

Building block Configuration Guide

CUSTOMER

Commercial Planning Send SAP Analytics Cloud Sales model
drivers to IBP for Demand
May 2023
English

Commercial Planning Send SAP Analytics Cloud Sales model drivers to IBP for Demand

Contents

1	Prerequisites.....	3
2	Documentation	4
2.1	Starting the flow.....	4
2.2	Reading data from SAP Analytics Cloud	4
2.3	Transformation	5
2.4	Writing into IBP.....	5
2.5	Properties of the Integration Flow.....	5
3	Configuration steps on SAP Cloud Integration.....	6
3.1	Configure Receiver Adapter	6

1 Prerequisites

The Integration Flow “Commercial Planning Send SAP Analytics Cloud Sales model drivers to IBP for Demand” connects the content package model for Sales Planning in SAP Analytic Cloud with SAP IBP for Demand. This flow sends the planned drivers from SAC into IBP. There is also an Integration Flow available to read baseline quantities from IBP to send them into SAC.

This Integration Flow is a possible implementation approach. But it is necessary to check the individual business needs.

2 Documentation

The flow reads fact data from SAP Analytics Cloud model, transforms the data and writes the data into IBP for demand. To minimize the memory footprint, a semantical partitioning on the timestamp is used. Data is read month by month, transformed month by month and written into IBP month by month. As the data is potentially aggregated, it is relevant to have all QUANTITY values for each property combination (each package, means each month) sent to IBP in one loop.

2.1 Starting the flow

The Integration Flow is stated via API call. Externalized Parameter <SAPHDA_API_ENDPOINT>. The Externalized Parameter could be defined as /write2ibp so that the Integration Flow can be called via the URL that can be found in the CI Monitor section.

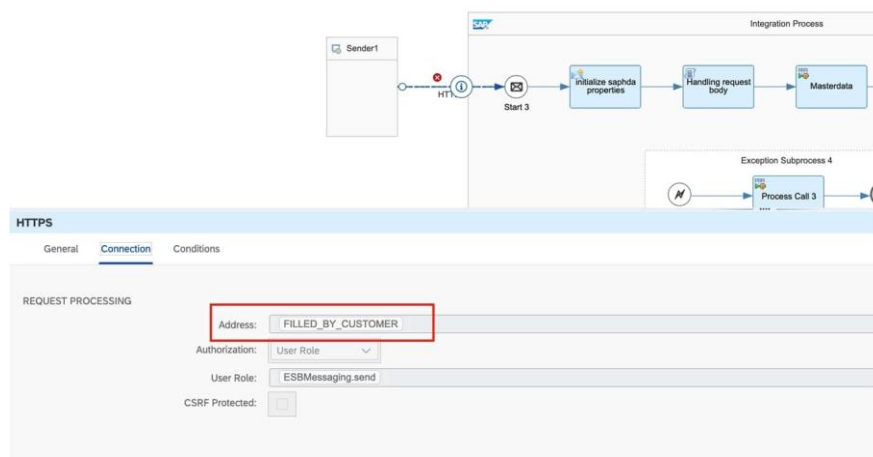


Figure 1 API endpoint definition

The payload that is expected to be sent with this call contains the SAP Analytics Cloud model ID, a date range from when data is read as well as the time horizon that should be exported (calmonthFrom and calmonthTo are included in the boundaries).

An example payload looks like:

```
{
  "modelID": "Cepk9k03peaigaeka98s7gqb32q",
  "calmonthFrom": "202309",
  "calmonthTo": "202402"
}
```

2.2 Reading data from SAP Analytics Cloud

Fact data is read from SAP Analytic Cloud via oData API with the query filter stored in property saphda_queryParameters.

By delivery the configuration is set to

- Version = public.Plan
- SAP_ALL_PLANT = #
- SAP_FI_XPA_GLAccount = 44001000 or 44002000
- Date = is filled automatically by the function semanticPartition of groovy script saphda_logic.groovy

2.3 Transformation

The transformation is processed in the method transform of groovy script saphda_logic.groovy. The transformation derives from the SAP Analytics Cloud fact data response payload the IBP for demand request payload. The script

- Aggregates the QUANTITY. The SAP Analytics Cloud API does not allow a selection query, all properties are selected, also the ones not necessary to send to IBP. Over these properties the QUANTITY is aggregated.
- The SAP Analytics Cloud model is based on Calendar Weeks (YYYYMM), IBP on Demand expects an ISO 8601-1:2019 extended timestamp format (YYYY-MM-DDTHH:MM:SS), so this mapping is done in the method as well
- Currency is added
- The Customer, SAP IBP for Demand field CUSTID, needs to be alpha converted with leading zeros.

2.4 Writing into IBP

Writing into IBP is done as separate flow via process direct call. All customizing is done via exchange properties in this Sales Integration Flow. Values are processed in the method transformation of groovy script saphda_logic.groovy.

- IBP hostname: saphda_ibp_url
Example: https://my-ibp-api.host.ondemand.com
- credentialsName: saphda_ipb_credential
The name of the security material, defined as basic authorization
- planningArea: saphda_ibp_planningarea Example: XPACNT2305

2.5 Properties of the Integration Flow

All custom properties used in this Integration Flow are declared in the content modifier “initialize saphda properties”. But the property saphda_queryParameters needs to be refreshed in each loop, which happens in the content modifier “Refresh saphda properties”. So the odata filter for IBP needs to be customized identically in these two content modifier.

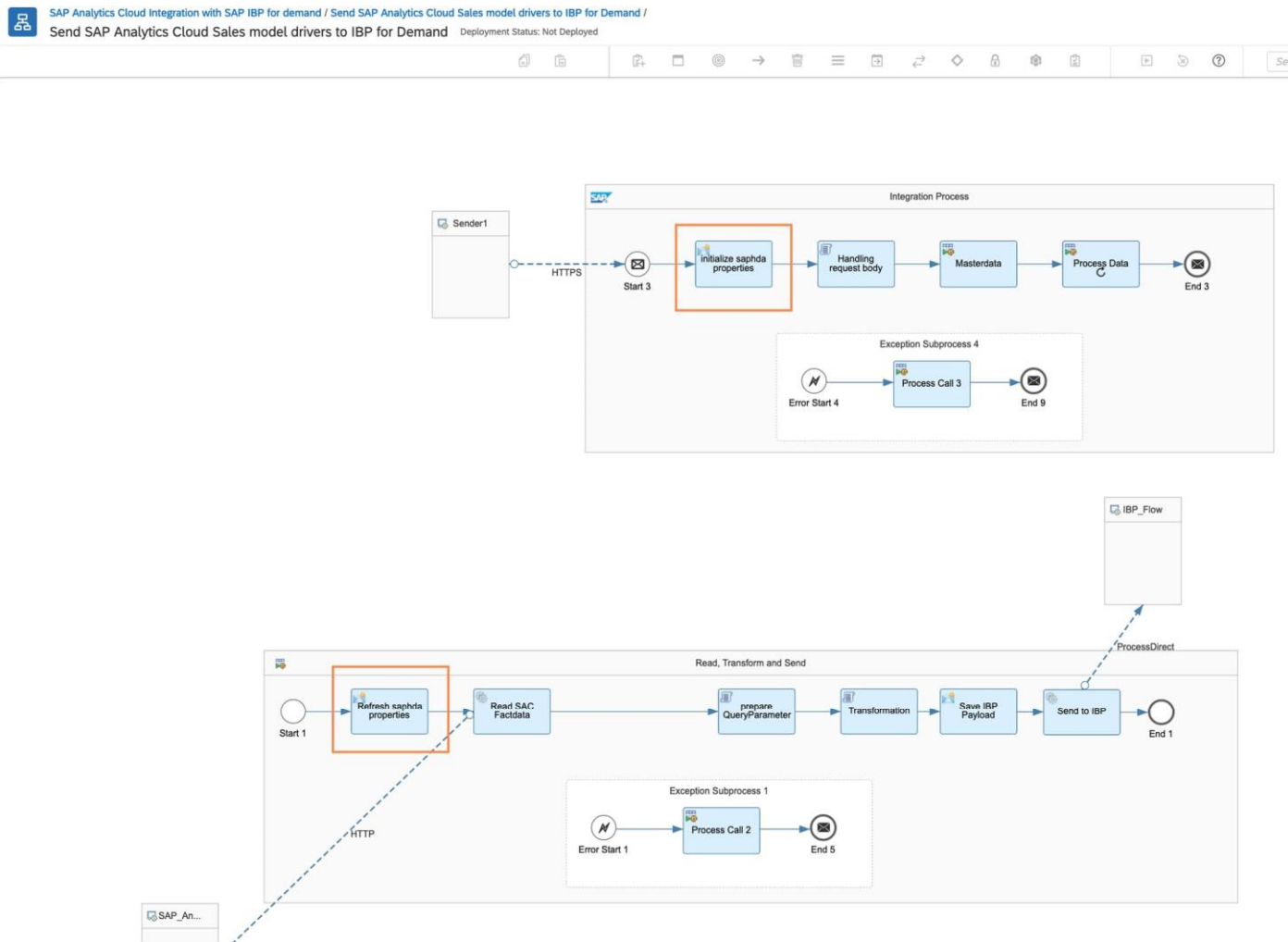


Figure 2 the property saphda_queryParameters needs to be adjusted in the red marked content modifier

3 Configuration steps on SAP Cloud Integration

3.1 Configure Receiver Adapter

Receivers are connecting SAP Analytics Cloud and SAP IBP for Demand. If the flow should be used without adjustments, it is necessary to have the Content Packages for SAP Analytics Cloud installed. In both systems user and authorizations needs to be granted. Please refer to the relevant documentation.

The following configuration is necessary

- Two Credential Artifacts
 - o SAP Analytics Cloud with the credentials of an App Integration OAuth authorization.
Externalized Parameter <SAPHDA_SAC_CREDENTIAL>
 - o IBP for Demand with IBP External Planning Data
Integration Communication Scenario SAP_COM_0720. Exchange Property: saphda_ibp_credential
- URLs for the two Systems
 - o SAP Analytics Cloud
Externalized Parameter <SAPHDA_SAC_URL>
Example: https://host.cloud.sap/api/v1/dataexport/providers/sac/
The URL ends with a slash
 - o IBP for Demand
Exchange Property: saphda_ibp_url
Example: https://host.ondemand.com
Be aware that there is no slash at the end of the URL
- Planning Area for IBP on demand
 - o Exchange Property: saphda_ibp_planningarea

Integration Flow

General

Runtime Configuration

Error Configuration

Resources

Externalized Parameters

Problems

Deployment Status

Name	Default Value
<SAPHDA_API_ENDPOINT>	FILLED_BY_CUSTOMER
<SAPHDA_AUTHENTICATION_OAUTH2>	OAuth2 Client Credentials
<SAPHDA_IFLOW_CREDENTIAL_ROLE>	ESBMessaging.send
<SAPHDA_IFLOW_CREDENTIAL>	RoleBased
<SAPHDA_IFLOW_CSRF>	<input type="checkbox"/>
<SAPHDA_LOOP_EXTYPE>	Non XML
<SAPHDA_LOOP_LIMIT>	throwException
<SAPHDA_SAC_CREDENTIAL>	FILLED_BY_CUSTOMER
<SAPHDA_SAC_URL>	FILLED_BY_CUSTOMER like https://host.cloud.sap/api/v1/da...
<SAPHDA_TIMEOUT>	60000

Figure 3 Configuration of the Externalized Parameter, like the SAC URL