字	半角カンマ
列一覧	QUESTION REPLY_ITEM ALERT
<b>読み取り設定</b>	
エンコード	UTF-8
読み取り開始行	1
最初の行は値として取得しない	有効

<b>コンポーネント名</b>	
put_AOAI_data 	
「テーブル書き込み処理」コンポーネントにより、読み込んだCSVデータセットをデータベースに一括更新する（バルク更新）	
<b>必須設定</b>	
接続先	{接続用コネクション} *{コネクション設定(HULFT Square側)}参照
テーブル名	AOAI_REPLY
テーブル名を引用符で囲う	指定なし（※Snowflakeで日本語テーブル名を表記する際は引用符で囲う必要がある）
キーの値と一致する行は更新	指定なし（※このチェックによりUPSERT処理が可能となる）
列一覧	QUESTION      VARCHAR REPLY_ITEM      VARCHAR ALERT              VARCHAR
<b>書き込み設定</b>	
インサートモード	バッチインサート（※この指定により一括更新を行うバルク処理となる）
バッチサイズ	1000(※デフォルト値：コミット行数単位)
クエリタイムアウト(秒)	指定なし

<b>コンポーネント名</b>	
commit 	
<b>コンポーネント概要</b>	
処理単位開始 通常概念のトランザクションのコミットと同義、COMMIT発生により当トランザクション配下の処理は実更新される	

**スクリプト利用手順**

1. 「コネクション設定Snowflake側」を設定する
2. 「コネクション設定(HULFT Square側)」を設定する
3. Snowflakeにアクセスするコンポーネントの接続先に「コネクション設定\_HULFT Square側」で設定した接続先を指定する
4. [/{利用するワークスペース}/AOAI]ディレクトリに[AOAI\_e-Stat\_0000010103\_RESULTS.csv]を配置する
5. 変数「L\_InputDataPathAndFileName」に上記の入力元のパスとファイル名を設定
6. スクリプトを実行する