

# Push data into SAP BW/4HANA

One option to connect cloud applications with other SAP and non-SAP cloud and on-premises applications is the usage of an integration flow.

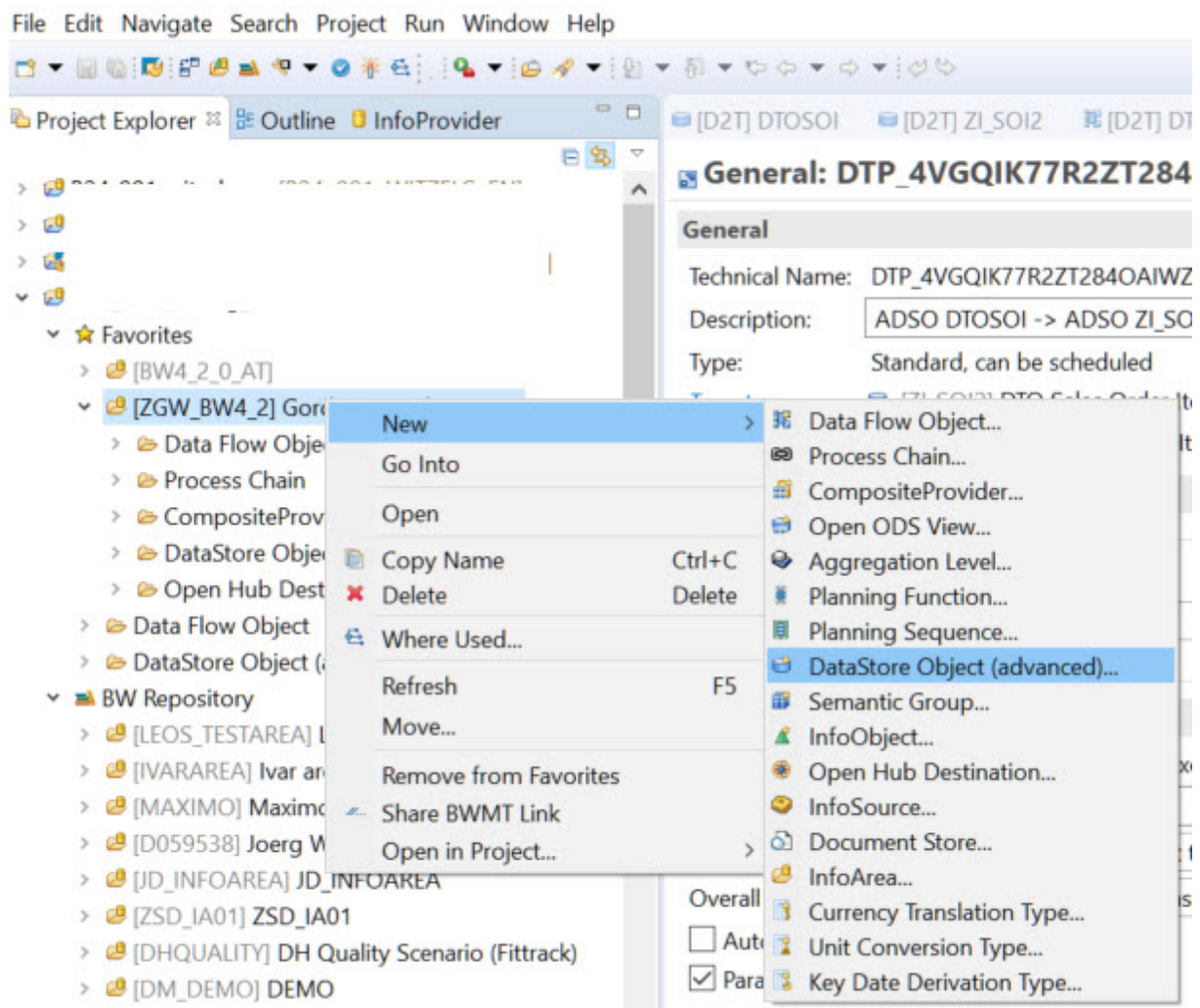
The Integration is using the new http (REST) based interface for data integration to SAP BW/4HANA 2.0.

This blog describes how to push data from the application into a SAP BW/4HANA advanced DataStore Object.

In the first step a BW persistency object must be created.

## 1. Create DataStore Object in SAP BW/4HANA 2.0

Open the BW Modeling Tools in Eclipse and create a new DataStore Object.



Maintain the technical name and the description.

Please use DataStore Object 0EPM\_ADSO3 as an template otherwise the integration flow **will not work**.

New DataStore Object (advanced)

### DataStore Object (advanced)

Create a DataStore Object (advanced)

BW Project:\* D2T\_003\_witzelg\_en Browse...

InfoArea:\* ZGW\_BW4\_2 Browse...

Add to Favorites

Name:\* ZGWCPI

Description: copy of 0EPM\_ADSO3

Copy From: 0EPM\_ADSO3 Browse...

Templates

None

DataSource Browse...

Source System: Browse...

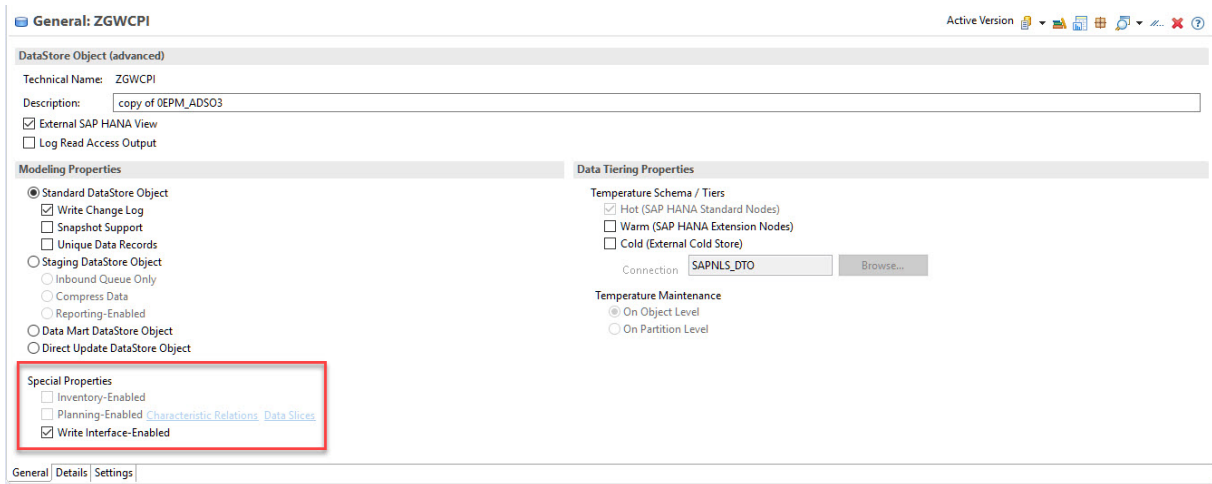
InfoProvider Browse...

InfoObject Browse...

InfoSource Browse...

? Finish Cancel

Maintain the Special Properties in the DataStore Object and set Write-Interface-Enabled.  
Activate the DataStore Object.



## 2. Install the SAP Cloud Connector

Please ensure that the SAP Cloud Connector is already installed and runs properly. Details are available [here](#).

## 3. Maintain the Integration Flow

Open the Integration Web Interface (System dependent).

Open Operations View.

SAP Cloud Platform Integration

Overview

Monitor Message Processing

All Integration Flows Past Hour 0 Messages	All Integration Flows Past Hour 0 Failed Messages	All Integration Flows Past Hour 0 Retry Messages	All Integration Flows Past Hour 0 Completed Messages	+
---	--	---	---	---

Manage Integration Content

All 0 All	All 0 Started	All 0 Error	+
-----------------	---------------------	-------------------	---

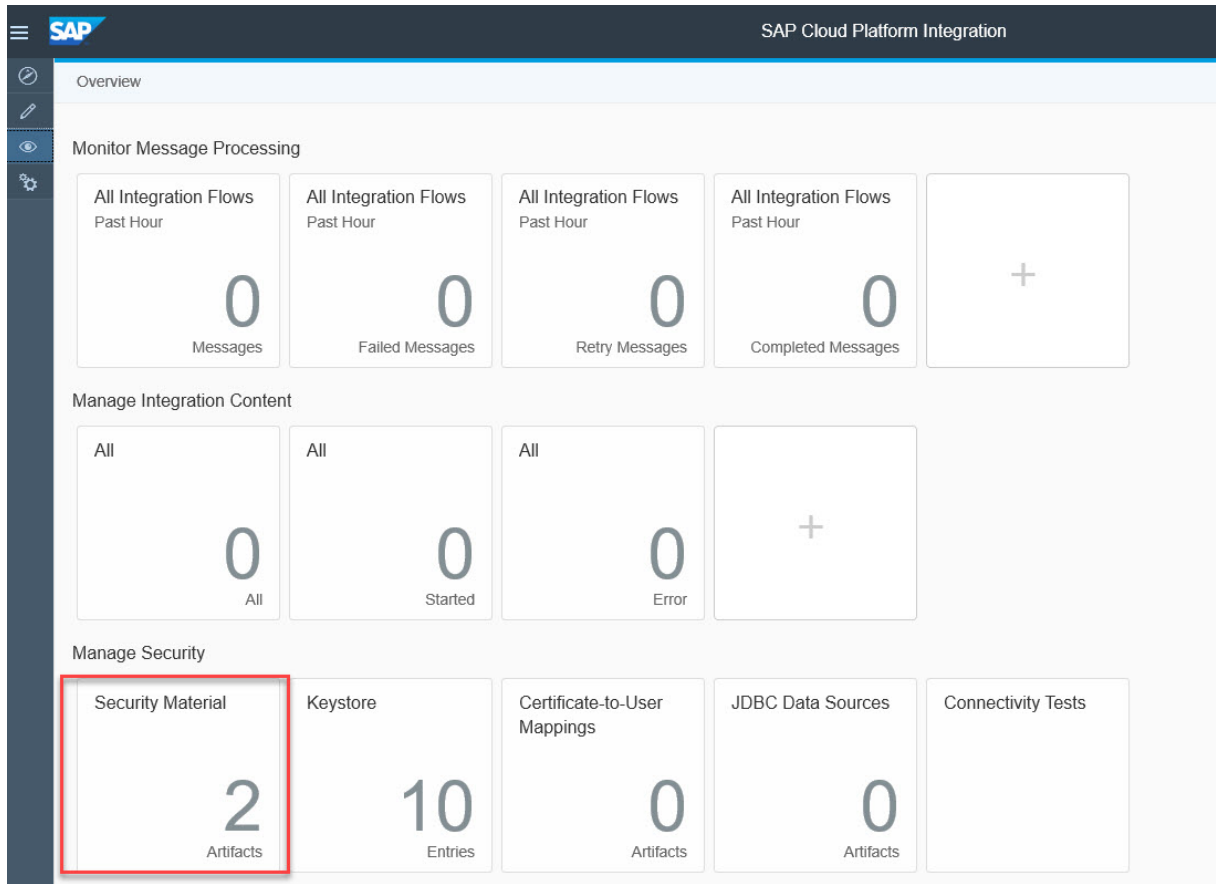
Manage Security

Security Material 2 Artifacts	Keystore 10 Entries	Certificate-to-User Mappings 0 Artifacts	JDBC Data Sources 0 Artifacts	Connectivity Tests
-------------------------------------	---------------------------	---	-------------------------------------	--------------------

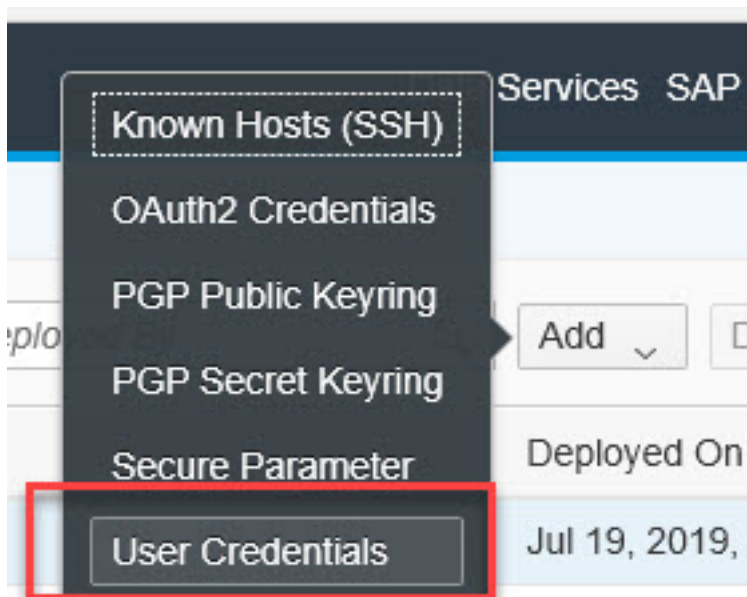
Manage Stores

Data Stores 0 Stores	Variables 0 Variables	Number Ranges 0 Artifacts
----------------------------	-----------------------------	---------------------------------

Open Security Material Tile.



Add new User Credentials to connect from CPI -> SAP BW/4HANA.



Add User Credentials and deploy.

### Add User Credentials

\*Name:

Description:

\*Type:

\*User:

Password:

Repeat Password:

[Deploy](#) [Cancel](#)

Open Discover Page and search for Package **SAP BW/4HANA Integration With Rest Based Data Load**.

Open Artifact View of package SAP BW/4HANA Integration With Rest Based Data Load.

☰ **SAP** SAP Cloud Platform Integration

🔍  
✎  
👁  
⚙

Design / SAP BW/4HANA Integration With Rest Based Data Load /

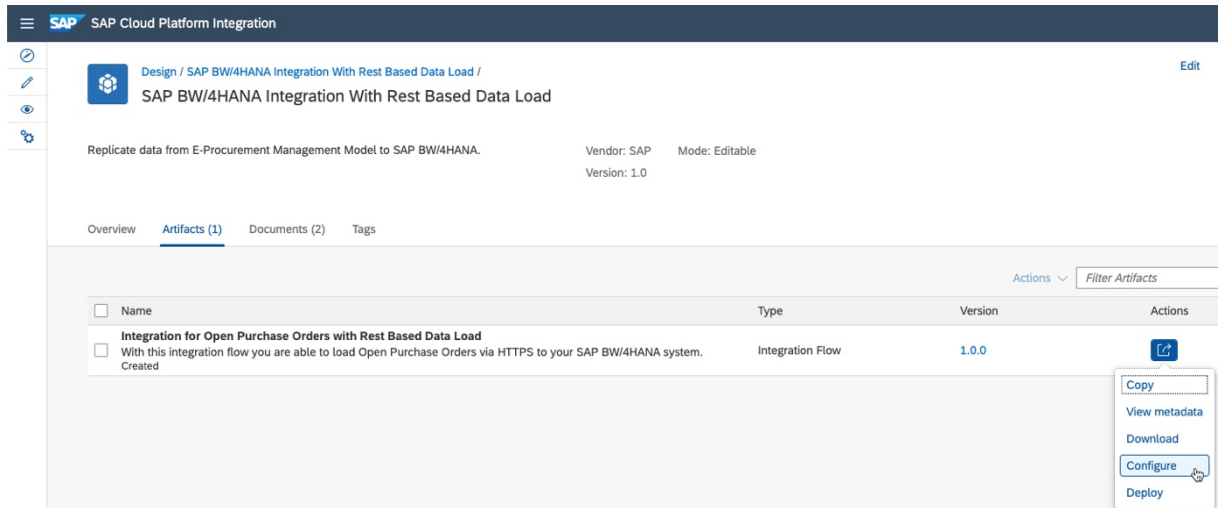
## SAP BW/4HANA Integration With Rest Based Data Load

Replicate data from E-Procurement Management Model to SAP BW/4HANA. Vendor: SAP    Mode: Editable  
Version: 1.0

Overview   Artifacts (1)   Documents (2)   Tags

<input type="checkbox"/>	Name	Type	Version
<input type="checkbox"/>	<b>Integration for Open Purchase Orders with Rest Based Data Load</b> <small>With this integration flow you are able to load Open Purchase Orders via HTTPS to your SAP BW/4HANA system. Created</small>	Integration Flow	1.0.0

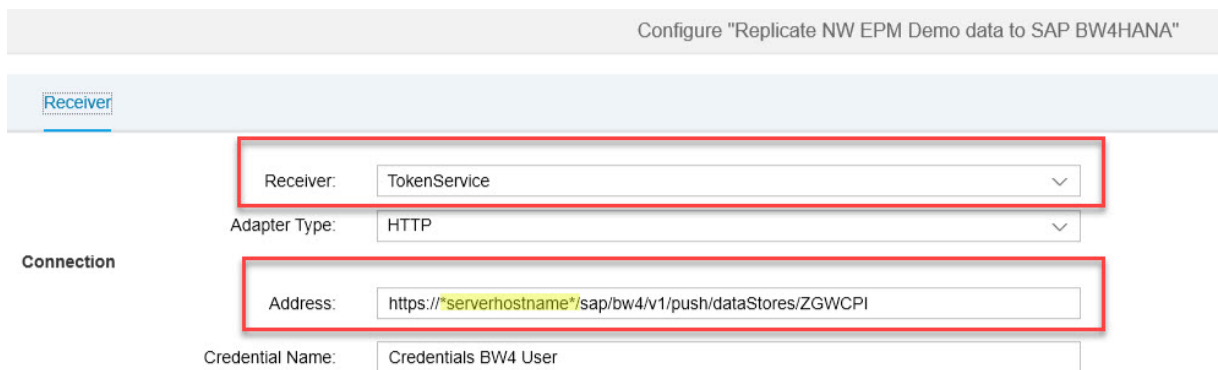
Configure Integration Flow **Integration for Open Purchase Orders with Rest Based Data Load**.



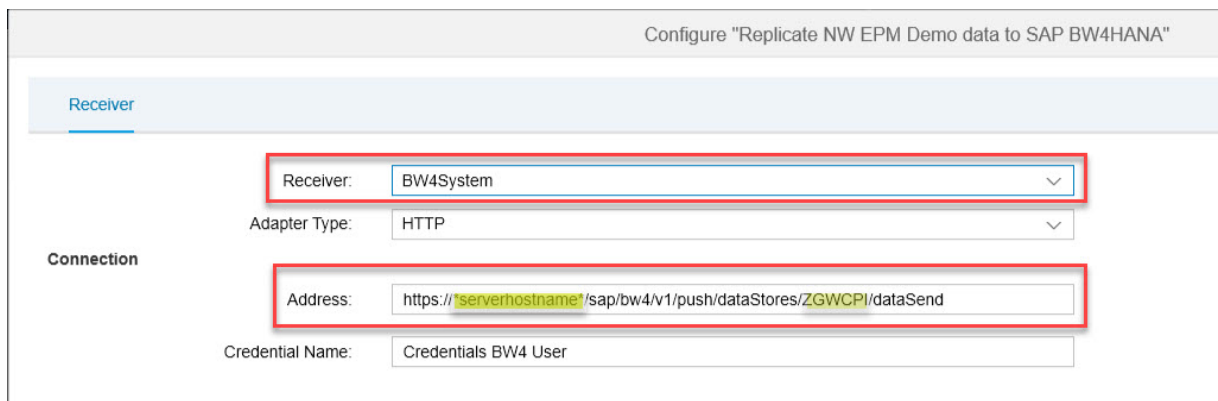
Maintain the Connection for the SAP BW/4HANA System to push data into the defined advanced DataStore Object (created in step 1 in that blog).

It is mandatory to maintain **both receiver connections (BW4System and TokenService)**.

Start with Token Service and maintain the Address -> BW Serverhostname and DataStore Object Name.

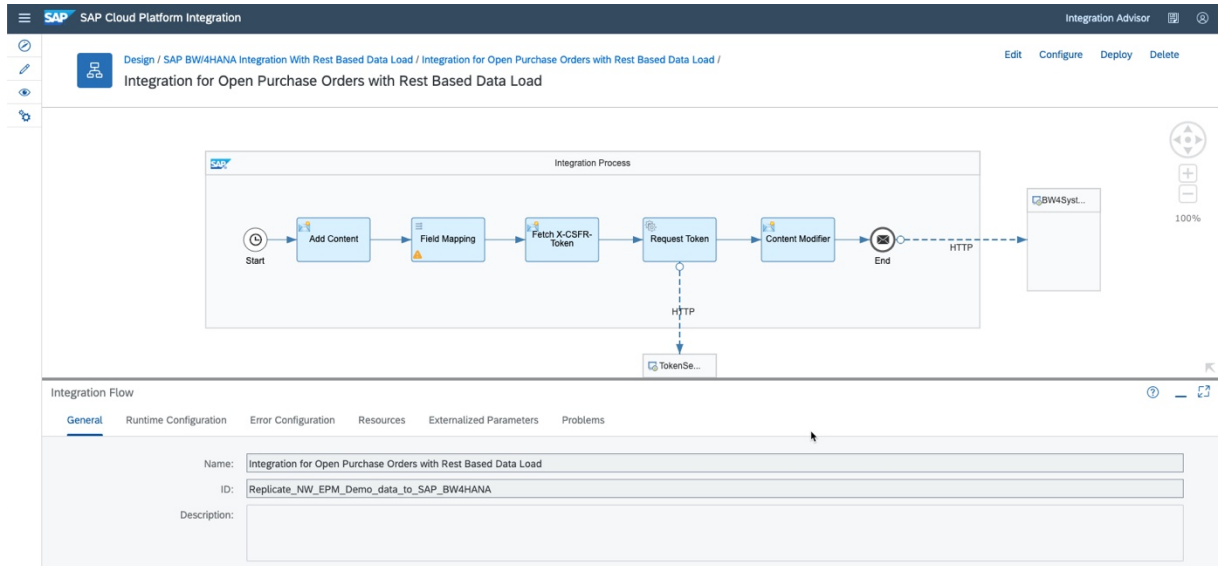


Open Receiver Connection BW4System and maintain Address here as well -> BW Serverhostname and DataStore Object Name.

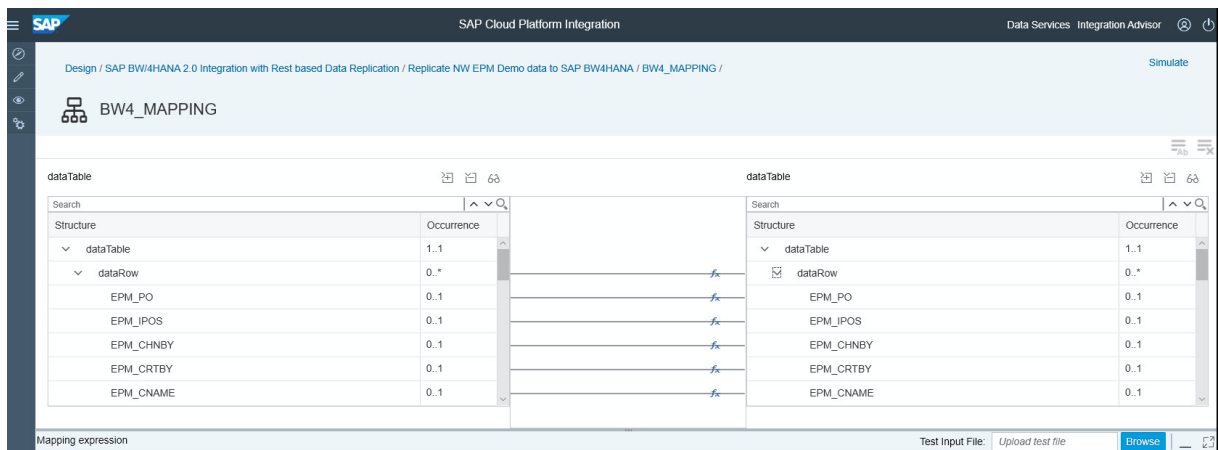


Save and deploy.

Open the Integration flow and check if the details are well maintained.



E.g. check the field mapping.



In 3 fields (EPM\_CHAT, EPM\_PDELD and EPM\_POAS) SAP included a simple type conversion for date and boolean. Here is one example:

```

SAP Cloud Platform Integration

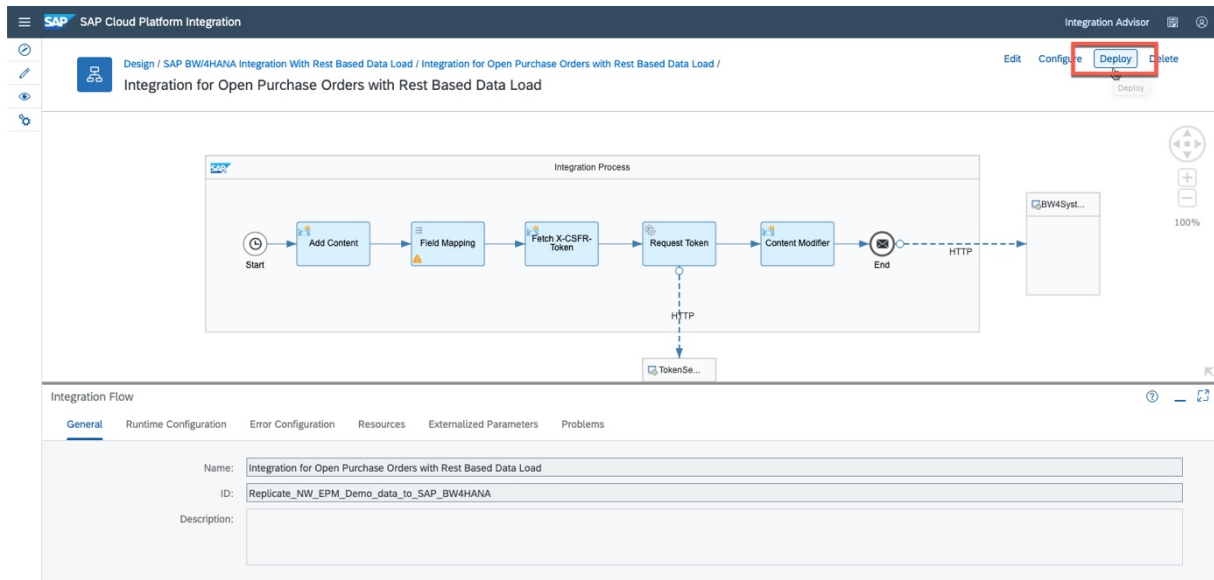
... / Replicate NW EPM Demo data to SAP BW4HANA / BW4_MAPPING / to_abap.groovy /

to_abap.groovy

1 //package src.main.resources.script
2 // <CREATED_ON>2016-05-03</CREATED_ON> 20160503 -> internal DATS format YYYYMMDD
3 def String abap_date(String source_date) {
4     Date date = Date.parse( 'yyyy-MM-dd', source_date)
5     String newDate = date.format( 'yyyyMMdd' )
6     return newDate;
7 }
8
9 //<INELIGIBLE_STATUTORY_MIN_WAGE>false</INELIGIBLE_STATUTORY_MIN_WAGE> -> internal ABAP bool expected: false -> space; true -> X
10 def String abap_bool(String source_bool) {
11     String result = ' ';
12     if (!source_bool?.trim()) return ' ';
13
14     if ('false'.equalsIgnoreCase(source_bool)) return ' ';
15     if ('true'.equalsIgnoreCase(source_bool)) return 'X';
16     return ' ';
17     //throw new IllegalArgumentException(String.format('Cannot convert %s to bool', source_bool));
18 }

```

Once all settings and mappings checked and maintained – deploy the Integration Flow.

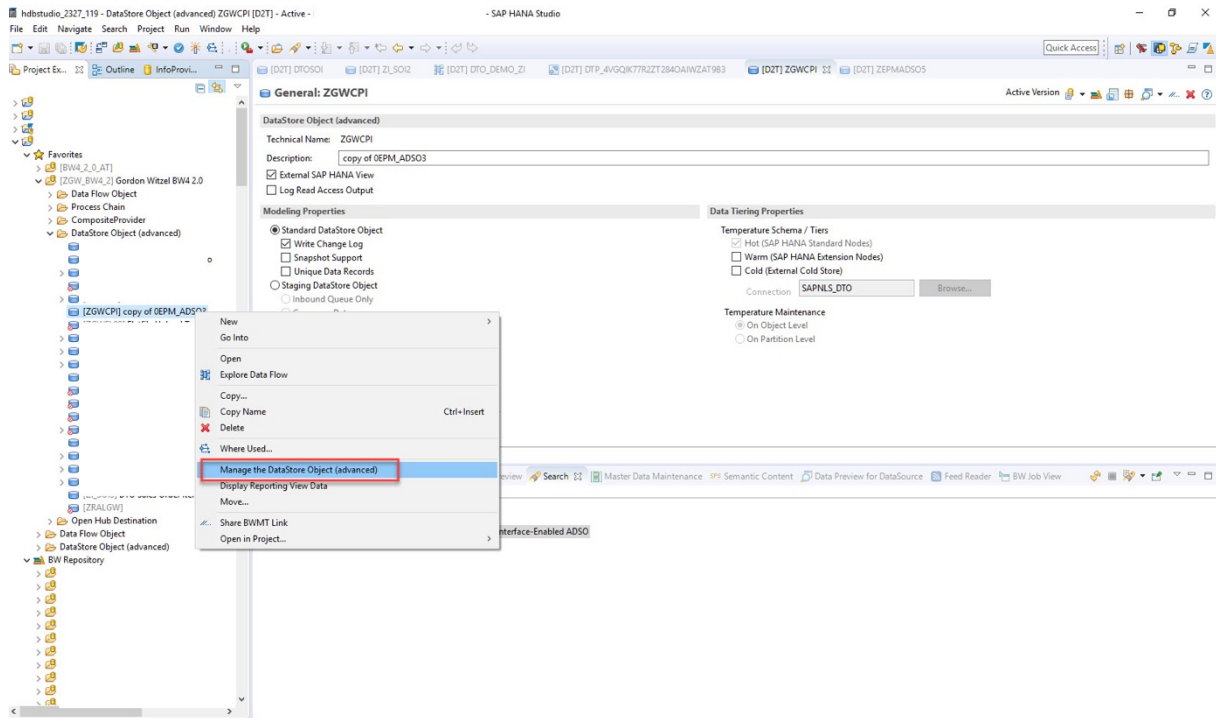


#### 4. Consume the data in SAP BW/4HANA

Go back to the SAP BW/4HANA System and open the BW Modeling Tools in Eclipse.

Search for the Write-Interface-Enabled advanced DataStore Object in which the already created Integration Flow is pushing the data.

Open the Manage view in the context menu of the advanced DataStore Object.



You will now see all the already pushed data from the source to the advanced DataStore Object in SAP BW/4HANA.

The screenshot displays the 'Manage DataStore' view in SAP BW Cockpit. The title bar indicates the object is 'Copy of OEPM\_ADSO3 test Write Interface-Enabled ADSO'. Below the title, there are tabs for 'Default View' and 'Advanced View'. A table titled 'Requests (10 of 12)' shows the following data:

Request	Storage	Records	Last Action	Log	Last Action On	Change Log Indicator
(2019-07-23 09:39:05 000001 CET)	Inbound Table	25	Loading (push)	<a href="#">Log</a>		>
(2019-06-18 09:14:42 000001 CET)	Inbound Table	25	Loading (push)	<a href="#">Log</a>		>
(2019-06-18 08:47:23 000001 CET)	Inbound Table	25	Loading (push)	<a href="#">Log</a>		>
(2019-06-18 08:15:20 000002 CET)	Inbound Table	25	Loading (push)	<a href="#">Log</a>		>
(2019-06-18 08:07:56 000001 CET)	Inbound Table	25	Loading (push)	<a href="#">Log</a>		>
(2019-06-18 07:32:55 000006 CET)	Inbound Table	25	Loading (push)	<a href="#">Log</a>		>
(2019-06-04 08:19:46 000002 CET)	Inbound Table	1	Loading (push)	<a href="#">Log</a>		>
(2019-06-04 08:19:32 000003 CET)	Inbound Table	1	Loading (push)	<a href="#">Log</a>		>
(2019-06-04 08:18:22 000023 CET)	Inbound Table	1	Loading (push)	<a href="#">Log</a>		>
(2019-06-04 08:15:27 000006 CET)	Inbound Table	1	Loading (push)	<a href="#">Log</a>		>

It is now possible to create a virtual model (Composite Provider) or transfer the data to the next layer of your Enterprise data model depending on your data model and the Business requirements.