

Integration Guide | PUBLIC
Document Version: 1.0.2 – 2026-10-30

Integrating Key Figures from SAP Integrated Business Planning to SAP S4HANA Cloud Private Edition or SAP S/4HANA or SAP ERP as Planned Independent Requirements

Integrating SAP Integrated Business Planning with SAP S/4HANA Cloud Private Edition and SAP S/4HANA and SAP ERP Using SAP Cloud Integration

© 2026 SAP SE or an SAP affiliate company. All rights reserved.

Content

1 Document History. 3

2 Introduction. 4

3 Prerequisites. 5

4 Configuring the Integration Flow. 6

4.1 Configuring the Authentication. 6

4.2 Data Mapping. 6

4.3 Defining Additional Parameters. 9

4.4 Adding Further Filters to the Integration Flow. 13

4.5 Working with Extensions. 15

Post Fetch Extension. 15

Custom Mapping Extension. 15

Custom Function Call Extension. 16

Pre-Process Extension. 16

5 Calling the Integration Flow. 17

6 Further Information. 18

1 Document History

The following table provides an overview of the most important changes.

Version	Date	Description
1.0.2	October 30, 2026	Added information about advanced planning filters to the `Filter ID` row and moved the misaligned example to the `Further Filters` row in the Defining Additional Parameters [page 9] section.
1.0.1	August 1, 2026	Updated the Defining Additional Parameters [page 9] section with information related to `I_SAPIBP2`-based planning areas.
1.0.0	May 3, 2026	Initial version

2 Introduction

Using the *Integrate Key Figures from SAP IBP to Add-On as Planned Independent Requirements* integration flow, you can integrate data from SAP Integrated Business Planning (SAP IBP) to SAP S/4HANA Cloud Private Edition or SAP S/4HANA or SAP ERP (target system).

The *Integrate Key Figures from SAP IBP to Add-On as Planned Independent Requirements* integration flow implements the end-to-end process of fetching demand forecasting results from SAP IBP and transferring them at location-product level as planned independent requirements to the target system.

With the default configuration, the integration flow uses an `I_SAPIBP2`-based planning area from the SAP IBP system as source and reads the values of the `I_FINALGLOBALDEMANDPLANQTY` key figure.

3 Prerequisites

You have already set up the connection and done the necessary steps to properly set up the environment, including the following:

- SAP ERP, supply chain integration add-on for SAP Integrated Business Planning or SAP S/4HANA, supply chain integration add-on for SAP Integrated Business Planning (add-ons) 2.0 Support Package 02 or higher.
- SAP Cloud Connector
- SAP Business Technology Platform (SAP BTP)
- SAP Integration Suite
- SAP IBP

You can find an end-to-end connection setup guide in SAP Note 3628813.

4 Configuring the Integration Flow

To be able to transfer data between the target system and SAP IBP, you first need to set up and configure the connection between these systems.

4.1 Configuring the Authentication

The integration flow connects the add-on and the SAP IBP system. Connections, including the authentication method, must be created and configured differently depending on the respective system and the selected authentication method.

Authentication Methods for Connecting to the Add-On

The following authentication methods are available when connecting to the source system:

- Basic authentication
- Technical user propagation
- Principal propagation

To set up the authentication, follow these steps:

1. Connect your SAP BTP subaccount in the SAP Cloud Connector.
2. Configure `Cloud-To-On Premises` mapping in the SAP Cloud Connector.
3. Enable resources for the mapped system.
4. Create RFC destination in SAP BTP.

For more information, see Cloud Connector

4.2 Data Mapping

You can specify the mapping between the attributes and key figure data fetched from SAP IBP and the planned independent requirements in the add-on.

The key externalized parameters of the integration flow for default mapping are the following:

- *Product Source Field*
  It must contain the product attribute name from SAP IBP.

- *Plant Source Field*
  It must contain the location attribute name from SAP IBP.
- *MRP Area Source Field*
  Can optionally contain the attribute name of the location for the MRP Area from SAP IBP.
- *Key Figure Name*
  It must contain name of the key figure from SAP IBP for the planned quantity.

All the above attributes must be on the same planning level in the SAP IBP planning area.

The following default mapping is available in the integration flow:

PIR Attributes in the Add-On	Source Field in SAP IBP	Further Hints
Material	`I_PRDID`	
Plant	`I_LOCID`	
Planned Quantity	`I_FINALGLOBALDEMANDPLANQTY`	The value must be in the base unit of measure that is defined for the material in the target system. Unit conversion happens during extraction from SAP IBP and can be configured by the *Attribute for Unit Conversion* and *Unit of Measure for SAP IBP Read* externalized parameters.

Note

MRP area level PIR integration is supported when using an `I_SAPIBP2`-based planning area with the following restrictions:

- MRP areas are represented as location master data in SAP IBP.
- The `I_LOCID` attribute of the master data contains the name of the MRP area and the `I_REFERENCELOCATION` attribute has the corresponding plant MRP area's name.
  Both the `I_LOCID` and `I_REFERENCELOCATION` attributes are required for the MRP area level PIR integration, but the `I_REFERENCELOCATION` attribute is not included on the planning level `I_WKPRODLOCTSPCUST`. To allow the integration on MRP area level the `I_REFERENCELOCATION` attribute must be added to the planning level of the key figure.

Example

For MRP area level PIR integration, we recommend you use the following mapping:

PIR Attributes in the Add-On	Source Field in SAP IBP
Material	`I_PRDID`
Plant	`I_REFERENCELOCATION`
MRP Area	`I_LOCID`

PIR Attributes in the Add-On	Source Field in SAP IBP
Planned Quantity	`I_FINALGLOBALDEMANDPLANQTY`

By default, the integration flow reads the values of the `I_FINALGLOBALDEMANDPLANQTY` key figure as the output of the demand planning process if you use a planning area based on the `I_SAPIBP2` sample model. However, you can change the key figure parameter to fit your business needs:

- To integrate planned independent requirements as part of the sales and operations planning (S&OP) process, use the `I_FINALCONSDEMANDPLANQTY` key figure.
- If you use a planning area based on the `SAPIBP1` sample model, use the `FINALDEMANDPLANNINGQTY` key figure.

Rounding Key Figure Values

The integration flow automatically converts key figure values from the decimal format of SAP IBP (DECIMAL 18,6 with up to 12 integer digits and 6 decimal places) to the PLNMG field format of the add-ons' (ABAP Type P with 10 integer digits and 3 decimal places). During this conversion, decimal places are rounded from 6 to 3 using a configurable rounding method.

The rounding method can be configured via the header parameter `RoundingMode` with the following supported values:

Value	Description
HALF_UP	Default. Most common rounding, where .5 rounds up to the next whole number.
FLOOR	Rounding towards negative infinity
CEILING	Rounding towards positive infinity
DOWN	Rounding towards zero
UP	Rounding away from zero
HALF_DOWN	.5 rounds down to the next whole number.
HALF_EVEN	Banker's rounding where .5 rounds to the nearest even number.

The header parameter `RoundingMode` is not an externalized parameter of the integration flow, it needs to be created in a wrapper integration flow. If a key figure value exceeds the add-on's maximum capacity of 10 integer digits after rounding, the RFC call will return a conversion error for that record, which should be addressed by correcting the source data in SAP IBP.

4.3 Defining Additional Parameters

You can configure your integration flow by setting externalized parameters. You can either create your own wrapper integration flow to alter these settings or directly change the values in the SAP-delivered integration flow. In the integration flow editor, click *Configure*, then *More* and assign values for the parameter names to set up your integration flow. If you call the integration flow using the *Process Direct*, use the parameter ID for the *Connection Address*.

The following list contains the delivered externalized parameters and their default values:

Parameter Name	Parameter ID	Default Value	Configuration of the Parameter
Attribute for Unit Conversion	`AttributeforUnitConversion`	`I_UOMTOID`	Define the attribute that is used for unit conversion during the key figure extraction from SAP IBP. Mandatory parameter.
Custom Function Call Extension Integration Flow Address	`CustomFunctionCallExtensionIntegrationFlowAddress`		Use this parameter to include a different function call for integrating the PIRs into the staging table in the add-on.
Custom Mapping Extension Integration Flow Address	`CustomMappingExtensionIntegrationFlowAddress`		Use this parameter to include a custom, extended mapping to the integration flow.
Date Type	`DateType`	**2**	Use this parameter to configure the date type for the PIRs in the add-on. Possible values: • 1: Day • 2: Week • 3: Month • 4: Posting Period • 5: Planning calendar period Mandatory parameter.

Parameter Name	Parameter ID	Default Value	Configuration of the Parameter
Delete Obsolete PIRs	`DeleteObsoletePIRs`	**X**	Use this parameter to control how previously integrated data is handled during integration runs. If the parameter is set to '**X**', planned independent requirement data is deleted for location materials before up-to-date data is integrated.
Destination for Add-on	`DestinationforAddon`		Enter the SAP BTP *Destination Name* which points to the target system. Mandatory parameter.
Destination for SAP IBP	`DestinationforSAPIBP`		Enter the SAP BTP *Destination Name* which points to the source SAP IBP system. Mandatory parameter.
Filter ID	`FilterID`		Optionally, enter the ID of a planning filter from the SAP IBP system to narrow down data collection. Applies filter conditions using advanced operators directly at the SAP IBP database level during data extraction, reducing the volume of data transferred. The filter conditions are defined in the SAP IBP *Planning Filters* app. It's only relevant for reading key figures. You can use this parameter along with `FilterUserId`. For the list of supported filter operators, see Using Advanced Operators.

Parameter Name	Parameter ID	Default Value	Configuration of the Parameter
Filter User ID	`FilterUserID`		To apply a planning filter for the data collection in SAP IBP, specify the SAP IBP user that is the owner of that filter definition.
Further Filters	`FurtherFilters`		Optionally, define additional filters for the key figure extraction. **Example** `I_FINALGLOBALDEMANDPLANQTY gt 100 and I_PRDID eq 'EXAMPLEPRODUCTID'`
Key Figure Name	`KeyFigureName`	`I_FINALGLOBALDEMANDPLANQTY`	Enter the name of the key figure in which the result of the demand planning is stored in SAP IBP. Mandatory parameter.
MRP Area Source Field	`MRPAreaSourceField`		Optionally define the attribute in SAP IBP from which the MRP area should be integrated in the add-on.
Package Size in Rows	`PackageSizeinRows`	**`10000`**	Define the package size of reading data from SAP IBP. Mandatory parameter.
PIR Version Active	`PIRVersionActive`	**`X`**	Configure if active or inactive PIRs should be created in the add-on. If the parameter is set to '**X**', active PIRs are created.
PIR Version Number	`PIRVersionNumber`	**`00`**	Configure the version number for the PIRs. Mandatory parameter.

Parameter Name	Parameter ID	Default Value	Configuration of the Parameter
Planning Area	`PlanningArea`		Define the planning area in SAP IBP from which you want data to be collected. Mandatory parameter.
Planning Area Version	`PlanningAreaVersion`		Define the version of the source planning area in SAP IBP.
Plant Source Field	`PlantSourceField`	`I_LOCID`	Define the attribute in SAP IBP from which the plant key is integrated in the add-on. Mandatory parameter.
Post-Fetch extension Integration Flow Address	`PostFetchExtensionIntegrationFlowAddress`		You can define a custom filter expression or enrich your data from an external source.
Pre-Process Extension Integration Flow Address	`Pre-ProcessExtensionIntegrationFlowAddress`		You can define additional steps to be executed before the post processing of the staged PIRs start.
Process PIRs	`ProcessPIRs`	**`X`**	Configure if the PIRs should be processed from the staging table in the add-on. If set to '**`X`**' records will be processed.
Product Source Field	`ProductSourceField`	`I_PRDID`	Define the attribute in SAP IBP from which the product key is integrated in the add-on. Mandatory parameter.
Relative Period End	`RelativePeriodEnd`	**`105`**	Configure the end of the time horizon for the key figure extraction. See the *Period Filtering* section for more information.

Parameter Name	Parameter ID	Default Value	Configuration of the Parameter
Relative Period Start	`RelativePeriodStart`	`0`	Configure the start of the time horizon for the key figure extraction. See the *Period Filtering* section for more information.
Requirements Plan Number	`RequirementsPlanNumber`		Configure the requirements plan number for the PIRs.
Requirements Type	`RequirementsType`	`VSF`	Configure the requirements type for the PIRs.
SAP IBP Read Limit in Rows	`SAPIBPReadLimitinRows`		Optionally, define the maximum number of rows to be collected from SAP IBP.
SAP IBP Time Profile Level	`SAPIBPTimeProfileLevel`	`3`	Define the time profile level in SAP IBP to be used as aggregation level for data collection.
Unit of Measure for SAP IBP Read	`UnitofMeasureforSAPIBP Read`		Define the unit of measure to be used for the key figure extraction from SAP IBP. The PIRs can only be maintained on their base unit of measure in the target system. To avoid data inconsistency the matching unit of measure must be used for the extraction from SAP IBP. Mandatory parameter.

4.4 Adding Further Filters to the Integration Flow

To filter data integrated from SAP IBP, you can use the *Further Filters* parameter in the integration flow configuration.

In the filter expression, you can use the following operators and functions:

- Relational operators
  These operators are `EQ, NE, LT, LE, GT, GE, LIKE`.
- Logical operators
  These operators are `AND, NOT, OR`. Brackets `()` are also supported for grouping.

- Functions
  These functions are `STARTSWITH`, `ENDSWITH`, and `SUBSTRINGOF`.

> **Example**
>
> The following example selects planned independent requirement data related to the `EXAMPLEPRODUCTID` product and has a higher value than 100:
>
> `I_FINALGLOBALDEMANDPLANQTY gt 100 and I_PRDID eq 'EXAMPLEPRODUCTID'`

> **Note**
>
> By default the integration flow selects all maintained values for the key figure from the SAP IBP system, including zero and NULL values. It can be configured to only select records with greater than zero key figure values by setting a filter in the *Further Filters* parameter. With this approach, you can achieve better performance. When integrating data with this configuration, all SAP IBP-relevant open planned independent requirements quantities must be set to zero in the target system before each import. This ensures proper alignment and accuracy.
>
> > **Example**
> >
> > *Further Filters*: `I_FINALGLOBALDEMANDPLANQTY gt 0`
>
> Ensuring that only null key figure values are omitted from the selection while integrating 0 values into the target system is also possible by setting a filter in the *Further Filters* parameter.
>
> > **Example**
> >
> > *Further Filters*: `I_FINALGLOBALDEMANDPLANQTY ne null`

Period Filtering

Use `RelativePeriodStart` and `RelativePeriodEnd` externalized parameters to filter data based on relative periods. This filtering enables you to narrow down your data based on time periods such as days, weeks, and months. You can also use relative period filtering to select data based on a specified offset from the current period. The selected data is dynamically adjusted in relation to the current period, based on the given offset in `RelativePeriodStart` and `RelativePeriodEnd` parameters. The period type used for this filtering is configured in the `SAPIBPPeriodType` parameter.

> **Example**
>
> Next week:
>
> - `RelativePeriodStart`: `0`
> - `RelativePeriodEnd`: `1`
> - `SAPIBPPeriodType`: `3`

4.5 Working with Extensions

Parameters for extensibility allow you to specify additional attribute mappings and filters that can be used to integrate data from external sources. You can choose to call a different function module to insert the data into the staging table.

In general, the extensions are custom-defined integration flows, that will be called by a *Process Direct* call. On the *Configuration* tab of the integration flow, in the *Post-Fetch Extension Integration Flow Address*, the *Custom Function Call Extension Integration Flow Address*, the *Custom Mapping Extension Integration Flow Address* and/or the *Pre-Process Extension Integration Flow Address* fields, you must only add the ID of the custom extension integration flow.

4.5.1 Post Fetch Extension

The *Post Fetch Extension* is primarily designed for enriching the data from a possibly external source, but generally can be used for any kind of custom extensibility implementation to filter or alter the extracted data.

The extension integration flow is called once for each data package, after it has been fetched from SAP IBP.

4.5.2 Custom Mapping Extension

With the *Custom Mapping Extension*, you can change how the data fetched from SAP IBP is mapped to create the PIR items.

The *Custom Mapping Extension* replaces the default mapping that transforms SAP IBP data into PIR items. The extension must return an XML containing planned independent requirement items in the following format:

Sample Code

```
<CT_PIR_IN>
<item>
  <MATERIAL_NUMBER/>
  <PLANT/>
  <SESSIONID/>
  <REQ_PLAN_NUMBER/>
  <REQ_TYPE/>
  <ORDER_FINISH_DATE/>
  <MRP_AREA/>
  <PLANNED_QUANTITY/>
  <UNIT_OF_MEASURE/>
  <VERSION_NUMBER/>
  <VERSION_ACTIVE/>
  <DATE_TYPE/>
</item>
</CT_PIR_IN>
```

4.5.3 Custom Function Call Extension

You can use the *Custom Function Call Extension* to change which function module is used to insert the PIRs into the staging table in the add-on. When using this extension, the mapping between the fetched records and the chosen function module should be implemented in the extension integration flow.

For the main integration flow to function correctly, the chosen function module called from the extension integration flow should:

- Insert the PIRs into the `/IBP/ETS_PIR_2` staging table in the add-on.
- Have an identical interface for the exporting parameters as the `/IBP/ETS_PIR_CI_IN` function module.

4.5.4 Pre-Process Extension

The *Pre-Process Extension* is called after all data has been fetched from SAP IBP, and has been inserted into the add-on staging table, and the SAP IBP connection has been closed, but before post-processing of the PIRs begins. It can be used to execute additional steps between staging and processing.

5 Calling the Integration Flow

If you want to call the *Integrate Key Figures from SAP IBP to Add-On as Planned Independent Requirements* integration flow with the *Process Direct* connection type, use the `Integrate_Key_Figures_from_SAP_IBP_to_Add-on_as_Planned_Independent_Requirements` address.

6 Further Information

To integrate planned independent requirements using SAP Cloud Integration, you need the add-ons 2.0 Support Package 02 or higher.

For more information, see the documentation of the add-ons::

- Application Help for SAP ERP, Supply Chain Integration Add-On for SAP Integrated Business Planning - Integrating Planned Independent Requirements Using SAP Cloud Integration
- Application Help for SAP S/4HANA, Supply Chain Integration Add-On for SAP Integrated Business Planning - Integrating Planned Independent Requirements Using SAP Cloud Integration

.

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

www.sap.com/contactsap

© 2026 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see https://www.sap.com/about/legal/trademark.html for additional trademark information and notices.