

**FACTSET** › SEE THE ADVANTAGE

# Snowflake Native App: PA Engine API - BETA Developer's Manual and Reference May 2024

Snowflake Native Application – Document Version 2.1



**Contents**

- 1. Overview ..... 3
- 2. Pre-requisites ..... 3
  - 2.1 Generate API key ..... 3
- 3. Steps for Installation..... 4
  - 3.1 Locate the Marketplace listing ..... 4
  - 3.2 Install app ..... 4
  - 3.3 Open app..... 4
  - 3.4 Run setup script..... 4
  - 3.5 Check installation ..... 5
- 4. PA Calculation Stored Procedures ..... 6
- 5. Lookups ..... 8
  - 5.1 Components Lookup ..... 8
  - 5.2 Components Settings Lookup..... 9
  - 5.3 Groups Lookup ..... 10
  - 5.4 Currencies Lookup ..... 11
  - 5.5 Accounts Lookup ..... 12
  - 5.6 Frequencies Lookup ..... 13
  - 5.7 Dates Lookup ..... 13
- 6. Calculation tables ..... 14
  - 6.1 Calculation Helper Tables ..... 14
- 7. Cleanup Job ..... 19
- 8. Time zone/ Region ..... 20
- 9. Known Issues ..... 20
- 10. Limitations & Resources ..... 21

## 1. Overview

The Snowflake Native App leverages FactSet’s Analytics API to request analytics with FactSet’s best in class Analytics engines. The app leverages Snowflake’s Native App framework to make API calls directly from client’s Snowflake account, which will be termed as the consumer account in this document. This app encapsulates the flexibility of requesting analytics through APIs and consuming the responses directly into the consumer account seamlessly as tabular formats.

## 2. Pre-requisites

To get full functionality of the Snowflake Native App, the following prerequisite steps, focused on authentication, needs to be completed. If this is not completed, only demo functionality would be accessible right using the app. If you would like to only use the demo functionality, in step 3.4 please install “DEMO SETUP DETAILS” instead of “APP SETUP DETAILS”.

**DESCRIPTION:**

After installing the PA Native App (BETA) from Snowflake Market Place, users will have access to three schemas API\_SCHEMA, APP\_SCHEMA and DEMO\_SCHEMA. APP\_SCHEMA allows FactSet users subscribed to FactSet's PA Engine API to set up the network rule and authenticate, set up external access integration and access the live app stored procedure implementations. API\_SCHEMA allows users to access the live app tables. For users not subscribed to PA Engine API, we have exposed DEMO\_SCHEMA that provides access to demo tables and stored procedures mimicking the request/response behavior associated with API and APP Schema. DEMO\_SCHEMA allows unsubscribed users to play around with PA Native App functionalities like stored procs and tables without explicitly needing a PA API access and understand the app behavior/workflows with mocked up data. A guide on how to set up and use DEMO\_SCHEMA and APP\_SCHEMA successfully is listed below. It is recommended to use "ACCOUNTADMIN" as role and keep the warehouse up and running.

**DEMO SETUP DETAILS:**

```
STEP 1 - This step will download the demo tables and stored procedures.

# QUERY:
RUN "CALL DEMO_SCHEMA.SETUPV2_DEMO();"

// Pre Populated Tables List:
SELECT *FROM DEMO_SCHEMA.DM_ACCOUNTS;
SELECT *FROM DEMO_SCHEMA.DM_CURRENCIES;
SELECT *FROM DEMO_SCHEMA.DM_PAFREQUENCIES;
SELECT *FROM DEMO_SCHEMA.DM_PAGROUPS;
```

### 2.1 Generate API key

- Please navigate to <https://developer.factset.com/learn/authentication-api-key> to generate an API Key.
- Once created, this will be needed in the setup script which is detailed below.
- Before making Snowflake calls, please ensure this API Key works directly with the API, i.e. you do not see a 403 and your account team has set up the appropriate access.

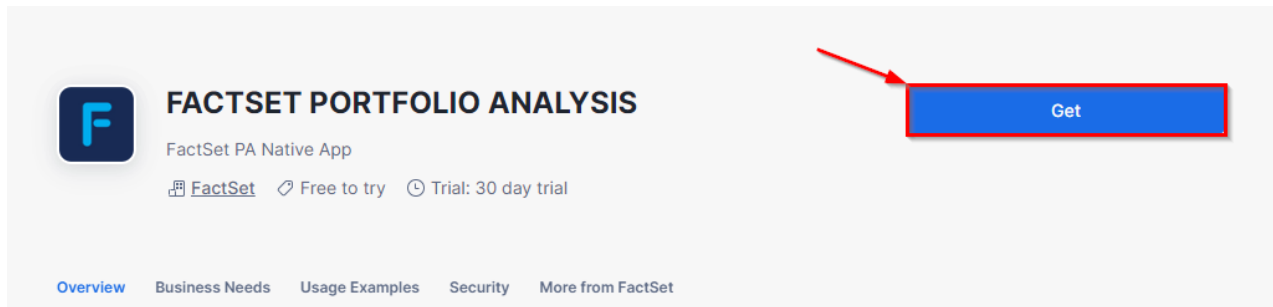
### 3. Steps for Installation

#### 3.1 Locate the Marketplace listing

<https://app.snowflake.com/marketplace/listing/GZT0ZGCQ5231/factset-factset-portfolio-analysis>

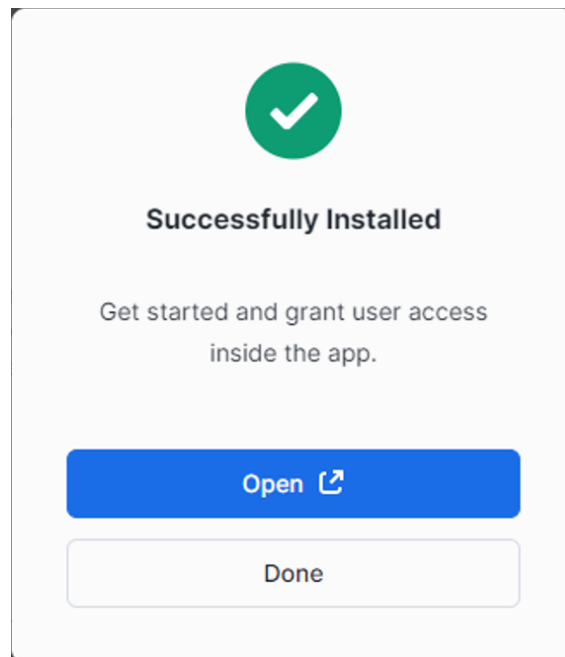
#### 3.2 Install app

Click “Get” (please note that a warehouse needs to be running)



#### 3.3 Open app

After selecting your Snowflake account and logging in, you should see the below:



#### 3.4 Run setup script

After opening the app, and ensuring an appropriate role is selected, the setup queries shown under “APP SETUP DETAILS” will need to be run:

← Apps FACTSET PORTFOLIO ANALYSIS Manage Access

Privileges Telemetry About the app

**APP SETUP DETAILS:**

```
STEP 1 - CREATE A NETWORK RULE WITH A SUITABLE NAME.
# QUERY:
CREATE OR REPLACE NETWORK RULE <APP_SCHEMA.$REPLACE_NETWORK_RULE>
  MODE = EGRESS
  TYPE = HOST_PORT
  VALUE_LIST = ('api.factset.com');

STEP 2 - CREATE BASIC AUTH TYPE SECRET WITH A SUITABLE NAME.
# QUERY:
CREATE OR REPLACE SECRET <APP_SCHEMA.$REPLACE_SECRET>
```

### 3.5 Check installation

Further to the SQL responses which indicate successful setup, you can also confirm the queries have been run successfully by checking Procedures under the APP\_SCHEMA in the FACTSET\_PORTFOLIO\_ANALYSIS database:

snowflake

+ Create

Search

Projects

Data

**Databases**

Add Data

Data Products

AI & ML

Monitoring

Admin

DP

Search

FACTSET\_PORTFOLIO\_ANALYSIS

- API\_SCHEMA
- APP\_SCHEMA
  - Procedures
    - DROP\_TABLE(VARCHAR)
    - GETACCOUNTS(VARCHAR)
    - GETCURRENCIES()
    - GETPACOMPONENTBYID(VARCHAR)
    - GETPACOMPONENTS(VARCHAR)
    - GETPADATES(VARCHAR, VARCHAR, VARCHAR, VARCH...
    - GETPAFREQUENCIES()
    - GETPAGROUPS()
    - PACALCULATION(VARCHAR, VARIANT, VARIANT, VARI...
    - PROCESSRESULT(VARIANT, VARCHAR)
    - SETUPV2(VARCHAR, VARCHAR)
- DEMO\_SCHEMA
- INFORMATION\_SCHEMA

## 4. PA Calculation Stored Procedures

Once the app is successfully installed, clients can use the app to run stored procs that will make API calls to FactSet's PA Engine:

- **PACalculation** – This stored procedure makes PA calculation requests by passing the required parameters and calling the [PA Engine API](#). This retrieves the calculation result and populates it in a new table.

```
call PACalculation('801B800245E468A52AEBEC4BE31CFF5AF82F371DAEF5F158AC2E98C2FA324B46',
  ARRAY_CONSTRUCT(object_construct('holdingsmode', 'b&h', 'Id', 'BENCH:SP50')),
  ARRAY_CONSTRUCT(object_construct('holdingsmode', 'b&h', 'Id', 'BENCH:R.1000')),
  object_construct('startdate', '20230101', 'enddate', '20230102', 'frequency', 'Daily'),
  'USD', 'SECURITIES', 'max-stale=0');
```

- **1<sup>st</sup> Parameter: *Component Id*** - Please use the GET call for component lookup endpoint to get a list of your component IDs that can be used to plug into *PACalculation* stored procedure.

**Example-** "801B800245E468A52AEBEC4BE31CFF5AF82F371DAEF5F158AC2E98C2FA324B46"

- **2<sup>nd</sup> Parameter: *Accounts*** - Please make a call to GET accounts lookup to get a list of your accounts that can be used to plug into *PACalculation* stored procedure.

Single account example

```
ARRAY_CONSTRUCT(object_construct('holdingsmode', 'B&H', 'Id', 'CLIENT:MY_ACCT.ACCT'))
```

Multiple accounts example

```
ARRAY_CONSTRUCT(object_construct('holdingsmode', 'B&H', 'Id', 'CLIENT:MY_ACCT.ACCT'),
  object_construct('holdingsmode', 'B&H', 'Id', 'CLIENT:MY_OTHER_ACCT.ACCT'))
```

- **3<sup>rd</sup> Parameter: *Benchmarks***

Single benchmark example

```
ARRAY_CONSTRUCT(object_construct('holdingsmode', 'B&H', 'Id', 'BENCH:SP50'))
```

Multiple benchmark example

```
ARRAY_CONSTRUCT(object_construct('holdingsmode', 'B&H', 'Id', 'BENCH:SP50'),
  object_construct('holdingsmode', 'B&H', 'Id', 'BENCH:DJII'))
```

- **4<sup>th</sup> Parameter: *Dates & frequency*** - Please make a call to GET frequencies lookup to get a list of frequencies that can be used to plug into *PACalculation* stored procedure. Use the dates endpoint to convert PA relative dates (for example: **0D**, **1M** etc..) to absolute date formats.

Example:

```
object_construct('enddate','0','frequency','Single','startdate','-2M')
```

- **5<sup>th</sup> Parameter: Currency ISO Code**

Example

```
'USD'
```

- **6<sup>th</sup> Parameter: Component Detail**

Example

```
'SECURITIES'
```

- **7<sup>th</sup> Parameter: Cache Control**

Example:

'max-stale=720' - fetches a cached response if available which is not older than 720 secs

'max-stale=0' - fetches a fresh calculation

**Multiport Calculation:**

```
call PACalculation('801B800245E468A52AEBEC4BE31CFF5AF82F371DAEF5F158AC2E98C2FA324B46',
ARRAY_CONSTRUCT(object_construct('holdingsmode','B&H','Id','CLIENT:MY_ACCT.ACCT'),
object_construct('holdingsmode','B&H','Id','CLIENT:MY_OTHER_ACCT.ACCT')),
ARRAY_CONSTRUCT(object_construct('holdingsmode','B&H','Id','BENCH:R.1000'),
object_construct('holdingsmode','B&H','Id','BENCH:R.2000')),
object_construct('startdate','20230101','enddate','20230102','frequency','Daily'),
'USD','SECURITIES','max-stale=0');
```

The calculation request parameters within the stored proc are outlined below:

Parameter name	Data type	Required	Description	Format
<b>componentid</b>	String	Yes	PA component identifier to analyze.	Component ID can be fetched using the <a href="#">Components Lookup</a> .
<b>accounts</b>	Array	Yes	List of user's FactSet account paths OR benchmarks	Account identifiers must have .ACCT or .ACTM extension or BENCH: prefix. Holdings mode can be optionally set for every account. Possible values for holdings mode are B&H (Buy and Hold), <a href="#">TBR (Transaction based returns)</a> , <a href="#">OMS (Order Management System)</a> , <a href="#">VLT (Vaulted returns)</a> or <a href="#">EXT (External Returns Data)</a> . Default holdings mode value is B&H.

Please refer [OA 21344](#) for creating accounts that can be uploaded through nightly parsers or <https://developer.factset.com/apicatalog/model-portfolio-api> for creating prospective accounts with model holdings.

<b>benchmarks</b>	Array	Yes	List of user's FactSet account paths OR benchmarks to compare against.	Benchmark identifiers must have .ACCT or .ACTM extension or BENCH: prefix. Holdings mode can be optionally set for every benchmark. Possible values for holdings mode are B&H (Buy and Hold), TBR (Transaction based returns) or OMS. Default holdings mode value is B&H.
<b>dates</b>	Object	Yes	Date range(s) to run against. Object takes startdate, enddate and frequency.	<a href="#">Supported Date Formats</a> and <a href="#">Frequency Lookup</a> .
<b>currencyisocode</b>	String	Yes	Currency ISO code for calculation.	E.g., USD. Currency ISO codes can be fetched using the <a href="#">Currency Lookup</a> .
<b>componentdetail</b>	String	Yes	Degree of granularity to return in the PA component	Valid values are 'groups', 'groupsall' 'totals', and 'security'. Default value is 'security'. While the groupsall returns all the group levels in the PA report, when componentdetail is set to groups it only returns the no of levels expanded/collapsed from the PA report.
<b>max-stale</b>	String	Yes		E.g. max-stale=0

## 5. Lookups

### 5.1 Components Lookup

#### Description

This endpoint returns a list of components within a PA document. Running calculations expects component identifiers returned by this endpoint. Component identifiers are of variable-length and may change. Users should always use this API before calling Run Multiple Calculations.

#### Table Schema

The table schema for components is outlined below:

Parameter name	Data type	Null	Primary Key	Unique Key	Example
<b>DOCUMENT_NAME</b>	VARCHAR (16777216)	No	Yes	No	pa3_documents:default



<b>COMPONENT_ID</b>	VARCHAR (1000)	No	Yes	No	801B800245E468A52AEBEC4BE31CFF5A F82F371DAEF5F158AC2E98C2FA324B46
<b>COMPONENT_NAME</b>	VARCHAR (16777216)	Yes	No	No	Weights
<b>CATEGORY</b>	VARCHAR (16777216)	Yes	No	No	Weights/Exposures
<b>CREATED_DATE</b>	VARCHAR (500)	Yes	No	No	2023-11-15 05:53:32.626 -0800

**Example**

**Query:**

```
CALL GETPACOMPONENTS('PA3_DOCUMENTS:DEFAULT');
```

**Response table:**

Table: AAPI\_V2\_PROD\_BETA.API\_SCHEMA.PACOMPONENTS Data Details

Filter result... Columns ▾

Row	DOCUMENT_NAME	COMPONENT_ID	COMPONENT_NAME	CATEGORY	CREATED_DATE
1	pa3_documents:default	801B800245E468A52AEB...	Weights	Weights / Exposures	2023-11-17 05:53:32.626 -...
2	pa3_documents:default	3C9AA796BF1E7C61A0A6...	Weights Difference	Weights / Exposures	2023-11-17 05:53:32.626 -...
3	pa3_documents:default	67077F67610EC30675E24...	Summary Characteristics	Characteristics / Characte...	2023-11-17 05:53:32.626 -...
4	pa3_documents:default	32E9F0A74F27A445304C...	Detail Characteristics	Characteristics / Characte...	2023-11-17 05:53:32.626 -...
5	pa3_documents:default	EFAA63FC81F51C5001500...	Characteristics Over Time	Characteristics / Characte...	2023-11-17 05:53:32.626 -...
6	pa3_documents:default	8ED5BDE018BF0F497EB6...	Top Down Attribution	Attribution / Top Down Attr...	2023-11-17 05:53:32.626 -...

5.2 Components Settings Lookup

**Description**

This endpoint returns the default settings of a PA component.

**Table Schema**

The table schema for components settings is outlined below:

Parameter name	Data type	Null	Primary Key	Unique Key	Example
<b>COMPONENT_ID</b>	VARCHAR (1000)	No	Yes	No	801B800245E468A52AEBEC4BE31CFF5A F82F371DAEF5F158AC2E98C2FA324B46

<b>ACCOUNTS</b>	VARCHAR (1000)	Yes	No	No	
<b>BENCHMARKS</b>	VARCHAR (500)	Yes	No	No	
<b>CURRENCYISOCODE</b>	VARCHAR (100)	Yes	No	No	USD
<b>STARTDATE</b>	VARCHAR (500)	Yes	No	No	
<b>ENDDATE</b>	VARCHAR (500)	Yes	No	No	0
<b>FREQUENCY</b>	VARCHAR (500)	Yes	No	No	Single
<b>SNAPSHOT</b>	VARCHAR (100)	Yes	No	No	true
<b>PATH</b>	VARCHAR (16777216)	Yes	No	No	pa3_documents:/default;report0;tile0
<b>COMPONENT_NAME</b>	VARCHAR (16777216)	Yes	No	No	Weights
<b>CATEGORY</b>	VARCHAR (16777216)	Yes	No	No	Weights / Exposures
<b>CREATED_DATE</b>	VARCHAR (500)	Yes	No	No	2023-11-17 02:42:27.180 -0800

**Example**

**Query:**

```
CALL
GETPACOMPONENTBYID('801B800245E468A52AEBEC4BE31CFF5AF82F371DAEF5F158AC2E98C2FA324B46');
```

**Response table:**

	COMPONENT_ID	ACCOUNTS	BENCHMARKS	CURRENCYISOCODE	STARTDATE	ENDDATE	FREQUENCY
1	026F941C86EDF715DB98F3E673A9BF9754FCF77DC831C51BDC366E8619E8E	[]	[]	USD		0M	Single

5.3 Groups Lookup

**Description**

This endpoint returns a list of FactSet-defined and user-defined groups. Group identifiers from this endpoint can be used in Run Calculation API.

Group identifiers are of variable-length and are expected to change.

**Table Schema**

The table schema for groups is outlined below:

Parameter name	Data type	Null	Primary Key	Unique Key	Example
<b>GROUP_ID</b>	VARCHAR (1000)	No	Yes	No	A29916A1E43511BBC10625F8D30DDCF3BFD3659D91DAC367D3CE118BC4AED339
<b>GROUP_NAME</b>	VARCHAR (16777216)	Yes	No	No	P/E for Exclu-Levels
<b>DIRECTORY</b>	VARCHAR (16777216)	Yes	No	No	Client
<b>CATEGORY</b>	VARCHAR (16777216)	Yes	No	No	Grp -Test_Security

**Example**

**Query:**

```
CALL GETPAGROUPS();
```

**Response table:**

Row	GROUP_ID	GROUP_NAME	DIRECTORY	CATEGORY
1	05F1339C04FBC9DABA06238E30B5CAF058D7D1B44502DA3C...	Lot-grp-test-da116d9a	Client	
2	A29916A1E43511BBC10625F8D30DDCF3BFD3659D91DAC367D...	grouping-2-fdf327ff	Client	
3	D9F07D23BDCD355072A4CAEE88AAE6F8558B88CB0B1D977E...	test_fractile_grouping-97c9154f	Client	
4	E2A8C7E74D1982908B364937F07EDAB932EC966D955779A49...	test_fractile_grouping-044e40de	Client	

5.4 Currencies Lookup

**Description**

This endpoint returns a list of supported currencies.

**Table Schema**

The table schema for currencies is outlined below:

Parameter name	Data type	Null	Primary Key	Unique Key	Example
<b>ISO_CODE</b>	VARCHAR (500)	No	Yes	No	EUR

CURRENCY\_NAME      VARCHAR (500)      Yes      No      No      Euro

**Example**

**Query:**

```
CALL GETCURRENCIES();
```

**Response:**

Row	ISO_CODE	CURRENCY_NAME
1	123	European Composite Unit
2	AFN	Afghanistan Afghani
3	ALL	Albanian Lek
4	DZD	Algerian Dinar

5.5 Accounts Lookup

**Description**

This endpoint looks up all ACCT and ACTM files and sub-directories in the given directory.

**Table Schema**

The table schema for accounts is outlined below:

Parameter name	Data type	Null	Primary Key	Unique Key	Example
PATH	VARCHAR (16777216)	Yes	No	No	client:
ACCOUNT_NAME	VARCHAR (16777216)	Yes	No	No	demo_account.acct
CREATED_DATE	VARCHAR (500)	Yes	No	No	2023-11-17 06:31:47.607-0800

**Example**

**Query:**

```
CALL GETACCOUNTS ();
```

**Response table:**

Row	PATH	ACCOUNT_NAME	CREATED_DATE
48	Client:	2103-ACTM-2.ACTM	2023-11-02 02:34:55.776 -0700
49	Client:	2103-C.ACCT	2023-11-02 02:34:55.776 -0700
50	Client:	2103-COPY.ACCT	2023-11-02 02:34:55.776 -0700
51	Client:	2103-TEST.ACTM	2023-11-02 02:34:55.776 -0700

## 5.6 Frequencies Lookup

### Description

This endpoint returns a list of supported frequencies.

### Table Schema

The table schema for frequencies is outlined below:

Parameter name	Data type	Null	Primary Key	Unique Key	Example
FREQUENCY_ID	VARCHAR (1000)	No	Yes	No	Single
FREQUENCY_NAME	VARCHAR (16777216)	Yes	No	No	Single

### Example

**Query:**

```
CALL GETPAFREQUENCIES ();
```

**Response table:**

Row	FREQUENCY_ID	FREQUENCY_NAME
1	Single	Single
2	FiscalYearly	Fiscal Yearly
3	Annually	Annually

## 5.7 Dates Lookup

### Description

This endpoint converts the given start and end dates in FactSet date format to yyyyymmdd format for a PA calculation. For more information on FactSet date format, please refer to [supported dates format](#) section.

### Table Schema

The table schema for frequencies is outlined below:

Parameter name	Data type	Null	Primary Key	Unique Key	Example
STARTDATE	VARCHAR (1000)	Yes	No	No	20240101
ENDDATE	VARCHAR (1000)	Yes	No	No	20231201
CREATED_DATE	VARCHAR (500)	Yes	No	No	2024-03-19 06:25:54.549 -0700

### Example

**Query:**

```
CALL GETPADATES
('1Y', '0', '801B800245E468A52AEBEC4BE31CFF5AF82F371DAEF5F158AC2E98C2FA324B46', '
demo_account.acct');
```

**Response table:**

Row	STARTDATE	ENDDATE	CREATED_DATE
1	20220318	20240319	2024-03-19 06:53:34.787 -0700

## 6. Calculation tables

On the successful completion of a PA calculation stored procedure, the results are stored into a table on the fly and the table name is provided as an output. Depending on the calculation requested, the calculations results table, which are created on the fly, are dynamic.

Example - API\_SCHEMA.CALC\_a0e7fd6dcfe543d48d8537a5d0eb8da9\_16112023\_233538

The app also creates “helper” tables like CALCULATION\_HISTORY, CALCULATION\_RESPONSE\_METAINFO and CALCULATION\_STATUS. These tables allow users to manage the calculation result tables.

For example, if the user is requesting the same calculation multiple times in day, that would result in multiple tables with same calculation id but different timestamps. Users can use the CALCULATION\_STATUS and CALCULATION\_HISTORY table to identify duplicate tables and retain only valid or latest tables.

### 6.1 Calculation Helper Tables

#### Calculation History

##### Description

The CALCULATION\_HISTORY table stores the details for every calculation that is triggered on Snowflake.

**Table Schema**

The table schema for calculation history is outlined below:

Parameter name	Data type	Null	Primary Key	Unique Key
CALC_ID	VARCHAR (1000)	No	Yes	No
SOURCE_NAME	VARCHAR (1000)	No	Yes	No
CREATED_DATE	TIMESTAMP_NTZ(9)	Yes	No	No
UPDATED_DATE	TIMESTAMP_NTZ(9)	Yes	No	No

**Calculation Response MetaInfo**

**Description**

The CALCULATION\_RESPONSE\_METAINFO table stores the metadata for every calculation.

**Table Schema**

The table schema for calculation response meta info is outlined below:

Parameter name	Data type	Null	Primary Key	Unique Key
CALC_ID	VARCHAR (1000)	No	No	Yes
ABSOLUTE_END_DATE	TIMESTAMP_NTZ (9)	Yes	No	No
ABSOLUTE_START_DATE	TIMESTAMP_NTZ (9)	Yes	No	No
ACCOUNTS	VARCHAR (16777216)	Yes	No	No
ADDITIONAL_PRICING_OPTIONS	VARCHAR (16777216)	Yes	No	No
ASSET_LEVEL	VARCHAR (16777216)	Yes	No	No
BENCHMARK	VARCHAR (16777216)	Yes	No	No
BENCHMARK_ANALYTICS_SOURCES	VARCHAR (16777216)	Yes	No	No
BENCHMARK_EXCHANGE_RATE_SOURCES	VARCHAR (16777216)	Yes	No	No
BENCHMARK_HOLDINGS_AS_OF_DATE	VARCHAR (16777216)	Yes	No	No

BENCHMARK_PRICING_SOURCES	VARCHAR (16777216)	Yes	No	No
BENCHMARK_RETURN	VARCHAR (16777216)	Yes	No	No
CALENDAR	VARCHAR (16777216)	Yes	No	No
COMPONENT_ID	VARCHAR (16777216)	Yes	No	No
COMPONENT_NAME	VARCHAR (16777216)	Yes	No	No
COMPOSITE_ASSETS	VARCHAR (16777216)	Yes	No	No
CURRENCY_ISO_CODE	VARCHAR (16777216)	Yes	No	No
DOCUMENT_NAME	VARCHAR (16777216)	Yes	No	No
ESTIMATES_DB_SOURCES	VARCHAR (16777216)	Yes	No	No
EXCLUDE_CASH	VARCHAR (16777216)	Yes	No	No
EXCLUDED_SECURITIES	VARCHAR (16777216)	Yes	No	No
EXCLUSIONS_NAME	VARCHAR(16777216)	Yes	No	No
EXPAND_COMPOSITE_ASSETS	VARCHAR(16777216)	Yes	No	No
FUNDAMENTAL_DB_SOURCES	VARCHAR(16777216)	Yes	No	No
GROUPING_FREQUENCY	VARCHAR(16777216)	Yes	No	No
HIDDEN_SECURITIES_OR_GROUPS	VARCHAR(16777216)	Yes	No	No
MARKET_PORTFOLIO	VARCHAR(16777216)	Yes	No	No
PORTFOLIO_ANALYTICS_SOURCES	VARCHAR(16777216)	Yes	No	No
PORTFOLIO_EXCHANGE_RATE_SOURCES	VARCHAR(16777216)	Yes	No	No
PORTFOLIO_HOLDINGS_AS_OF_DATE	VARCHAR(16777216)	Yes	No	No
PORTFOLIO_PRICING_SOURCES	VARCHAR(16777216)	Yes	No	No
PORTFOLIO_RETURN	VARCHAR(16777216)	Yes	No	No
REPORT_FREQUENCY	VARCHAR(16777216)	Yes	No	No
RISK_MODELS	VARCHAR(16777216)	Yes	No	No
RISK_MODEL_DATE	VARCHAR(16777216)	Yes	No	No
CREATED_DATE	TIMESTAMP_NTZ(9)	Yes	No	No
UPDATED_DATE	TIMESTAMP_NTZ(9)	Yes	No	No

## Calculation Status



**Description**

The CALCULATION\_STATUS table stores the latest calculation tables.

**Table Schema**

The table schema for calculation status is outlined below:

Parameter name	Data type	Null	Primary Key	Unique Key
CALC_ID	VARCHAR (1000)	No	Yes	No
SOURCE_NAME	VARCHAR (1000)	Yes	No	No
MAX_STALE	NUMBER(38,0)	Yes	No	No
REQUEST_PARAMETERS	VARCHAR(16777216)	Yes	No	No
CREATED_DATE	TIMESTAMP_NTZ(9)	Yes	No	No
UPDATED_DATE	TIMESTAMP_NTZ(9)	Yes	No	No

**Caching Remarks**

- All results will be cached for 12 hours by default. This means that by default (without sending any cache-control headers), after the first request successfully completes, all subsequent requests with unchanged request parameters will return the same results for 12 hours.
- Setting the max-stale=<staleness limit in seconds> allows you to fetch pre-calculated results with any subsequent POST requests, as long as they were last calculated within the staleness limit.
  - Once set, the API will check to see if the stored results are within the staleness limit.
    - If they are within the limit, then the request will return the cached results, allowing for a quick response and eliminating unnecessary points accrual.
    - If the results are not within the limit, a brand-new calculation request will be triggered to get the latest results.
- To immediately request the latest results, override the cache by setting “max-stale=0” in the Cache-control parameter.
- Changes made to the underlying PA document via the workstation will not trigger a results re-calculation. The cache-control parameter should be set to “max-stale=0” to immediately request the latest results.
- The max cache-control value is “max-stale=43200” (12 hours).
- The cached result period is applied to each individual calculation unit after it completes. It is not applied to the entire calculation.

**Supported Date Formats**

**Absolute Date Formats**

Format	Description	Examples
MM/DD/YYYY	A day.	7/11/2016, 1/1/2015, 3/6/2014.
MM/YYYY	A month end.	6/2016, 4/2015, 11/2014, 5/2013.
YYYY/MMF	A fiscal quarter end.	2016/3F, 2015/2F.
YYYY/MMC	A calendar quarter end.	2016/1C, 2015/3C.
YYYY	A fiscal year end	2016, 2015.

### Relative Date Formats

Format	Examples
<b>D</b>	0D is the most recently completed trading day, -1D is one trading day prior.
<b>AW</b>	0AW is the most recently completed trading day, -1AW is the one actual week (7 days) prior to the most recently completed trading day.
<b>W</b>	0W is the last day of the most recently completed trading week (usually Friday), -1W is the last trading day of the prior week.
<b>AM</b>	0AM is the most recently completed trading day, -1AM is the same day, one actual month prior.
<b>M</b>	0M is the last trading day of the most recent month, -1M is the last trading day of the prior month.
<b>Q</b>	0Q is the last trading day of the company's most recent fiscal quarter, -1Q is the last day of the prior fiscal quarter.
<b>CQ</b>	0CQ is the last trading day of the most recent calendar quarter (March, June, September, or December), -1CQ is the last trading day of the prior calendar quarter.
<b>AY</b>	0AY is the most recently completed trading day, -1AY is one actual year (365 days) prior.
<b>Y</b>	0Y is the last trading day of the company's most recent fiscal year, -1Y is the last trading day of the prior fiscal year.
<b>CY</b>	0CY is the last trading day of the most recent calendar year (the last trading day in December), 1CY is the last trading day of the prior calendar year.
<b>M/D/Y</b>	M, D, and Y can be zero, negative, or a positive number. Examples: 0/0/-1 is the day one year ago, 0/-7/0 is the day one week ago, -1/0/0 is the day one month ago, 12/31/-1 is the day at the end of last year.

### Special FactSet Date Identifiers

The user needs to set the following properties in their account (.ACCT/.ACTM file) if they wish to use these identifiers:

Identifier	Definition
<b>INCEPTION</b>	Date the account was created. This typically corresponds to the first day of data in the holdings portfolio.

<b>ALTERINCEPTION</b>	Alternative date option to define when the portfolio was created. This is helpful in cases where the inception date isn't the official day the portfolio was created.
<b>TERMINATION</b>	Date the account is terminated.
<b>END_OF_LAST_FISCAL</b>	End of last fiscal year.

## 7. Cleanup Job

The cleanup job aids in the effective management of a database by removing calculation tables that have surpassed a certain number of days. The number of days is represented as the value 'n' in the query. This procedure has been established to ensure optimal storage space, enhance performance, and minimize clutter within the system. After executing this function, only the most current and pertinent data are retained in the system, while any redundant or outdated calculation tables are discarded after 'n' days. If a user inputs '0' for 'n', they will receive a message stating that the number of days must be greater than zero.

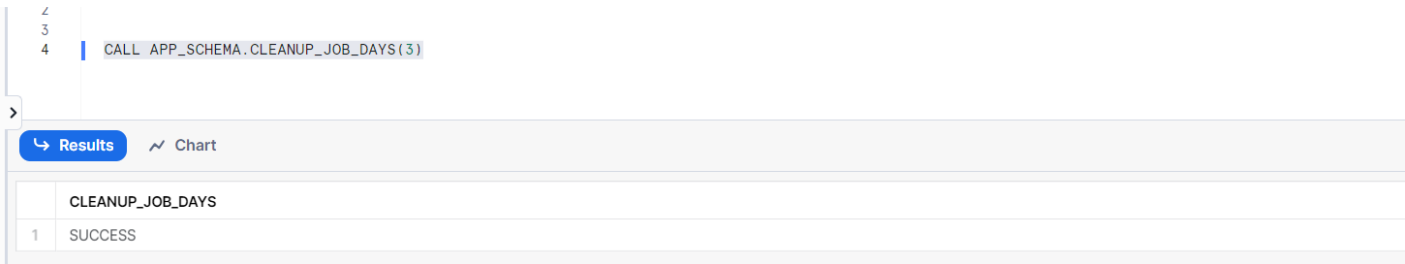
Upon running the cleanup job, the calculation history, calculation response meta-info, and calculation status tables will be updated to reflect the changes while the older calculation tables get deleted.

Syntax - CALL APP\_SCHEMA.CLEANUP\_JOB\_DAYS(n)

**Query:**

```
CALL APP_SCHEMA.CLEANUP_JOB_DAYS(3)
```

**Response table:**



The screenshot shows a query execution window with the following content:

```

2
3
4 | CALL APP_SCHEMA.CLEANUP_JOB_DAYS(3)

```

Below the query, there are tabs for "Results" and "Chart". The "Results" tab is active, displaying a table with the following data:

CLEANUP_JOB_DAYS	
1	SUCCESS

In the above query, the cleanup job is triggered on "25-03-2024" so any calculations that are more than three days old will be deleted.

**Before running the cleanup job query:**

Calculation history table

		SOURCE_NAME	CREATED_DATE	UPDATED_DATE
1	5a1b68ecb25ade4e75a	API_SCHEMA.CALC_dcf1da2b83c845a1b68ecb25ade4e75a_20240320_004...	2024-03-20 00:43:35	2024-03-20 00:43:35
2	08c8892c64b8be1cbce	API_SCHEMA.CALC_acd3ac09f891408c8892c64b8be1cbce_20240320_004...	2024-03-20 00:44:03	2024-03-20 00:44:03
3	fce97d92cd11dc04974	API_SCHEMA.CALC_f57e53b35f614fce97d92cd11dc04974_20240321_025...	2024-03-21 02:55:14	2024-03-21 02:55:14
4	fce97d92cd11dc04974	API_SCHEMA.CALC_f57e53b35f614fce97d92cd11dc04974_20240321_025...	2024-03-21 02:55:51	2024-03-21 02:55:51
5	fce97d92cd11dc04974	API_SCHEMA.CALC_f57e53b35f614fce97d92cd11dc04974_20240321_025...	2024-03-21 02:56:18	2024-03-21 02:56:18
6	469816e5af570a167b2	API_SCHEMA.CALC_8ebe2a81afbb4469816e5af570a167b2_20240321_025...	2024-03-21 02:56:52	2024-03-21 02:56:52
7	099a00ad36de079ee03	API_SCHEMA.CALC_2ab03e585ebf4099a00ad36de079ee03_20240321_025...	2024-03-21 02:57:50	2024-03-21 02:57:50
8	4b5baa2b09d65a79babb	API_SCHEMA.CALC_3b1ac7c6184e4b5baa2b09d65a79babb_20240325_050...	2024-03-25 05:05:54	2024-03-25 05:05:54

The calculation records that are highlighted will be removed, after running the cleanup job query.

		SOURCE_NAME	CREATED_DATE	UPDATED_DATE
1	5a1b68ecb25ade4e75a	API_SCHEMA.CALC_dcf1da2b83c845a1b68ecb25ade4e75a_20240320_004...	2024-03-20 00:43:35	2024-03-20 00:43:35
2	08c8892c64b8be1cbce	API_SCHEMA.CALC_acd3ac09f891408c8892c64b8be1cbce_20240320_004...	2024-03-20 00:44:03	2024-03-20 00:44:03
3	fce97d92cd11dc04974	API_SCHEMA.CALC_f57e53b35f614fce97d92cd11dc04974_20240321_025...	2024-03-21 02:55:14	2024-03-21 02:55:14
4	fce97d92cd11dc04974	API_SCHEMA.CALC_f57e53b35f614fce97d92cd11dc04974_20240321_025...	2024-03-21 02:55:51	2024-03-21 02:55:51
5	fce97d92cd11dc04974	API_SCHEMA.CALC_f57e53b35f614fce97d92cd11dc04974_20240321_025...	2024-03-21 02:56:18	2024-03-21 02:56:18
6	469816e5af570a167b2	API_SCHEMA.CALC_8ebe2a81afbb4469816e5af570a167b2_20240321_025...	2024-03-21 02:56:52	2024-03-21 02:56:52
7	099a00ad36de079ee03	API_SCHEMA.CALC_2ab03e585ebf4099a00ad36de079ee03_20240321_025...	2024-03-21 02:57:50	2024-03-21 02:57:50
8	4b5baa2b09d65a79babb	API_SCHEMA.CALC_3b1ac7c6184e4b5baa2b09d65a79babb_20240325_050...	2024-03-25 05:05:54	2024-03-25 05:05:54

**After running the cleanup job query:**

Results		Chart		
CALC_ID	SOURCE_NAME	...	CREATED_DATE	UPDATED_DATE
1	3b1ac7c6184e4b5baa2b09d65a79babb	API_SCHEMA.CALC_3b1ac7c6184e4b5baa2b09d65a79babb_20240325_050548	2024-03-25 05:05:54.646	2024-03-25 05:05:54.646

## 8. Time zone/ Region

Time entries in your database tables align with your account's Snowflake region. Track all activities accurately as per your local timezone for easy and efficient data management.

## 9. Known Issues

- The 'ACCOUNTADMIN' role is required to run setup and other queries.
- The PACALCULATION() stored procedure call supports only responses up to 16 MB in size as this is a hard limit set by Snowflake.
- The PACALCULATION() stored procedure expects optional parameters as empty.

- DROPPING tables is currently not supported, however users can use DELETE.

## 10. Limitations & Resources

- Currently, the app can successfully consume response sizes under 16MB only.
- There is a need to run some implementation queries to integrate with the PA Engine API.
- Unlike the API, multiple calculations are not supported.
- [https://assets.ctfassets.net/lmz2w5z92b9u/1y6Vd3cUXrjFEPIWc9GRsc/c101d6347018255b703324752d5d0b91/FactSet.PA\\_Engine.API\\_v3.documentation.pdf](https://assets.ctfassets.net/lmz2w5z92b9u/1y6Vd3cUXrjFEPIWc9GRsc/c101d6347018255b703324752d5d0b91/FactSet.PA_Engine.API_v3.documentation.pdf)
- <https://other-docs.snowflake.com/en/native-apps/consumer-about>
- <https://other-docs.snowflake.com/en/native-apps/consumer-installing>
- <https://other-docs.snowflake.com/en/native-apps/consumer-granting-privs>
- <https://other-docs.snowflake.com/en/native-apps/consumer-enable-logging>
- <https://other-docs.snowflake.com/en/native-apps/consumer-managing-applications>