Informatica®

Informatica® Cloud Data Integration

SAP Connector

Informatica Cloud Data Integration SAP Connector
April 2024

Publication Date: 2024-04-07

Table of Contents

Preface

Use *SAP Connector* to learn how to read from or write to SAP by using Cloud Data Integration. Learn to create an SAP connection, develop and run synchronization tasks, mappings, mapping tasks, and data transfer tasks in Cloud Data Integration.

Informatica Resources

Informatica provides you with a range of product resources through the Informatica Network and other online portals. Use the resources to get the most from your Informatica products and solutions and to learn from other Informatica users and subject matter experts.

Informatica Documentation

Use the Informatica Documentation Portal to explore an extensive library of documentation for current and recent product releases. To explore the Documentation Portal, visit https://docs.informatica.com.

If you have questions, comments, or ideas about the product documentation, contact the Informatica Documentation team at infa_documentation@informatica.com.

Informatica Intelligent Cloud Services web site

You can access the Informatica Intelligent Cloud Services web site at http://www.informatica.com/cloud. This site contains information about Informatica Cloud integration services.

Informatica Intelligent Cloud Services Communities

Use the Informatica Intelligent Cloud Services Community to discuss and resolve technical issues. You can also find technical tips, documentation updates, and answers to frequently asked questions.

Access the Informatica Intelligent Cloud Services Community at:

https://network.informatica.com/community/informatica-network/products/cloud-integration

Developers can learn more and share tips at the Cloud Developer community:

https://network.informatica.com/community/informatica-network/products/cloud-integration/cloud-developers

Informatica Intelligent Cloud Services Marketplace

Visit the Informatica Marketplace to try and buy Data Integration Connectors, templates, and mapplets:

https://marketplace.informatica.com/

Data Integration connector documentation

You can access documentation for Data Integration Connectors at the Documentation Portal. To explore the Documentation Portal, visit https://docs.informatica.com.

Informatica Knowledge Base

Use the Informatica Knowledge Base to find product resources such as how-to articles, best practices, video tutorials, and answers to frequently asked questions.

To search the Knowledge Base, visit https://search.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Intelligent Cloud Services Trust Center

The Informatica Intelligent Cloud Services Trust Center provides information about Informatica security policies and real-time system availability.

You can access the trust center at https://www.informatica.com/trust-center.html.

Subscribe to the Informatica Intelligent Cloud Services Trust Center to receive upgrade, maintenance, and incident notifications. The Informatica Intelligent Cloud Services Status page displays the production status of all the Informatica cloud products. All maintenance updates are posted to this page, and during an outage, it will have the most current information. To ensure you are notified of updates and outages, you can subscribe to receive updates for a single component or all Informatica Intelligent Cloud Services components. Subscribing to all components is the best way to be certain you never miss an update.

To subscribe, on the Informatica Intelligent Cloud Services Status page, click **SUBSCRIBE TO UPDATES**. You can choose to receive notifications sent as emails, SMS text messages, webhooks, RSS feeds, or any combination of the four.

Informatica Global Customer Support

You can contact a Global Support Center through the Informatica Network or by telephone.

To find online support resources on the Informatica Network, click **Contact Support** in the Informatica Intelligent Cloud Services Help menu to go to the **Cloud Support** page. The **Cloud Support** page includes system status information and community discussions. Log in to Informatica Network and click **Need Help** to find additional resources and to contact Informatica Global Customer Support through email.

The telephone numbers for Informatica Global Customer Support are available from the Informatica web site at https://www.informatica.com/services-and-training/support-services/contact-us.html.

Part I: Introduction to SAP Connector

This part contains the following chapter:

CHAPTER 1

Introduction to SAP Connector

You can use SAP Connector to integrate with SAP systems in batch, asynchronous, or synchronous modes based on your requirements.

SAP integrates multiple business applications and solutions, such as Customer Relationship Management (CRM), Advanced Planner and Optimizer (APO), and Bank Analyzer. Developers can add business logic within SAP using Java 2 Enterprise Edition (J2EE) or Advanced Business Application Programming-Fourth Generation (ABAP/4 or ABAP), a language proprietary to SAP.

Data Integration supports ABAP, IDoc read, IDoc write, BAPI/RFC, ADSO write, and ODP read functions to integrate with SAP systems. You can choose an SAP connection type to connect to SAP systems based on the interface requirements.

You can use the SAP connection in synchronization tasks, mappings, mapping tasks, and data transfer tasks. You can switch mappings to advanced mode to include transformations and functions that enable advanced functionality. Create a synchronization task to synchronize data between a source and target. Create a mapping task to process data based on the data flow logic defined in a mapping or integration template.

For more information about configuring assets and transformations, see *Mappings*, *Transformations*, and *Tasks* in the Data Integration documentation.

Data integration and SAP integration methods

You can use different integration methods to read from and write to SAP at the different level of integrations.

The following table lists the integration method corresponding to the different levels and connections:

SAP Integration Method	Level of Integration	Connections
SAP Table	Data-level integration	SAP Table connection The SAP Table connection information is available in the SAP Table Connector documentation. For more information about the SAP Table connection, see the *SAP Table Connector* documentation.
BAPI/RFC functions	Object-level integration	BAPI/RFC Interface connection
IDoc	Message-level integration	iDoc Reader and iDoc Writer connections

SAP Integration Method	Level of Integration	Connections
SAP ADSO	Object-level integration	SAP Table and SAP ADSO Writer connections The SAP Table connection information is available in the SAP Table Connector documentation. For more information about the SAP Table connection, see the *SAP Table Connector* documentation.
SAP ODP	Object-level integration	SAP ODP Extractor connection The SAP ODP Extractor connection information is available in the SAP ODP Extractor documentation. For more information about the SAP ODP Extractor connection, see the *SAP ODP Extractor Connector* documentation.
SAP BW	Object-level integration	SAP BW connection

Communication interfaces

SAP uses TCP/IP as the native communication interface to communicate with Data Integration.

SAP also uses the Remote Function Call (RFC) communication protocol to communicate with Data Integration. To execute remote calls from Data Integration, SAP requires connection information, and the service name and gateway on the application server. The service and gateway parameters, and connection information is stored in a configuration file named `sapnwrfc.ini` on the Secure Agent machine.

Part II: SAP Connector administration

This part contains the following chapters:

CHAPTER 2

SAP Connector administration

SAP Connector requires configuration on the machine that hosts the Secure Agent and also on the SAP systems. The administrators for each of these systems must perform the configuration tasks for their respective systems.

To process data through SAP, the administrator must verify the required licenses are enabled for the SAP systems.

SAP IDocs and RFCs/BAPI Connector administration

Before you use an SAP connection to process data through IDocs or RFCs/BAPIs, the SAP Administrator must verify the required licenses are enabled and perform prerequisite tasks.

1. Download and install the Microsoft Visual C++ Redistributable.
2. Download and configure the SAP libraries.
3. Configure the `sapnwrfc.ini` file.
4. Define SAP Connector as an external logical system in SAP.
5. Configure SAP user authorization.

After the administrator has performed the configuration, you can create and use SAP RFC/BAPI, IDoc Reader, and IDoc Writer connections in mappings.

Downloading and installing the Microsoft Visual C++ Redistributable

Before you run applications developed with VC++, download and install Microsoft Visual C++ Redistributable Packages for Visual Studio 2013 on the Windows machine that hosts the Secure Agent.

1. Click the following URL:

   https://www.microsoft.com/en-in/download/details.aspx?id=40784
2. Download and install the package titled **Visual C++ Redistributable Packages for Visual Studio 2013**.

   For more information, see the following SAP Note: 2573790

Downloading and configuring SAP libraries for IDoc and BAPI/RFC

Download and configure the SAP NetWeaver RFC SDK libraries. Contact SAP Customer Support if you encounter any issues when you download the libraries.

1. Go to the SAP Support Portal, and then click **Software Downloads**.

   Note: You need to have SAP credentials to access **Software Downloads** from the SAP Support Portal.

2. Download the SAP NetWeaver RFC SDK 7.50 libraries that are specific to the operating system that hosts the Secure Agent process.

   The following table lists the libraries corresponding to the different operating systems:

Operating System	SAP NetWeaver RFC SDK Libraries
Linux 64	- libicudata.so.50 - libicui18n.so.50 - libicuuc.so.50 - libsapnwrfc.so - libsapucum.so
Windows 64	- icudt50.dll - icuin50.dll - icuuc50.dll - libsapucum.dll - sapnwrfc.dll

   Verify that you download the most recent version of the libraries.

3. Copy the SAP NetWeaver RFC SDK 7.50 libraries to the following directory:

   `<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\ext\deploy_to_main\bin\rdtm\`

   Create the `deploy_to_main\bin\rdtm` directory if it does not already exist.

   Note: If you upgrade from a 32-bit operating system, the Secure Agent copies the 32-bit SAP NetWeaver RFC SDK 7.50 libraries to the directory. You must replace the 32-bit libraries with 64-bit libraries. If you upgrade from a 64-bit operating system, you do not need to perform this step. The Secure Agent copies the 64-bit SAP NetWeaver RFC SDK 7.50 libraries to the directory.

4. Set the following permissions for each NetWeaver RFC SDK library:
   - Read, write, and execute permissions for the current user.
   - Read and execute permissions for all other users.

5. Download the 64-bit SAP JCo libraries based on the operating system on which the Secure Agent runs:

Secure Agent System	SAP File Name
Windows	sapjco3.jar sapjco3.dll
Linux	sapjco3.jar libsapjco3.so

   Verify that you download the most recent version of the libraries.

6. Copy the JCo libraries to the following directory:

   `<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\ext\deploy_to_main\bin\rdtm-extra\tpl\sap`

   Create the `deploy_to_main\bin\rdtm-extra\tpl\sap` directory if it does not already exist.

   Note: If you upgrade from a 32-bit operating system, the Secure Agent copies the 32-bit SAP JCo libraries to the directory. You must replace the 32-bit JCo libraries with 64-bit JCo libraries. If you upgrade from a 64-bit operating system, you do not need to perform this step. The Secure Agent copies the 64-bit SAP JCo libraries to the directory.

7. Configure the JAVA_LIBS property in Data Integration.
   1. Log in to Data Integration.
   2. Click **Runtime Environments** to access the **Runtime Environments** page.
   3. To the left of the agent name, click **Edit Secure Agent**.
   4. From the **Service** list, select **Data Integration Server**.
   5. From the **Type** list, select **Tomcat JRE**.
   6. Enter the JAVA_LIBS value based on the operating system on which the Secure Agent runs.

Operating System	Value
Windows	../bin/rdtm-extra/tpl/sap/sapjco3.jar;../bin/rdtm/javalib/sap/sap-adapter-common.jar
Linux	../bin/rdtm-extra/tpl/sap/sapjco3.jar:../bin/rdtm/javalib/sap/sap-adapter-common.jar

      Note: If you copy the value directly from the table, the hyphens (-) in the value are incorrectly copied. Copy the value to a text editor and make sure that the value you copied is not corrupted.

   7. Click **OK** to save the changes.
   8. Repeat steps 2 through 7 on every machine where you installed the Secure Agent.
8. Restart the Secure Agent.

Configuring sapnwrfc.ini

SAP uses the communications protocol, Remote Function Call (RFC), to communicate with other systems. SAP stores RFC-specific parameters and connection information in a file named `sapnwrfc.ini`. To enable the Secure Agent to connect to the SAP system as an RFC client, create and configure the `sapnwrfc.ini` file on the machines that host the Secure Agent.

When you read data from SAP, if you define the path and file name of the `sapnwrfc.ini` file in the SAP connection, the Secure Agent uses the `sapnwrfc.ini` file. However, if you define only the path of the `sapnwrfc.ini` file in the connection, the Secure Agent first verifies if an `sapnwrfc.ini` file exists in the specified path. If the `sapnwrfc.ini` file exists, the Secure Agent uses the `sapnwrfc.ini` file. Else, an exception occurs.

To process data through RFC/BAPIs, read IDocs, and write IDocs, you cannot use the `sapnwrfc.ini` file.

Use a DOS editor or WordPad to configure the `sapnwrfc.ini` file. Notepad can introduce errors to the `sapnwrfc.ini` file.

After you create the `sapnwrfc.ini` file, copy the file to the following directory and restart the Secure Agent:

<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\ext\deploy_to_main\bin\rdtm\

Create the deploy_to_main\bin\rdtm directory if it does not already exist.

Note: If you are upgrading from an earlier version, you do not need to perform this step. The Secure Agent copies the sapnwrfc.ini file to the directory.

For the serverless runtime environment, the sapnwrfc.ini file is copied from the AWS or Azure location to the following serverless agent directory:

/data2/home/cldagnt/SystemAgent/apps/Data_Integration_Server/ext/deploy_to_main/bin/rdtm

Configure the connection entries in the sapnwrfc.ini file

Use the sapnwrfc.ini file to configure the connections that you want to use.

You can configure the following types of connections in the sapnwrfc.ini file:

Connection to a specific SAP application server

Create this connection to enable communication between an RFC client and an SAP system. Each connection entry specifies one application server and one SAP system.

The following sample shows a connection entry for a specific SAP application server in the sapnwrfc.ini file:

```
DEST=sapr3
ASHOST=sapr3
SYSNR=00
```

Connection to use SAP load balancing

Create this connection to enable SAP to create an RFC connection to the application server with the least load at run time. Use this connection when you want to use SAP load balancing.

The following sample shows a connection entry for SAP load balancing in the sapnwrfc.ini file:

```
DEST=sapr3
R3NAME=ABV
MSHOST=infamessageserver.informatica.com
GROUP=INFADEV
```

Connection to an RFC server program registered at an SAP gateway

Create this connection to connect to an SAP system from which you want to receive outbound IDocs.

The following sample shows a connection entry for an RFC server program registered at an SAP gateway in the sapnwrfc.ini file:

```
DEST=sapr346CLSQA
PROGRAM_ID=PID_LSRECEIVE
GWHOST=sapr346c
GWSERV=sapgw00
```

sapnwrfc.ini parameters

The following table describes the parameters that you can define for various connection types in the `sapnwrfc.ini` file.

sapnwrfc.ini Parameter	Description	Applicable Connection Types
DEST	Logical name of the SAP system for the connection. All DEST entries must be unique. You must have only one DEST entry for each SAP system. For SAP versions 4.6C and later, use up to 32 characters. For earlier versions, use up to eight characters.	Use this parameter for the following types of connections: - Connection to a specific SAP application server - Connection to use load balancing - Connection to an RFC server program registered at an SAP gateway
ASHOST	Host name or IP address of the SAP application. The Secure Agent uses this entry to attach to the application server.	Use this parameter to create a connection to a specific SAP application server.
SYSNR	SAP system number.	Use this parameter to create a connection to a specific SAP application server.
R3NAME	Name of the SAP system.	Use this parameter to create a connection to use SAP load balancing.
MSHOST	Host name of the SAP message server.	Use this parameter to create a connection to use SAP load balancing.
GROUP	Group name of the SAP application server.	Use this parameter to create a connection to use SAP load balancing.
PROGRAM_ID	Program ID. The Program ID must be the same as the Program ID for the logical system that you define in the SAP system to send or receive IDocs.	Use this parameter to create a connection to an RFC server program registered at an SAP gateway.
GWHOST	Host name of the SAP gateway.	Use this parameter to create a connection to an RFC server program registered at an SAP gateway.
GWSERV	Server name of the SAP gateway.	Use this parameter to create a connection to an RFC server program registered at an SAP gateway.
TRACE	Debugs RFC connection-related problems. Set one of the following values based on the level of detail that you want in the trace: - 0. Off - 1. Brief - 2. Verbose - 3. Full	Use this parameter for the following types of connections: - Connection to a specific SAP application server - Connection to use load balancing - Connection to an RFC server program registered at an SAP gateway

Sample sapnwrfc.ini file

The following snippet shows a sample `sapnwrfc.ini` file:

```
/*=====================================================================*/
/* Connection to an RFC server program registered at an SAP gateway */
/*=====================================================================*/
```

```
DEST=<destination in RfcRegisterServer>
PROGRAM_ID=<program-ID, optional; default: destination>
GWHOST=<host name of the SAP gateway>
GWSERV=<service name of the SAP gateway>
*=====================================================================*/
/* Connection to a specific SAP application server */
/*=====================================================================*/
DEST=<destination in RfcOpenConnection>
ASHOST=<Host name of the application server.>
SYSNR=<The back-end system number.>
/*=====================================================================*/
/* Connection to use SAP load balancing */
/* The application server will be determined at run time. */
/*=====================================================================*/
DEST=<destination in RfcOpenConnection>
R3NAME=<name of SAP system, optional; default: destination>
MSHOST=<host name of the message server>
GROUP=<group name of the application servers, optional; default: PUBLIC>
```

Defining SAP Connector as a logical system in SAP

To use SAP Connector to send and receive IDocs from SAP, you must define SAP Connector as an external logical system in SAP.

Create a single logical system in SAP for IDoc ALE integration with SAP Connector. When you define SAP Connector as a logical system, SAP acknowledges SAP Connector as an external system that can receive outbound IDocs from SAP and send inbound IDocs to SAP.

Perform the following steps to define SAP Connector as a logical system:

1. Create a logical system in SAP for SAP Connector.
2. Create an RFC destination for SAP Connector.
3. Create a tRFC port for the RFC destination.
4. Create a partner profile for SAP Connector.
5. Create outbound and inbound parameters for the partner profile.

Note: These steps are based on SAP version 4.6C. The steps may differ if you use a different version. For complete instructions on creating a logical system in SAP, see the SAP documentation.

Step 1. Create a logical system for SAP Connector

To uniquely identify SAP Connector as a client within a network, define SAP Connector as an external logical system in SAP.

1. Go to transaction SALE.

   The **Display IMG** window appears.
2. Expand the tree to navigate to the **Application Link Enabling** > **Sending and Receiving Systems** > **Logical Systems** > **Define Logical System** operation.
3. Click the **IMG - Activity** icon to run the Define Logical System operation.

   An informational dialog box appears.
4. Click **Enter**.

   The **Change View Logical Systems** window appears.
5. Click **New Entries**.

   The **New Entries** window appears.
6. Enter a name and description for the logical system entry for SAP Connector.

Step 2. Create an RFC destination

Create an RFC destination and program ID for SAP Connector.

1. Go to transaction SM59.

   The **Display and Maintain RFC Destinations** window appears.
2. Click **Create**.

   The **RFC Destination** window appears.
3. Enter the name of the logical system you created as the RFC destination.
4. To create a TCP/IP connection, enter T as the connection type.
5. Enter a description for the RFC destination.
6. Click **Save**.
7. For Activation Type, click **Registration**.
8. For Program ID, enter the same name as the RFC destination name.

   Use the Program ID as the value for the PROGRAM_ID parameter in the `sapnwrfc.ini` file.

Step 3. Create a tRFC port for the RFC destination

Create a tRFC port for the RFC destination you defined in SAP. SAP uses the tRFC port to communicate with SAP Connector.

1. Go to transaction WE21.
2. Click **Ports** > **Transactional RFC**.
3. Click **Create**.

   The **Ports in IDoc Processing** dialog box appears.
4. Click **Generate Port Name** or **Own Port Name** and enter a name.
5. Click **Enter**.
6. Enter a description for the port.
7. Select the IDoc record version type.
8. Enter the name of the RFC destination you created.

Step 4. Create a partner profile for SAP Connector

Create a partner profile for the logical system you defined for SAP Connector. When SAP communicates with an external system, it uses the partner profile to identify the external system.

1. Go to transaction WE20.
2. Click **Create**.
3. Enter the following properties:

Partner Profile Property	Description
Partner number	Name of the logical system you created for SAP Connector.
Partner type	Partner profile type. Enter LS for logical system for ALE distribution systems.

4. In the **Post-processing** tab, enter the following properties:

Partner Profile Property	Description
Type	User type. Enter US for user.
Agent	The SAP user login name.
Lang	Language code that corresponds to the SAP language. Enter EN for English.

5. In the **Classification** tab, enter the following properties:

Partner Profile Property	Description
Partner class	Enter ALE.
Partner status	Indicates the status of communication with the partner. To communicate with the partner, enter A for active.

Step 5. Create outbound and inbound parameters for the partner profile

Outbound parameters define the IDoc message type, IDoc basic type, and port number for outbound IDocs. Inbound parameters define the IDoc message type for inbound IDocs.

SAP uses outbound parameters when it sends IDocs to SAP Connector. Create an outbound parameter for each IDoc message type that SAP sends to SAP Connector. SAP uses inbound parameters when it receives IDocs from SAP Connector. Create an inbound parameter for each IDoc message type that SAP receives from SAP Connector.

1. From the partner profiles window, click **Create Outbound Parameter**.

   The **Partner Profiles: Outbound Parameters** window appears.

2. Enter the following properties:

Outbound Parameter Property	Description
Message Type	The IDoc message type the SAP system sends to SAP Connector.
Receiver Port	The tRFC port number you defined.
IDoc Type	The IDoc basic type of the IDocs the SAP system sends to SAP Connector.

3. Click **Save**.

   The **Packet Size** property appears.

4. Enter a value between 10 and 200 IDocs as the packet size.

   The packet size determines the number of IDocs that SAP sends in one packet to SAP Connector.

5. Click **Enter**.
6. Repeat steps from 1 to 5 to create an outbound parameter for each IDoc message type that SAP sends to SAP Connector.
7. Click **Create Inbound Parameter**.

   The **Partner Profiles: Inbound Parameters** window appears.

8. For each inbound parameter, enter the following properties:

Inbound Parameter Property	Description
Message Type	The IDoc message type the SAP system receives from SAP Connector.
Process Code	The process code. The SAP system uses the process code to call the appropriate function module to process the IDocs it receives.

9. Click **Enter**.
10. Repeat steps 7 through 9 to create an inbound parameter for each IDoc message type that SAP receives from SAP Connector.

Configuring SAP user authorizations

The SAP administrator needs to create a profile in the development, test, and production SAP system so that you can use the integration features. This profile name must include authorization for the objects and related activities. The profile on the test system should be the same as the profile on the production system.

The setup of the user and profiles is done within SAP using the SAP GUI. This activity is external to Data Integration.

BAPI/RFC

The following table describes the authorization an SAP user requires to run tasks using the BAPI/RFC functions:

Authorization Object	Authorization Value
S_RFC	SYST, SDTX, SDIFRUNTIME, RFC_METADATA, RFC1, RFC2, ABAP4_COMMIT_WORK, BAPI_TRANSACTION_COMMIT

Note: You must add the S_RFC authorization object to run tasks with the BAPI/RFC functions at runtime. In addition to the above authorization, you must have access to any BAPI/RFC function that you want to run.

IDoc

The following table describes the authorization an SAP user requires to run tasks with IDoc messages:

Authorization Object	Authorization Value
S_RFC	SYST, SDTX, SDIFRUNTIME, RFC1, RFC2, EDIMEXT

Note: You must add the S_RFC authorization object to run tasks with the IDoc messages at runtime. In addition to the above authorization, you must have access to specific IDoc and underlying transactions that you want to run.

SAP BW Connector administration

Before you use an SAP BW Reader connection to read SAP BW data, you must verify the required licenses are enabled and perform prerequisite tasks.

1. Download and configure the SAP libraries to read SAP BW data.
2. Install transport files.
3. Configure HTTPS.
4. Configure SAP user authorizations.

Downloading and configuring the libraries for SAP BW data extraction

Before you can use an SAP BW Reader connection, you need to download the SAP JCo libraries and configure them on the machine where the Secure Agent runs.

Contact SAP Customer Support if you encounter any issues with downloading the libraries.

1. Go to the SAP Support Portal, and then click **Software Downloads**.

   Note: You need to have SAP credentials to access **Software Downloads** from the SAP Support Portal.
2. Download the 64-bit SAP JCo libraries based on the operating system on which the Secure Agent runs:

Secure Agent System	SAP File Name
Windows	`sapjco3.jar` `sapjco3.dll`
Linux	`sapjco3.jar` `libsapjco3.so`

   Verify that you download the most recent version of the libraries.

3. Copy the JCo libraries to the following directory:

   `<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\ext\deploy_to_main\bin\rdtm-extra\tpl\sap`

   Create the `deploy_to_main\bin\rdtm-extra\tpl\sap` directory if it does not already exist.

   Note: If you upgrade from a 32-bit operating system, the Secure Agent copies the 32-bit SAP JCo libraries to the directory. You must replace the 32-bit JCo libraries with 64-bit JCo libraries. If you upgrade from a 64-bit operating system, you do not need to perform this step. The Secure Agent copies the 64-bit SAP JCo libraries to the directory.
4. Configure the *JAVA_LIBS* property in Data Integration.
   a. Log in to Data Integration.
   b. Click **Runtime Environments** to access the **Runtime Environments** page.
   c. To the left of the agent name, click **Edit Secure Agent**.
   d. From the **Service** list, select **Data Integration Server**.
   e. From the **Type** list, select **Tomcat JRE**.

f. Enter the following *JAVA_LIBS* value based on the operating system on which the Secure Agent runs:

Operating System	Value
Windows	../bin/rdtm-extra/tpl/sap/sapjco3.jar;../bin/rdtm/javalib/sap/sap-adapter-common.jar
Linux	../bin/rdtm-extra/tpl/sap/sapjco3.jar:../bin/rdtm/javalib/sap/sap-adapter-common.jar

Note: If you copy the value directly from the table, the hyphens (-) in the value are incorrectly copied. Copy the value to a text editor and make sure that the value you copied is not corrupted.

g. From the **Type** list, select **DTM**.

h. Enter the following *JVMClassPath* value based on the operating system on which the Secure Agent runs:

Operating System	Value
Windows	`pmserversdk.jar;../../bin/rdtm-extra/tpl/sap/sapjco3.jar;../../bin/rdtm/javalib/sap/sap-adapter-common.jar`
Linux	`pmserversdk.jar:../../bin/rdtm-extra/tpl/sap/sapjco3.jar:../../bin/rdtm/javalib/sap/sap-adapter-common.jar`

Note: If you copy the value directly from the table, the hyphens (-) in the value are incorrectly copied. Copy the value to a text editor and make sure that the value you copied is not corrupted.

i. Click **OK** to save the changes.

j. Repeat steps 2 through 4 on every machine where you installed the Secure Agent.

5. Restart the Secure Agent.

Installing SAP BW Reader transport files

Install the SAP BW Reader transport files on the SAP machines that you want to connect to. Before you install the transports on the production system, install and test the transports in a development system. Verify that you install the latest transport files to read from the SAP BW object.

Installing transport files

Install the latest transport files from a Secure Agent directory to read data from a Unicode SAP system. The transport files are for SAP NetWeaver BW version 7.x.

1. Find the transport files in the following directory on the Secure Agent machine:

   `<Informatica Secure Agent installation directory>\downloads\package-bwreader.<Latest version>\package\rdtm\sap-transport\SAPBWReader`

2. Copy the cofile transport file to the `Cofile` directory in the SAP transport management directory on each SAP machine that you want to access.

   The cofile transport file uses the following naming convention: `RUN_BWRDR_K<number>.g00`

3. Remove "RUN_BWRDR_" from the file name to rename the cofile. For example, for a cofile transport file named `RUN_BWRDR_K900723.g00`, rename the file to `K900723.g00`.

4. Copy the data transport file to the `Data` directory in the SAP transport management directory on each SAP machine that you want to access.

   The data transport file uses the following naming convention: `RUN_BWRDR_R<number>.g00`

5. Remove "RUN_BWRDR_" from the file name to rename the file. For example, for a data transport file named `RUN_BWRDR_R900723.g00`, rename the file to `R900723.g00`.
6. To import the transports to SAP, in the STMS, click **Extras** > **Other Requests** > **Add** and add the transport request to the system queue.
7. In the **Add Transport Request to Import Queue** dialog box, enter the request number for the cofile transport.

   The request number inverts the order of the renamed cofile as follows: `g00K<number>`.

   For example, for a cofile transport file renamed as `K900723.g00`, enter the request number as `g00K900723`.

8. In the Request area of the import queue, select the transport request number that you added, and click **Import**.

Configuring HTTPS

To connect to SAP through HTTPS and read data from SAP BW sources, you must configure the machine that hosts the Secure Agent and the machine that hosts the SAP system. You must also enable HTTPS when you configure an SAP BW Reader connection in Data Integration.

Perform the following configuration tasks on the Secure Agent and SAP systems:

HTTPS Configuration on the Secure Agent System

To configure HTTPS on the machine that hosts the Secure Agent, perform the following tasks:

1. Create a certificate using OpenSSL and JAVA KeyTool.
2. Convert the OpenSSL certificate (PKCS#12 certificate) to SAP specific format (PSE) using the SAPGENPSE tool.

Currently, self-signed certificates are supported.

HTTPS Configuration on the SAP System

To configure HTTPS on the machine that hosts the SAP system, perform the following tasks:

1. Enable the HTTPS service on the SAP system.
2. Import the certificate in PSE format to the SAP system trust store.

Configuring SAP user authorizations

To process SAP BW objects such as InfoCubes, InfoSets, MultiProviders, and DataStore objects, the SAP administrator needs to configure user authorization within the SAP system. This activity is external to Data Integration.

The SAP administrator must create a profile in the development, test, and production SAP systems so that you can use the integration features. The profile name must include authorization for the objects and related activities. The profile on the test system must be the same as the profile on the production system.

The following table describes the authorization required for SAP objects before you run tasks using SAP BW Connector:

Authorization Object	Field	Value
S_RFC	RFC_TYPE	FUNC, FUGR
	RFC_NAME	/INFADI/BWRDR, /INFADI/ZTEST_COMMUNICATION, BAPI_CUBE_GETLIST, DDIF_FIELDINFO_GET, BAPI_IOBJ_GETDETAIL, RFCPING, RFC_GET_FUNCTION_INTERFACE, RSAB, SYST
	ACTVT	16
S_BTCH_JOB	JOBACTION	RELE
	JOBGROUP	*
S_RS_ADMWB	RSADMWBOBJ	Provide the Administrator Workbench Object name from which you want to read data based on the requirements. For more information about Administrator Workbench Objects, see the SAP documentation.
	ACTVT	3
S_RS_ICUBE	RSINFOAREA	Provide the InfoArea names that you want to access based on the requirements. For more information about InfoAreas, see the SAP documentation.
	RSINFOCUBE	Provide the InfoCube object names that you want to access based on the requirements. For more information about InfoCubes, see the SAP documentation.
	RSIRSICUBEOBJ	DEFINITION, DATA, UPDATERULE
	ACTVT	3

Note: You can also add the following authorization objects based on the requirements of the SAP BW tasks that you run from Data Integration:

- The S_RFC authorization object with the BAPI_ODSO_GETLIST and BAPI_ISET_GETLIST optional RFC objects.
- The S_RS_ISET and S_RS_ODSO authorization objects with the activity value ACTVT=3.

SAP ADSO Writer Connector administration

Before you use an SAP ADSO Writer connection to load data into an ADSO, the SAP Administrator must verify the required licenses are enabled and perform prerequisite tasks.

1. Verify the prerequisites in the SAP server.
2. Install the SAP ADSO Writer connection transport files.

3. Configure the SAP user authorization.
4. Download and configure the SAP libraries.

After the administrator has performed the configuration, you can set up and use an SAP ADSO Writer connection in mapping tasks.

Verifying the prerequisites for SAP server

SAP ADSO Writer Connector supports only the SAP BW/4HANA system.

The minimum version of the BW/4HANA system must be 2.0, and is identified through the SAP component *DW4CORE, Release 200*.

Installing SAP ADSO Writer Connector transport files

Install the SAP ADSO Writer transport files on the SAP machines that you want to access. Before you install the transports on your production system, install and test the transports in a development system.

Verify that you install the latest transport files to write to an SAP ADSO object.

Install the following data file and cofile to load data to an SAP BW/4HANA ADSO object.

Data and Cofile Names	Required/ Optional	Transport Request	Functionality
- K900131.B42 - R900131.B42	Required	- B42900131	Install the transports to write to an SAP ADSO.

Installing transport files

Install the latest transport files from a Secure Agent directory to write to a Unicode SAP BW/4HANA system. The transport files are for SAP BW/4HANA version 2.0 or later.

1. Find the transport files in the following directory on the Secure Agent machine:

   `<Informatica Secure Agent installation directory>\downloads\package-SAPADSO.<Latest version>\package\SAPADSO\sap-transport`
2. Copy the cofile transport file to the `Cofile` directory in the SAP transport management directory on each SAP machine that you want to access.

   The cofile transport file uses the following naming convention: `<TR Number>.<Sap system>`.
3. Copy the data transport file to the `Data` directory in the SAP transport management directory on each SAP machine that you want to access.

   The data transport file uses the following naming convention: `ADSO_WRITER_R<TR Number>.<Sap system>`.
4. To import the transports to SAP, in the STMS, click **Extras** > **Other Requests** > **Add** and add the transport request to the system queue.
5. In the **Add Transport Request to Import Queue** dialog box, enter the request number for the cofile transport.

   The request number inverts the order of the renamed cofile as follows: `<Sap system><number>`.
6. In the Request area of the import queue, select the transport request number that you added, and click **Import**.

7. If you are upgrading from a previous version of the Informatica Transports, select the **Overwrite Originals** option.

Configure SAP user authorization

Configure the SAP user account to process SAP BW/4HANA ADSO data.

The following table describes the required authorization to read from an SAP BW/4HANA ADSO:

Read Object Name	Required Authorization Values	Value	Activity	Design Time/Run Time
S_RFC	RFC_TYPE - Function Group	SYST	16	Both
	RFC_TYPE - Function Module	RFCPING		Both
	RFC_TYPE - Function Group	RFC1		Both
	RFC_TYPE - Function Module	RFC_GET_FUNCTION_INTERFACE		Both
	RFC_TYPE - Function Group	RFC_METADATA		Both
	RFC_TYPE - Function Module	RFC_METADATA_GET		Both
	RFC_TYPE - Function Group	RSDSO_UPDATE		Both
	RFC_TYPE - Function Module	DDIF_FIELDINFO_GET		Both
	RFC_TYPE - Function Module	BW4_FLP_NOTIFICATIONS_PUSH		Run Time
	RFC_TYPE - Function Module	/INFADI/ADSO_LIST		Both
	RFC_TYPE - Function Module	/INFADI/ADSO_METADATA		Both
	RFC_TYPE - Function Module	/INFADI/RSDSO_WRITE_API_RFC		Run Time
S_RS_ADSO	Field Name - RSINFOAREA	*	23	Both
	Field Name - RSOADSONM	*		Both
	Field Name - RSOADSOPAR	Data, Definition		Both

Read Object Name	Required Authorization Values	Value	Activity	Design Time/Run Time
Optional Authorization Object's				
S_RS_AREA	RSINFOAREA	'INFO Area Name'	3	Both
S_ADMI_FCD	S_ADMI_FCD	PADM	N/A	Both

Download and configure the SAP libraries

To read data from an SAP BW/4HANA ADSO, you need to download the SAP JCo libraries and configure them on the machine where the Secure Agent runs. Contact SAP Customer Support if you encounter any issues with downloading the libraries.

1. Go to the SAP Support Portal, and then click **Software Downloads**.

   Note: You need to have SAP credentials to access **Software Downloads** from the SAP Support Portal.
2. Download the 64-bit SAP JCo libraries based on the operating system on which the Secure Agent runs:

Secure Agent System	SAP File Name
Windows	`sapjco3.jar` `sapjco3.dll`
Linux	`sapjco3.jar` `libsapjco3.so`

   Verify that you download the most recent version of the libraries.
3. Copy the JCo libraries to the following directory: `<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\ext\deploy_to_main\bin\rdtm-extra\tpl\sap`

   Create the `deploy_to_main\bin\rdtm-extra\tpl\sap` directory if it does not already exist.
4. Configure the *JAVA_LIBS* property in Data Integration:
   a. Log in to Data Integration.
   b. Click **Runtime Environments** to access the Runtime Environments page.
   c. To the left of the agent name, click **Edit Secure Agent**.
   d. From the **Service** list, select **Data Integration Server**.
   e. From the **Type** list, select **Tomcat JRE**.

f. Enter the JAVA_LIBS value based on the operating system on which the Secure Agent runs.

Operating System	Value
Windows	`../bin/rdtm-extra/tpl/sap/sapjco3.jar;../bin/rdtm/javalib/sap/sap-adapter-common.jar`
Linux	`../bin/rdtm-extra/tpl/sap/sapjco3.jar:../bin/rdtm/javalib/sap/sap-adapter-common.jar`

Note: If you copy the value directly from the table, the hyphens (-) in the value are incorrectly copied. Copy the value to a text editor and make sure that the value you copied is not corrupted.

g. From the **Type** list, select **DTM**.

h. Enter the *JVMClassPath* value based on the operating system on which the Secure Agent runs.

Operating System	Value
Windows	`pmserversdk.jar;../../bin/rdtm-extra/tpl/sap/sapjco3.jar;../../bin/rdtm/javalib/sap/sap-adapter-common.jar`
Linux	`pmserversdk.jar:../../bin/rdtm-extra/tpl/sap/sapjco3.jar:../../bin/rdtm/javalib/sap/sap-adapter-common.jar`

Note: If you copy the value directly from the table, the hyphens (-) in the value are incorrectly copied. Copy the value to a text editor and make sure that the value you copied is not corrupted.

i. Click **OK** to save the changes.

j. Repeat steps 2 through 7 on every machine where you installed the Secure Agent.

5. Restart the Secure Agent.

CHAPTER 3

Serverless runtime environment

You can use the serverless runtime environment to connect to the SAP system when you configure an SAP connection in Data Integration.

When you use the serverless runtime environment, you do not need to download, install, configure, or maintain the Secure Agent machine.

Use the serverless runtime environment for the following SAP connections:

- SAP ADSO Writer
- SAP BAPI
- SAP iDoc Reader
- SAP iDoc Writer
- SAP RFC/BAPI Interface

For more information about serverless runtime environment properties, see *Runtime Environments* in the Administrator documentation.

Using the serverless runtime environment for an SAP connection

Before you use the serverless runtime environment for an SAP connection, you need to perform the prerequisite tasks.

1. Create the following structure for the serverless agent configuration in AWS or Azure: `<Supplementary file location>/serverless_agent_config`
2. Add the libraries in the Amazon S3 bucket or Azure container in the following location in your AWS or Azure account: `<Supplementary file location>/serverless_agent_config/sap`
3. Copy the following code snippet to a text editor:

```
version: 1
agent:
  dataIntegrationServer:
    autoDeploy:
      sap:
        jcos:
          - fileCopy:
              sourcePath: sap/jco/<sapjco_libary_filename>
          - fileCopy:
              sourcePath: sap/jco/<sapjco_libary_filename>
        nwrfcs:
```

```
- fileCopy:
    sourcePath: sap/nwrfc/<rfc_libary_filename>
- fileCopy:
    sourcePath: sap/nwrfc/<sapnwrfc_filename>
```

where the source path is the directory path of the library files in AWS or Azure.

4. Ensure that the syntax and indentations are valid, and then save the file as `serverlessUserAgentConfig.yml` in the following AWS or Azure location: `<Supplementary file location>/serverless_agent_config`
   When the .yml file runs, the libraries are copied from the AWS or Azure location to the serverless agent directory.
5. To configure the JAVA_LIBS or JVMClassPath property for the serverless runtime environment on Linux, perform the following tasks in Administrator:
   a. On the **Serverless Environments** tab, select the serverless runtime environment.
   b. Click **Edit**.
   c. On the **Runtime Configuration Properties** tab, select **Data Integration Server** as the service.
   d. Select the following **Type** and **Name**, and then enter the JAVA_LIBS or JVMClassPath value based on the connection type you want to use for the serverless runtime environment:

Connection	Type	Name	Value
SAP iDoc Reader	Tomcat_JRE	JAVA_LIBS	`../bin/rdtm-extra/tpl/sap/sapjco3.jar:../bin/rdtm/javalib/sap/sap-adapter-common.jar`
SAP iDoc Writer	Tomcat_JRE	JAVA_LIBS	`../bin/rdtm-extra/tpl/sap/sapjco3.jar:../bin/rdtm/javalib/sap/sap-adapter-common.jar`
SAP RFC/ BAPI	Tomcat_JRE	JAVA_LIBS	`../bin/rdtm-extra/tpl/sap/sapjco3.jar:../bin/rdtm/javalib/sap/sap-adapter-common.jar`
SAP BAPI	Tomcat_JRE	JAVA_LIBS	`../bin/rdtm-extra/tpl/sap/sapjco3.jar:../bin/rdtm/javalib/sap/sap-adapter-common.jar`
SAP ADSO Writer	Tomcat_JRE	JAVA_LIBS	`../bin/rdtm-extra/tpl/sap/sapjco3.jar:../bin/rdtm/javalib/sap/sap-adapter-common.jar`
	PMRDTM_CFG	JVMClassPath	`pmserversdk.jar:../../bin/rdtm-extra/tpl/sap/sapjco3.jar:../../bin/rdtm/javalib/sap/sap-adapter-common.jar`

For more information about serverless runtime environment properties, see *Runtime Environments* in the Administrator documentation.

Note: You cannot create an SNC connection when you use the serverless runtime environment.

Part III: Connections

This part contains the following chapters:

CHAPTER 4

SAP connections

Use an SAP connection to read from and write data to SAP systems.

Data Integration supports ABAP, IDoc read, IDoc write, BAPI/RFC, ADSO write, and ODP read functions to integrate with SAP systems. You can choose an SAP connection type to connect to SAP systems based on the interface requirements.

The following table describes the different SAP connections:

SAP Connection Type	Use the connection to ...
SAP Table Connector	Read data from SAP tables and write to any target. You can also write data from any source to custom tables in SAP. Contact Global Customer Support for information about using the SAP Table connection to write data to SAP systems. The SAP Table connection information is available in the SAP Table Connector documentation. For more information about the SAP Table connection, see the *SAP Table Connector* documentation.
SAP RFC/BAPI Interface	Read and write data by using BAPI/RFC functions.
IDoc Reader	Read Intermediate Documents (IDocs) from SAP systems.
IDoc Writer	Write IDocs to SAP systems.
SAP BW Reader	Read data from SAP BW objects such as InfoCubes, InfoSets, MultiProviders, and DataStore objects.
SAP ADSO Writer	Write data to SAP ADSO objects in BW4/HANA.
SAP ODP Extractor	Read data from SAP ODP objects from S/4HANA. The SAP ODP Extractor connection information is available in the SAP ODP Extractor documentation. For more information about the SAP ODP Extractor connection, see the *SAP ODP Extractor Connector* documentation.

SAP IDoc and BAPI/RFC connections

SAP connections enable you to access SAP data through the IDoc or BAPI/RFC interfaces.

When you select **SAP** as the type, you can configure the following connections from the **SAP Connection Type** list:

- IDoc Reader
- IDoc Writer
- SAP RFC/BAPI Interface

SAP RFC/BAPI interface connection properties

To access SAP data through the RFC/BAPI interface, select the **SAP RFC/BAPI Interface** connection type and configure the connection properties.

The following table describes the SAP RFC/BAPI Interface connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	SAP RFC/BAPI Interface
Runtime Environment	The name of the runtime environment where you want to run tasks. Select a Secure Agent or serverless runtime environment.
User Name	Required. SAP user name with authorization on S_DATASET, S_TABU_DIS, S_PROGRAM, and B_BTCH_JOB objects.
Password	Required. SAP password.
Connection String	Required. DEST entry that you specified in the `sapnwrfc.ini` file for the SAP application server.
Code Page	The code page compatible with the SAP target. Select one of the following code pages: - MS Windows Latin 1. Select for ISO 8859-1 Western European data. - UTF-8. Select for Unicode data. - Shift-JIS. Select for double-byte character data. - ISO 8859-15 Latin 9 (Western European). - ISO 8859-2 Eastern European. - ISO 8859-3 Southeast European. - ISO 8859-5 Cyrillic. - ISO 8859-9 Latin 5 (Turkish). - IBM EBCDIC International Latin-1.

Property	Description
Language Code	Required. Language code that corresponds to the SAP language.
Client Code	Required. SAP client number.

SAP IDoc Reader connection properties

To read SAP data through the IDoc interface, select the **iDoc Reader** connection type and configure the connection properties.

The following table describes the SAP IDoc Reader connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	IDoc Reader
Runtime Environment	The name of the runtime environment where you want to run tasks. Select a Secure Agent or serverless runtime environment.
Destination Entry	Required. DEST entry that you specified in the `sapnwrfc.ini` file for the RFC server program registered at an SAP gateway. The Program ID for this destination entry must be the same as the Program ID for the logical system you defined in SAP to receive IDocs.
Code Page	Required. The code page compatible with the SAP source. Select one of the following code pages: - MS Windows Latin 1. Select for ISO 8859-1 Western European data. - UTF-8. Select for Unicode data. - Shift-JIS. Select for double-byte character data. - ISO 8859-15 Latin 9 (Western European). - ISO 8859-2 Eastern European. - ISO 8859-3 Southeast European. - ISO 8859-5 Cyrillic. - ISO 8859-9 Latin 5 (Turkish). - IBM EBCDIC International Latin-1.

SAP IDoc Writer connection properties

To write SAP data through the IDoc interface, select the **iDoc Writer** connection type and configure the connection properties.

The following table describes the SAP IDoc Writer connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	IDoc Writer
Runtime Environment	The name of the runtime environment where you want to run tasks. Select a Secure Agent or serverless runtime environment.
User Name	Required. SAP user name with authorization on S_DATASET, S_TABU_DIS, S_PROGRAM, and B_BTCH_JOB objects.
Password	Required. SAP password.
Connection String	Required. DEST entry that you specified in the `sapnwrfc.ini` file for the SAP application server.
Code Page	Required. The code page compatible with the SAP target. Select one of the following code pages: - MS Windows Latin 1. Select for ISO 8859-1 Western European data. - UTF-8. Select for Unicode data. - Shift-JIS. Select for double-byte character data. - ISO 8859-15 Latin 9 (Western European). - ISO 8859-2 Eastern European. - ISO 8859-3 Southeast European. - ISO 8859-5 Cyrillic. - ISO 8859-9 Latin 5 (Turkish). - IBM EBCDIC International Latin-1.
Language Code	Required. Language code that corresponds to the SAP language.
Client code	Required. SAP client number.

SAP BW Reader connections

You must create an SAP BW Connector connection to read data from SAP BW objects such as InfoCubes, InfoSets, MultiProviders, and DataStore objects. You can use the SAP BW Reader connection in synchronization tasks, mappings, and mapping tasks.

To enable the Secure Agent to connect to the SAP BW system through HTTPS, you must enable HTTPS and specify the keystore details in the SAP BW Reader connection. To read data from SAP BW objects through

HTTPS, specify an SAP BW Reader connection configured for HTTPS when you create synchronization tasks, mappings, or mapping tasks.

SAP BW Reader connection properties

To read data from SAP BW objects, select the **SAP BW Connector** connection type and configure the connection properties.

The following table describes the SAP BW connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	SAP BW Connector
Runtime Environment	The name of the runtime environment where you want to run tasks. Select a Secure Agent.
Username	Required. SAP user name with the appropriate user authorization.
Password	Required. SAP password.
Connection type	Required. Type of connection that you want to create. Select one of the following values: - Application. Create an application connection when you want to connect to a specific SAP BW server. - Load balancing. Create a load balancing connection when you want to use SAP load balancing. Default is Application.
Host name	Required when you create an SAP application connection. Host name or IP address of the SAP BW server that you want to connect to.
System number	Required when you create an SAP application connection. SAP system number.
Message host name	Required when you create an SAP load balancing connection. Host name of the SAP message server.
R3 name/SysID	Required when you create an SAP load balancing connection. SAP system name.
Group	Required when you create an SAP load balancing connection. Group name of the SAP application server.
Client	Required. SAP client number.
Language	Language code that corresponds to the language used in the SAP system.

Property	Description
Trace	Use this option to track the JCo calls that the SAP system makes. Specify one of the following values: - 0. Off - 1. Full Default is 0. SAP stores information about the JCo calls in a trace file. You can access the trace files from the following directories: - Design-time information: `<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<Latest version>\ICS\main\tomcat` - Run-time information: `<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<Latest version>\ICS\main\bin\rdtm`
Additional parameters	Additional JCo connection parameters that you want to use. Use the following format: `<parameter name1>=<value1>, <parameter name2>=<value2>`
Port Range	HTTP port range that the Secure Agent must use to read data from the SAP BW server in streaming mode. Enter the minimum and maximum port numbers with a hyphen as the separator. The minimum and maximum port number can range between 10000 and 65535. Default is 10000-65535.
Use HTTPS	Select this option to enable https streaming.
Keystore location	Absolute path to the JKS keystore file.
Keystore password	Password for the .JKS file.

Property	Description
Private key password	Export password specified for the .P12 file.

SAP Additional Parameters	Additional SAP parameters that the Secure Agent uses to connect to the SAP system as an RFC client.
	Specify the required RFC-specific parameters and connection information to enable communication between Data Integration and SAP.
	You can specify the Secure Network Communication (SNC) parameters as additional arguments to securely connect to SAP as shown in the following format:
	`MSHOST= <Message server hostname>`
	`GROUP=PUBLIC`
	`R3NAME=SLT`
	`SNC_MODE=1`
	`SNC_QOP=3`
	`SNC_MYNAME=p:CN=<Common name>, OU=<Organizational unit>, O=<Organization>, C=<Country> This is the SNC name of the Secure Agent machine.`
	`SNC_PARTNERNAME=p:CN=<Common name>, OU=<Organizational unit>, OU=SAP Web AS, O=<Organization>, C=<Country>. This is the SNC name of the SAP system.`
	`SNC_LIB =<Secure Agent installation directory>/apps/Data_Integration_Server/ext/deploy_to_main/bin/<libsapcrypto.so for Linux/sapcrypto.dll for Windows>`
	`X509CERT=<X509 certificate>`
	For more information about the SNC parameters that you can configure in this field, see the How-To Library article, How to Configure the SAP Secure Network Communication Protocol in Informatica Cloud Data Integration.
	Note: The values of any required connection parameters override SAP additional parameter values that you have entered.

SAP ADSO Writer connections

Use SAP ADSO Writer Connector to connect to the SAP BW/4HANA on-premise application.

You can use SAP ADSO Writer Connector to create an application server connection or a load balancing connection to connect using the SAP user name and password. You can also use create a Secured Network Application (SNC) application server connection or a load balancing connection to connect to SAP BW/4HANA using the SNC parameters.

You must create a connection to SAP BW/4 HANA on-premise systems before designing the mapping or mapping task.

SAP ADSO Writer connection properties

Select the **SAP ADSO Writer** connection type and configure the connection properties.

The following table describes the SAP ADSO Writer connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	SAP ADSO Writer
Runtime Environment	The name of the runtime environment where you want to run tasks. Select a Secure Agent or serverless runtime environment.

SAP Server Connection Type	The SAP server connection type to use.
	Select from the following options:
	- **Application Server Connection**. Connect to an SAP Application Server using the SAP user name and password.
	- **Application Server SNC Connection**. Connect to an SAP Application Server using the secured network connection:
	  - With X.509 Certificate. You do not need to specify the SAP user name and password explicitly. You must provide the path of the x.509 certificate file.
	  - Without X.509 Certificate. You must provide the SAP user name.
	- **Load Balancing Server Connection**. Connect to an SAP Application Server with the least load at run time.
	- **Load Balancing Server SNC Connection**. Connect to an SAP Application Server using SNC with the least load at run time.
	Note: Before you use an SNC connection, you must verify that SNC is configured both on the SAP Server and the machine where the Secure Agent runs.

The following table describes the properties that must configure when you select **Application Server Connection** as the connection type:

Connection property	Description
SAP Client Number	The client number of the SAP Server.
SAP Language	Language code that corresponds to the SAP language.
SAP Application Server	The host name of the SAP Application Server.
SAP System Number	The system number of the SAP Server to connect.
SAP Username	The SAP user name with the appropriate user authorization.

Connection property	Description
SAP Password	The SAP password.

Additional Parameters	Additional SAP parameters that the Secure Agent uses to connect to the SAP system.
	For example, to generate SAP JCo and SAP CPIC trace, specify the following properties:
	`jco.client.trace="1";`
	`jco.client.cpic_trace="3";`
	During the runtime, the JCo and CPIC traces file are generated in the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>\ICS\main\bin\rdtm`
	During the design time, the CPIC traces are generated in the `tomcat.out` files at the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>tomcat.out`

The following table describes the properties that must configure when you select **Load Balancing Server Connection** as the connection type:

Connection property	Description
SAP Client Number	The client number of the SAP Server.
SAP Language	Language code that corresponds to the SAP language.
SAP Message Server	The IP address or the host name of the SAP Message Server.
SAP System ID	The system ID of the SAP Message Server.
SAP Group	The login group name, for example, PUBLIC.
SAP Username	The SAP user name with the appropriate user authorization.
SAP Password	The SAP password.

Additional Parameters	Additional SAP parameters that the Secure Agent uses to connect to the SAP system.
	For example, to generate SAP JCo and SAP CPIC trace, specify the following properties:
	`jco.client.trace="1";`
	`jco.client.cpic_trace="3";`
	During the runtime, the JCo and CPIC traces file are generated in the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>\ICS\main\bin\rdtm`
	During the design time, the CPIC traces are generated in the `tomcat.out` files at the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>tomcat.out`

The following table describes the properties that must configure when you select **Application Server SNC Connection** as the connection type:

Connection property	Description
SAP Client Number	The client number of the SAP Server.
SAP Language	Language code that corresponds to the SAP language.
SAP Application Server	The host name of the SAP Application Server.
SAP System Number	The system number of the SAP Server to connect.
SNC My Name	Optional. The Informatica client Personal Security Environment (PSE) or certificate name. Default length is 256.
SNC Partner Name	The Informatica client PSE or certificate name. Default length is 256.
SNC Quality of Protection (QoP)	Specifies the SAP PSE or certificate name. You can select from the following options: - 1 - Apply authentication only. - 2 - Apply integrity protection (authentication). - 3 - Apply privacy protection (integrity and authentication). - 8 - Apply the default protection. - 9 - Apply the maximum protection. Default is *3 - Apply privacy protection (integrity and authentication)*.
SAP Cryptographic Library Path	The path to the cryptographic library. Specify `sapcrypto.dll` for Windows or `libsapcrypto.so` for Linux.
Use X509 Certificate	Specifies the quality of protection. Select to use X509 Certificate based SNC connection.
X509 Certificate Path or SAP Username	The path to the X509 certificate file. If you select to use the X509 certificate, specify the path to the X509 certificate file with .crt extension. You do not need to specify the SAP user name and password. If you do not want to use the X509 certificate, specify the SAP username for which SNC is configured in SAP Server.

Additional Parameters	Additional SAP parameters that the Secure Agent uses to connect to the SAP system.
	For example, to generate SAP JCo and SAP CPIC trace, specify the following properties:
	`jco.client.trace="1";`
	`jco.client.cpic_trace="3";`
	During the runtime, the JCo and CPIC traces file are generated in the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>\ICS\main\bin\rdtm`
	During the design time, the CPIC traces are generated in the `tomcat.out` files at the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>tomcat.out`

The following table describes the properties that must configure when you select **Load Balancing Server SNC Connection** as the connection type:

Connection property	Description
SAP Client Number	The client number of the SAP Server.
SAP Language	Language code that corresponds to the SAP language.
SAP Message Server	The IP address or the host name of the SAP Message Server.
SAP System ID	The system ID of the SAP Message Server.
SAP Group	The login group name, for example, PUBLIC.
SNC My Name	Optional. The Informatica client PSE or certificate name generated on the Secure Agent machine. Default length is 256.
SNC Partner Name	The Informatica client PSE or certificate name generated on the SAP Server. Default length is 256.
SNC Quality of Protection (QoP)	Specifies the SAP PSE or certificate name. You can select from the following options: - 1 - Apply authentication only. - 2 - Apply integrity protection (authentication). - 3 - Apply privacy protection (integrity and authentication). - 8 - Apply the default protection. - 9 - Apply the maximum protection. Default is *3 - Apply privacy protection (integrity and authentication)*.
SAP Cryptographic Library Path	The path to the cryptographic library. Specify `sapcrypto.dll` for Windows or `libsapcrypto.so` for Linux.
Use X509 Certificate	Specifies the quality of protection. Select to use X509 Certificate based SNC connection.
X509 Certificate Path or SAP Username	The path to the X509 certificate file. If you select to use the X509 certificate, specify the path to the X509 certificate file with .crt extension. You do not need to specify the SAP user name and password. If you do not want to use the X509 certificate, specify the SAP user name for which SNC is configured in the SAP Server.

Additional Parameters	Additional SAP parameters that the Secure Agent uses to connect to the SAP system.
	For example, to generate SAP JCo and SAP CPIC trace, specify the following properties:
	`jco.client.trace="1";`
	`jco.client.cpic_trace="3";`
	During the runtime, the JCo and CPIC traces file are generated in the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>\ICS\main\bin\rdtm`
	During the design time, the CPIC traces are generated in the `tomcat.out` files at the following location:
	`<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<DIS version>tomcat.out`

Creating an SAP IDoc Reader connection

1. Click **Administrator** > **Connections**, and then click **New Connection** to create a connection.

   The **New Connection** page appears.
2. Enter a name for the SAP IDoc Reader connection.

   Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -

   Connection names are not case sensitive.
3. Enter a description for the connection.

   The description can have a maximum length of 255 characters.
4. Select **SAP** as the connection type.

   The **SAP Connection Properties** section appears.
5. Select the name of the runtime environment where you want to run the tasks.
6. Select **iDoc Reader** as the SAP connection type.

   The **iDoc Reader Connection Properties** section appears.
7. Enter the destination entry that you specified in the `sapnwrfc.ini` file for the RFC server program registered at an SAP gateway.

   The Program ID for this destination entry must be the same as the Program ID for the logical system you defined in SAP to receive IDocs.
8. Select UTF-8 as the code page compatible with the SAP source.
9. Click **Test Connection** to test the connection.
10. Click **Save** to save the connection.

Creating an SAP IDoc Writer or SAP RFC/BAPI interface connection

1. Click **Administrator** > **Connections**, and then click **New Connection** to create a connection.

   The **New Connection** page appears.
2. Enter a name for the connection.

   Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -

   Connection names are not case sensitive.
3. Enter a description for the connection.

   The description can have a maximum length of 255 characters.
4. Select **SAP** as the connection type.

   The **SAP Connection Properties** section appears.
5. Select the name of the runtime environment where you want to run the tasks.
6. Select the SAP connection type. You can choose one of the following options:

- To create an IDoc Writer connection, select **iDoc Writer**.
- To create a BAPI/RFC connection, select **SAP RFC/BAPI Interface**.

The connection properties appear.

7. Enter an SAP user name with the appropriate user authorization.
8. Enter the SAP password.
9. Enter the destination entry that you specified in the `sapnwrfc.ini` file for the SAP application server.
10. Select the code page compatible with the SAP system.
11. Enter the language code that corresponds to the SAP language.
12. Enter the SAP client number.
13. Click **Test Connection** to test the connection.
14. Click **Save** to save the connection.

Creating an SAP BW Reader connection

1. Click **Administrator** > **Connections**, and then click **New Connection** to create a connection.

   The **New Connection** page appears.
2. Enter a name for the SAP BW Reader connection.

   Connection names are not case sensitive. Connection names can contain alphanumeric characters, spaces, and the following special characters:

   `_ . + -`
3. Enter a description for the connection.

   The description can have a maximum length of 255 characters.
4. Select **SAP BW Connector** as the connection type.

   The **SAP BW Connector Connection Properties** section appears.
5. Select the name of the runtime environment where you want to run the tasks.
6. Enter an SAP user name with the appropriate user authorization.
7. Enter the SAP password.
8. Select one of the following connection types:
   - Application. Create an application connection if you want to connect to a specific SAP BW server.
   - Load balancing. Create a load balancing connection if you want SAP to connect to the BW server with the least load at run time.
9. Enter the host name or IP address of the SAP BW server that you want to connect to.
10. Enter the SAP system number.
11. Enter the host name of the SAP message server.
12. Enter the SAP system name.
13. Enter the group name of the SAP BW server.
14. Enter the SAP client number.
15. Enter the language code that corresponds to the language used in the SAP system.

16. Enable the trace option to track the JCo calls that the SAP system makes.

    You can enter one of the following values:

    - 0. Off
    - 1. Full

    Default is 0.

    SAP stores information about the JCo calls in a trace file.

    You can access the trace files from the following directories:

    - Design-time information: `<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<Latest version>\ICS\main\tomcat`
    - Run-time information: `<Informatica Secure Agent installation directory>\apps\Data_Integration_Server\<Latest version>\ICS\main\bin\rdtm`

17. Enter any SAP additional connection parameters that you want to use.
18. Enter the HTTP port range that the Secure Agent must use to read data from the SAP BW server in streaming mode.

    Enter the minimum and maximum port numbers with a hyphen as the separator. The minimum and maximum port number can range between 10000 and 65535. When a range is outside the default range, the connection uses the default range. Ensure that you specify valid numbers to prevent connection errors.

    Default is 10000-65535.

19. Select the **Use HTTPS** option to enable https streaming.
    a. Enter the absolute path to the JKS keystore file.
    b. Enter the password for the .JKS file.
    c. Enter the export password for the .P12 file.
20. Click **Test Connection** to determine if the connection to the SAP BW system is successful.
21. Click **Save** to save the connection.

Creating an SAP ADSO Writer connection

1. Click **Administrator** > **Connections**, and then click **New Connection** to create a connection.

   The **New Connection** page appears.
2. Enter a name for the SAP ADSO Writer connection.
3. Enter a description for the connection.

   The description can have a maximum length of 255 characters.
4. Select **SAP ADSO Writer** as the connection type.

   The **SAP ADSO Writer Connector Connection Properties** section appears.
5. Select the name of the runtime environment where you want to run the tasks.

6. In the **Connection** section, perform the following tasks based on your requirement:
   a. Select from one of the following connection types:
      - Application Server. Use this connection type to connect to the SAP Application Server using the SAP user name and password.
      - Application Server SNC. Use this connection type to connect to the SAP Application Server using the secured network.
      - Load Balancing. Use this connection type to connect to the SAP message server that uses a load balanced application server. This connection type uses the SAP user name and password.
      - Load Balancing Server SNC. Use this connection type to connect to the SAP message server using the secured network. This connection type uses the secured network connection.

      Note: The connection properties appear based on the selected connection type. When you use an SNC connection type, you must configure SNC in both SAP and in the Informatica Secure Agent server.
   b. Enter the SAP client number.
   c. Enter the SAP language with which you want to establish the connection.
7. In the **System Connection Details** section, specify the system connection details based on the type of SAP connection type.
   a. If you select the Application Server or the Load Balancing connection type, you must specify the following system details:
      - SAP Application Server. Specify the IP address or the host name of the Application Server to which you want to connect.
      - SAP System Number. Specify the SAP system number.
   b. If you select the Application Server SNC connection or the Load Balancing Server SNC connection, you must specify the following system details:
      - SAP Message Server. Specify the details of the SAP message server, for example, the IP address or the host name.
      - SAP System ID. Specify the system ID of the SAP message server.
      - SAP Group. Specify the logon group name, for example, *PUBLIC*.
8. In the **Connection Logon Details** section, specify the login details depending on the connection type.
   a. If you select the Application Server or Load Balancing Server connection, specify the following SAP credentials:
      - Username. Specify the SAP user name.
      - Password. Specify the SAP password.
   b. If you select the Application Server SNC Server or Load Balancing Server SNC connection, specify the following connection parameters:
      - SNC My Name. The client PSE or certificate name generated for the Secure Agent.
      - SNC Partner Name. The server PSE or certificate name generated on the SAP server.
      - SNC Quality of Protection (QoP). Select from one of the following options:
        - 1 - Apply authentication only.
        - 2 - Apply authentication and integrity protection.
        - 3 - Apply authentication, integrity and privacy protection (encryption).

- 8 - Apply global default protection.
- 9 - Apply the maximum protection.

c. SAP Cryptographic Library Path. Specify the path of the SAP Cryptographic library.

   You must specify `sapcrypto.dll` for Windows or `libsapcrypto.so` for Linux.

d. Use X509 Certificate. Select this option to use the X.509 certificate to log in.

e. X509 Certificate Path or SAP Username. Specify the path of the X509 certificate file with `.crt` extension if you selected the X509 certificate. Otherwise, specify the SAP user name for which SNC is configured in the SAP server.

f. Additional Parameters. Enter any additional connection parameters that you want to use.

9. Click **Test Connection** to determine if the connection to the SAP BW4/HANA system is successful.
10. Click **Save** to save the connection.

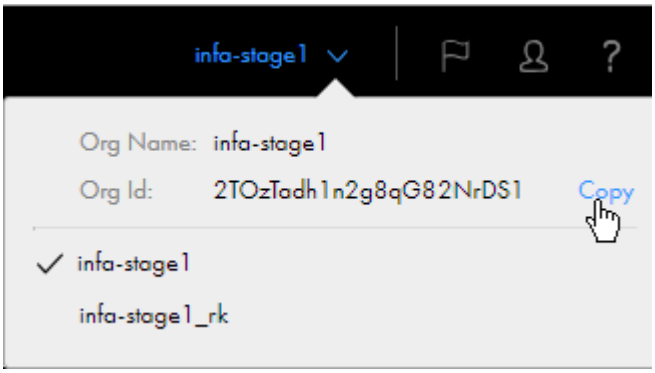
CHAPTER 5

Troubleshooting

Use the following sections to troubleshoot errors in Data Integration.

Note: To get support for Data Integration, you might need to give your organization ID to Informatica Global Customer Support. You can find your organization ID through the **Organization** menu in the upper right corner.

The following image shows the **Organization** menu:



To copy the organization ID, click the **Copy** option that appears when you hover the cursor to the right of the **Org ID** field.

You can also find your organization ID on the **Organization** page in Administrator.

Part IV: Data integration using BAPI/RFC functions

This part contains the following chapters:

CHAPTER 6

Data integration using BAPI/RFC functions

You can import a BAPI/RFC function as a mapplet to Data Integration. You can then use the mapplet in a mapping to read or write data by using the SAP RFC/BAPI Interface connection.

You can also use the BAPI/RFC Interface connection to read data from or write data to an SAP BAPI object that contains a hierarchical structure.

When you run a mapping or mapping task, Data Integration makes the RFC function calls to SAP to process data synchronously.

Business Application Programming Interfaces (BAPI) provide a way for third-party applications to synchronously integrate with SAP at the object-level. You use BAPIs to read, create, change, or delete data in SAP.

BAPIs allow access to the SAP system objects through methods for the business object types. Together with the business object types, BAPIs define and document the interface standard at the business level.

You define BAPIs in the SAP Business Objects Repository. You can call BAPIs as an ABAP program within SAP or from any external application. SAP Connector uses RFC protocol to call BAPI/RFC functions outside of SAP.

You can view and test the BAPI interface definitions in SAP using transaction SE37.

CHAPTER 7

BAPI/RFC mapplets

You can generate SAP BAPI/RFC mapplets in Data Integration.

The BAPI/RFC mapplet includes a BAPI/RFC transformation. The BAPI/RFC transformation makes BAPI/RFC calls in SAP. BAPI/RFC calls include requests to the SAP system, such as creating, changing, or deleting data in SAP applications. To perform these tasks, BAPI/RFC functions use function parameter values.

Configuring SAP BAPI Connector as a Business Service

You can configure SAP BAPI Connector as a business service within a mapping or a mapping task for SAP BAPI data integration.

For more information about configuring SAP BAPI Connector as a business service within a mapping or a mapping task, see the How-To Library article,
How to Configure SAP BAPI Connector as a Business Service in Cloud Data Integration.

BAPI/RFC mapplet parameters

BAPI/RFC functions use function parameter values to perform tasks. A BAPI/RFC mapplet includes input and output groups based on the BAPI/RFC transformation.

Function Parameters

BAPI/RFC functions can have the following parameters:

Function Parameter	Description
Scalar input parameters.	Scalar input values. Some BAPI functions require scalar input values to perform tasks such as changing data.
Scalar output parameters.	Scalar output values that a BAPI function returns after performing a task.
Table parameters.	SAP structures with more than one row. Table parameters can be input, output, or both. Input table parameters pass table input values to a BAPI/RFC function. For example, some BAPI/RFC functions require table inputs to change data.

Input and Output Groups

The BAPI/RFC transformation can contain the following groups:

Group Name	Description
Scalar input	Input group for scalar parameters. Contains a field for each scalar input parameter. The group name is SCALAR_INPUT and the field names are SI_<FieldName>.
Table input	One group for each table parameter. The fields represent import structures. The group name is TABLE_INPUT_<structure name> and field names are TI_<StructureName>_<FieldName>.
Scalar output	Output group for scalar parameters. Contains a field for each scalar output parameter. The group name is SCALAR_OUTPUT and field names are SO_<FieldName>.
Table output	One group for each table parameter. The fields represent output structures. The group name is TABLE_OUTPUT_<structure name> and field names are TO_<StructureName>_<FieldName>.
Error output	Passes data from data conversion errors and invalid BAPI/RFC calls. Map the error output field to a target to see error messages about data conversion and BAPI/RFC calls.

BAPI/RFC parameter properties

When you generate a BAPI/RFC mapplet, you can view the parameter properties and return structures.

You can also change the direction of the BAPI/RFC table parameters. The direction of the table parameters determine which groups in the transformation are input groups and which are output groups.

If the BAPI return structure is custom, you can edit the return structure properties that you selected during import for reusable BAPI/RFC transformations. Otherwise, you can view the BAPI/RFC return structure parameters.

The following table describes the BAPI/RFC function parameter properties:

Parameter	Description
Name	Name of the export, import, and table parameters and columns.
Associated Type	Definition of the parameter in SAP.
Short Description	Short description of the export, import, and table parameters and columns.
Optional	Indicates if the Secure Agent should pass a value to the parameter when it calls the BAPI.
Direction	Indicates if the parameter requires input or provides output. Values are: Input, Output, Both, or None.
Datatype	Object data type.
Precision	Object precision.
Scale	Object scale.
Default	Default value of the parameter in SAP, if any. SAP uses the default value when the Secure Agent does not pass a value to SAP.

The following table describes the parameter properties in the Return Structure tab:

Return Structure Parameter	Description
Return Structure	Return parameter name to determine the status of function calls. Value is RETURN if the BAPI contains a default return structure. If the BAPI does not contain a default return structure, select any table output parameter or scalar output parameter of type STRUCTURE. Default is None.
Status Field	Required if you select a value for the return structure. Select a field from the structure for status. If you select None for the return structure, this parameter is blank.
Text Field	Required if you select a value for the return structure. Select a field from the structure for status messages. If you select None for the return structure, this parameter is blank.
Status Indicator For Warning	Enter an indicator message for warning. If you select None for the return structure, the value is W. Default is W.
Status Indicator for Error	Enter an indicator message for error. If you select None for the return structure, the value is E. Default is E.
Status Indicator for Abort	Enter an indicator message for abort. If you select None for the return structure, the value is A. Default is A.

BAPI/RFC functions with nested structures

You can import metadata for a BAPI/RFC functions with nested structures.

A BAPI/RFC transformation includes multiple groups. When a BAPI function contains a nested structure, ports for the input and output groups in a BAPI/RFC transformation use the following naming convention:

```
<group_name>_<parameter_name>_<field name>
```

For example:

```
SCALAR_INPUT_PARAM1_FIELD1
```

If there are multiple input or output structures, the BAPI/RFC transformation includes each structure parameter name in the port names. For example, BAPI Z_TST2 has the parameter INPUT1, which is of the type ZTYPE1. ZTYPE1 has several components such as FIELD1 and FIELD2. FIELD2 is a component of the type structure. It contains field F1. The naming convention in the BAPI/RFC transformation for FIELD 1 is:

```
SCALAR_INPUT_INPUT1_FIELD1
```

The naming convention in the BAPI/RFC transformation for the field F1 is:

```
SCALAR_INPUT_INPUT1_FIELD2_F1
```

System variables

SAP uses system variables to set default values for some BAPI import parameters. The variables provide information, such as current date and time for the operating system on which SAP runs. System variables start with “SY-”. For example, SY-DATLO represents the local date of the SAP system.

The Secure Agent provides values for some system variables to define default input values for BAPI/RFC parameters. The Secure Agent uses the values as default input values for some ports of BAPI/RFC transformations. The Secure Agent uses the default values when there is no input for a port or when the port is not connected to an upstream transformation or source.

You can use the following system variables:

System Variable Name	Description
SY-LANGU	Log in language from the SAP application connection properties.
SY-MODNO	RFC handle value.
SY-MANDT	Value taken from the SAP application connection properties.
SY-DATUM	Local date on the Security Agent machine processing the data.
SY-UZEIT	Local time on the Security Agent machine processing the data.
SY-UNAME	Logon user ID from the SAP application connection properties.
SY-HOST	SAP host name from the SAP application connection properties.

Integration ID in BAPI/RFC mapplet

The Integration ID field is a key field in the BAPI mapplet. Each BAPI/RFC mapplet includes an Integration ID input field and output field.

When you run a mapping with a BAPI/RFC mapplet, the Secure Agent makes a BAPI/RFC call to SAP to process the data. By default, the BAPI/RFC mapplet commits every 1000 rows when you run a mapping.

The BAPI/RFC call is based on the input data of the Integration ID ports. The Secure Agent makes one call to SAP for each Integration ID. Pass a value to the Integration ID ports in the scalar input group and all mandatory table input groups of the BAPI/RFC mapplet.

Note: You must map the Integration ID input field even when a BAPI or RFC does not require other input fields.

If the BAPI/RFC call fails or if there is a data conversion error, SAP passes the data for the integration ID in comma-separated format to an error output group. If the mapping contains a target instance that is connected to the error output group, the Secure Agent writes the data to the target.

Target object for BAPI/RFC error output

To receive input data from a BAPI/RFC function call or data conversion errors from SAP, you can map a target transformation to a BAPI/RFC mapplet.

Create a target transformation with a column of the String data type and precision of 65535. Connect the column in the target object to the Error Output Group in the BAPI/RFC mapplet. The Secure Agent writes the error output data up to 65,535 characters to the target in comma-delimited format. If the error output data is longer than 65,535 characters, the Secure Agent truncates the data.

Rules and guidelines for BAPI/RFC mapplets in mappings

When you configure a mapping with a BAPI/RFC mapplet, use the following rules and guidelines:

- Pass a value to the Integration ID ports in the scalar input group and all mandatory table input groups of the BAPI/RFC transformation.
- Add a target object if you want to receive BAPI/RFC function call errors from the BAPI error group.

Use the following guidelines when passing data to BAPI/RFC function input parameters:

- When the function input parameter data type is INT1 or NUMC, provide positive values for the function input.
- When the source input data for a BAPI/RFC function is of the integer data type, do not use string data in the source transformation. Otherwise, the mapping fails.
- If the input data for a BAPI/RFC function mapping contains a higher scale than the SAP metadata specification, the Secure Agent rounds the data to comply with the SAP metadata. When you run a mapping in high precision mode, the mapping can fail due to overflow if the round-off data cascades to the precision digits. For example, the data type and precision for a BAPI/RFC function parameter is DEC (6,5). The input data that you pass to the function parameter is 9.99999. When the Secure Agent processes the input data, it rounds the input data to 10, which is not compatible with the SAP metadata. The mapping fails.
- When you run an SAP BAPI mapping, even though the mapping runs successfully, the system log of the SAP system displays the CPIC error.
  To avoid the CPIC error at the SAP system, see KB article 000176711.

CHAPTER 8

Mapping and mapping tasks using BAPI/RFC functions

You can generate a BAPI/RFC mapplet in Data Integration. You can then configure a mapping with the mapplet to manage data in SAP systems.

For example, to update sales order data in SAP, generate a BAPI/RFC mapplet from the BAPI_SALESORDER_CHANGE function and configure a mapping using the mapplet.

You can configure a mapping with a BAPI/RFC mapplet to pass input data to BAPI/RFC function input parameters.

In advanced mode, the Mapping Designer updates the mapping canvas to include transformations and functions that enable advanced functionality.

To access the BAPI/RFC functionality through Data Integration, perform the following tasks:

1. Generate a BAPI/RFC mapplet in Data Integration.
2. Configure a mapping using the generated BAPI/RFC mapplet. Map the Integration ID field and other inputs and outputs. Save and run the mapping or create a mapping task using this mapping.

Configuring SAP BAPI Connector as a Business Service

You can configure SAP BAPI Connector as a business service within a mapping or a mapping task for SAP BAPI data integration.

For more information about configuring SAP BAPI Connector as a business service within a mapping or a mapping task, see the How-To Library article,
How to Configure SAP BAPI Connector as a Business Service in Cloud Data Integration.

Generating a BAPI/RFC mapplet

You can generate a BAPI/RFC mapplet in Data Integration and use the mapplet in a mapping.

Perform the following steps to generate a BAPI/RFC mapplet:

1. To create a mapplet, click **Data Integration** > **New** > **Mapplets** > **Mapplet - SAP BAPI** and then click **Create**.

   To edit a mapplet, on the **Explore** page, navigate to the mapplet. In the row that contains the mapplet, click **Actions** and select **Edit**.
2. Enter a unique name for the BAPI/RFC mapplet.
3. Optionally, enter a description for the BAPI/RFC mapplet that you want to generate.

4. Select the transformation type as **BAPI**.
5. Select the BAPI/RFC connection that you want to use.
6. Select the scope of the transformation.

   For real-time processing, select the transformation scope as **Transaction**. For batch processing and validation in a non-production environment, select the transformation scope as **All Input**.
7. Enter the name of the BAPI/RFC function.

   You can also search for a BAPI/RFC function by entering `%` as a wildcard character along with the search string. For example, to search for all BAPI/RFC functions whose names start with the term `Customer`, enter the search string as `Customer%`. To search for all BAPI/RFC functions whose names end with the term `Customer`, enter the search string as `%Customer`.
8. Click **Get Objects**.

   The **Select Source Object** dialog box appears displaying the search results. You can drill down the search results further by entering a search string in the **Select Source Object** dialog box.
9. Click the required BAPI/RFC function name and click **Select**.

   The import, export, and table parameter details of the BAPI/RFC function appear.
10. Specify the direction for the tables to indicate if the table parameters in the BAPI are input, ouput, or both.
11. Click **Save**.

    The mapplet for the specified BAPI/RFC function is created. You can create a mapping for the generated mapplet.

Configuring a mapping with a BAPI/RFC mapplet

1. To create a mapping, click **Data Integration** > **New** > **Mappings**. Select **Mapping** and click **Create**.
2. Enter a name and description for the mapping, and click **OK**.

   You can use alphanumeric characters and underscores (_) in the mapping name.
3. To configure a source, on the **Transformation** palette, click **Source**.
4. In the **Properties** panel, on the **General** tab, you can enter a name and description.
5. Click the **Source** tab and configure the source details.

   Source details and advanced source properties appear based on the connection type.
6. To add a BAPI/RFC mapplet transformation, on the **Transformation palette**, click Mapplet.
   a. On the **General** tab, enter a name and description for the mapplet.
   b. Draw a link to connect the previous transformation to the mapplet transformation.
   c. On the **Mapplet** tab, click **Select**.

      The **Select Mapplet** dialog box appears.
   d. Specify a BAPI/RFC mapplet that you generated in Data Integration and click **OK**.
   e. Click **Connection** to specify an SAP RFC/BAPI Interface connection. You can create a connection, select a connection, or specify a parameter name for the connection.
   f. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.

g. Click **Field Mapping** and map the incoming source fields with the Integration ID in the BAPI/RFC mapplet.

7. To add any other transformation, on the **Transformation palette**, click the transformation name. Or, drag the transformation onto the mapping canvas.
   a. On the **General** tab, enter a name and description for the transformation.
   b. Draw a link to connect the previous transformation to the transformation.

      When you link transformations, the downstream transformation inherits the incoming fields from the previous transformation.

      For a Joiner transformation, draw a master link and a detail link.
   c. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.
   d. Configure additional transformation properties, as needed.

      The properties that you configure vary based on the type of transformation you create.
   e. To add another transformation, repeat these steps.
8. To add a Target transformation, on the **Transformation palette**, click **Target**.
   a. On the **General** tab, you can enter a name and description.
   b. Draw a link to connect the previous transformation to the Target transformation.
   c. Click the **Target** tab and configure target details. If necessary, configure the advanced target properties.

      Target details and advanced target properties appear based on the connection type.
   d. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.
   e. Click **Field Mapping** and map the fields that you want to write to the target.
   f. To add another Target transformation, repeat these steps.
9. Save and run the mapping or save and create a mapping task.

For more information about transformations, see *Transformations*.

Mappings with BAPI/RFC function example

You can use the `bapi_salesorder_createfromdat1` BAPI function to create sales order details for a customer in your organization.

In this example to create a sales order, perform the following tasks:

Step1: Generate a BAPI/RFC mapplet in Data Integration for the bapi_salesorder_createfromdat1 BAPI function.

Log in to Data Integration and generate a BAPI/RFC mapplet that contains the metadata of the bapi_salesorder_createfromdat1 BAPI function.

Step: 2 Configure a mapping using the generated mapplet.

Perform the following steps to configure a mapping:

1. Specify source objects to enter the order header data from the ORDER_HEADER_IN structure, the partner data using the ORDER_PARTNERS table, and item data using the ORDER_ITEMS_IN table as input parameters.

2. Add the Mapplet transformation. Draw a link to connect the flat file Source transformation to the Mapplet transformation. Draw the following links:
   - ORDER_HEADER_IN source object to the Scalar_Input input port of the BAPI mapplet.
   - ORDER_ITEMS_IN source object to the Table_Input_Order_Items_IN input port of the BAPI mapplet.
   - ORDER_PARTNERS source object to the Table_Input_Order_partners input port of the BAPI mapplet.
3. Configure the mapplet transformation.
   a. Select the generated mapplet. Verify that you specify an SAP RFC/BAPI Interface connection for the mapplet.
   b. Map the incoming source fields with the BAPI parameter properties.
4. Configure a flat file object to which you can write the sales order details. Draw a link to connect the Table Output in the Mapplet transformation to the flat file Target transformation. Create multiple flat file target objects to write the sales order. Create the following target objects and map them to the associated output ports in the BAPI mapplet:
   - ORDER_CFGS_BLOB, ORDER_CFGS_INST, ORDER_CFGS_PART_OF, ORDER_CFGS_REF, and ORDER_CFGS_VALUE to write item configuration data.
   - ORDER_ITEMS_OUT to write detailed item data.
   - ORDER_CCARD to write the credit card details.
   - ORDER_SCHEDULE_EX to write the structure of VBEP with English field names.
   - Scalar_Output to write the Scalar_Output from the mapplet.
   - Error_Output to write the Error_Output from the mapplet.

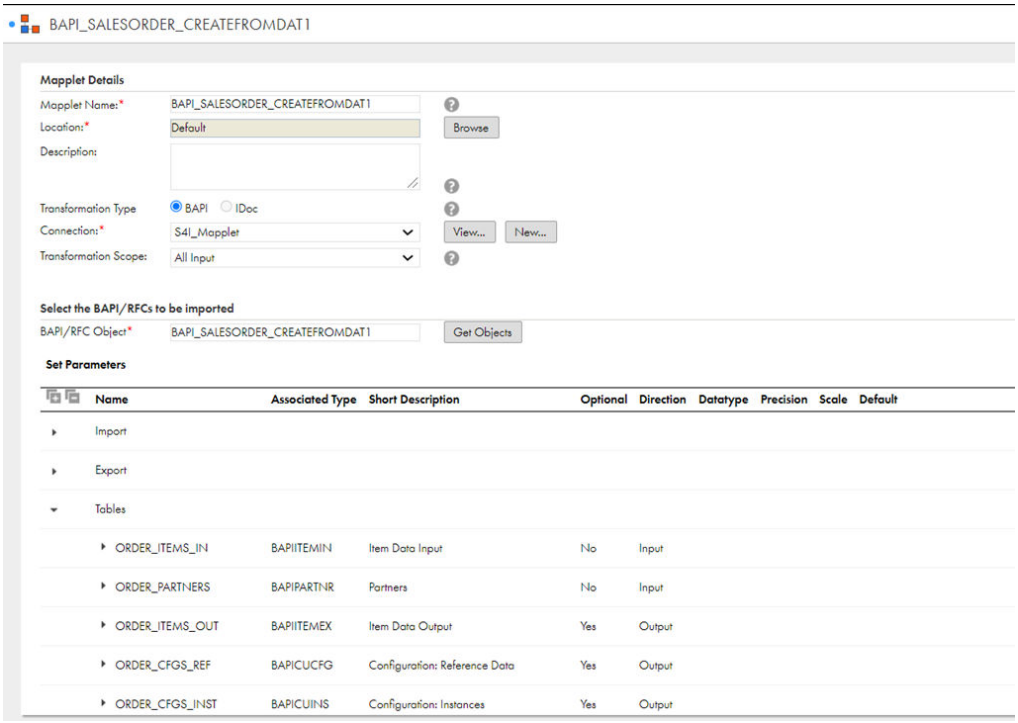
Step 1: Generating a BAPI/RFC mapplet in Data Integration for the BAPI_SALESORDER_CREATEFROMDAT1 BAPI function

Perform the following steps to generate a BAPI/RFC mapplet:

1. To create a mapplet, click **Data Integration** > **New** > **Mapplets** > **Mapplet - SAP** and then click **Create**.

   To edit a mapplet, on the **Explore** page, navigate to the mapplet. In the row that contains the mapplet, click **Actions** and select **Edit**.

   The **New SAP Mapplet** page appears. The following image shows the **New SAP Mapplet** page:



2. Enter a unique name for the BAPI/RFC mapplet.
3. Optionally, enter a description for the BAPI/RFC mapplet you want to generate.
4. Select the transformation type as **BAPI**.
5. Select the BAPI/RFC connection that you want to use.
6. Select the scope of the transformation.

   In real time, select the transformation option as Transaction. For batch processing and validating in non-production environment, select ALL INPUT as the transformation scope.
7. Enter the name of the BAPI/RFC function.

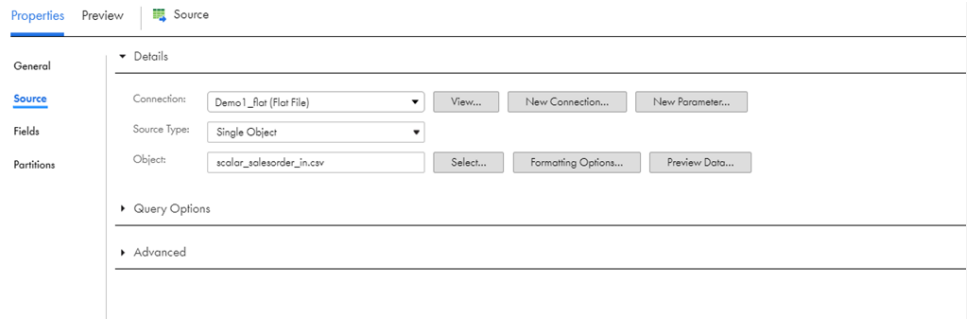
   Note: Verify that you enter the exact name of the BAPI/RFC function that you want to retrieve.
8. Click **Get Objects**.

   You can view the import, export, and table parameter details of the BAPI/RFC function.
9. Specify the direction for the tables to indicate if the table parameters in the BAPI are input, ouput, or both.
10. Click **OK**.

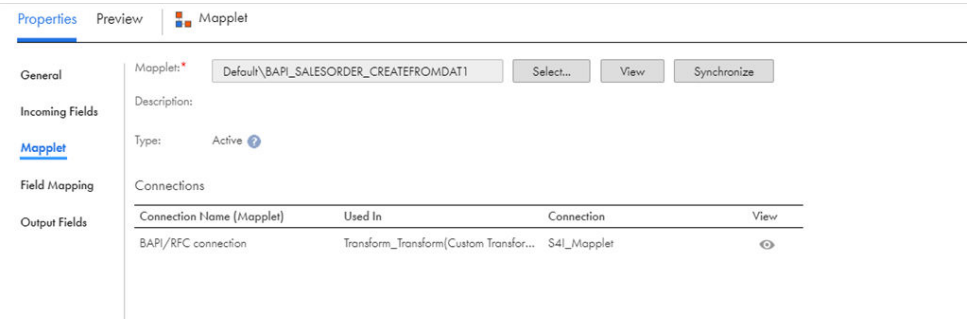
    The mapplet for the specified BAPI/RFC function is created. You can create a mapping for the generated mapplet.

Step 2: Configuring a mapping with the bapi_salesorder_createfromdat1 mapplet

1. To create a mapping, click **Data Integration** > **New** > **Mappings**. Select **Mapping** and click **Create**.
2. Enter a name and description for the mapping, and click **OK**.
3. To configure the flat file source, on the **Transformation** palette, click **Source**.
4. In the **Properties** panel on the **General** tab, enter a name and description.
5. Click the **Source** tab and configure the source details.
   a. Select a flat file connection.
   b. Select Single Object as the **Source Type**.
   c. Click **Select** to specify a flat file that contains the source fields.
   The following image shows the flat file source details:



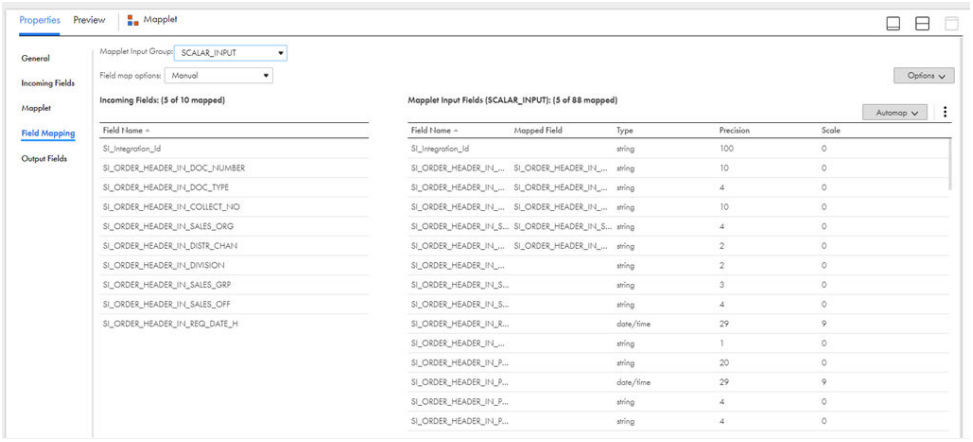
6. Add order header, order items, and order partners as flat file source objects for the mapping.
7. To add a BAPI/RFC mapplet transformation, on the **Transformation palette**, click Mapplet.
   a. On the **General** tab, enter a name and description for the mapplet.
   b. Draw a link to connect the flat file Source transformation to the Mapplet transformation.
   c. On the **Mapplet** tab, click **Select**.
   The **Select Mapplet** dialog box appears.
   d. Select the `bapi_salesorder_createfromdat1` mapplet and click **OK**.
   e. Click **Connection** to specify an SAP RFC/BAPI Interface connection.
   The following image shows the bapi_salesorder_createfromdat1 mapplet details:



   f. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.

g. Click **Field Mapping** and map the incoming source fields with the appropriate mapplet input fields.

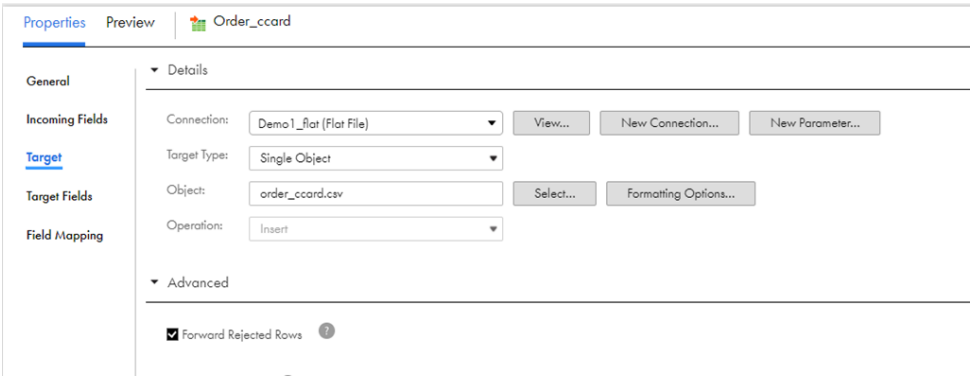
The following image shows the field mapping of the incoming source fields with the mapplet input fields:



8. To add a flat file target, on the **Transformation palette**, click **Target**.
9. On the **General** tab, enter a name and description for the target.
10. Click the **Target** tab and configure target details.
    a. Select a flat file connection for the target.
    b. Select Single Object as the **Source Type**.
    c. Click **Select** to specify the target object.

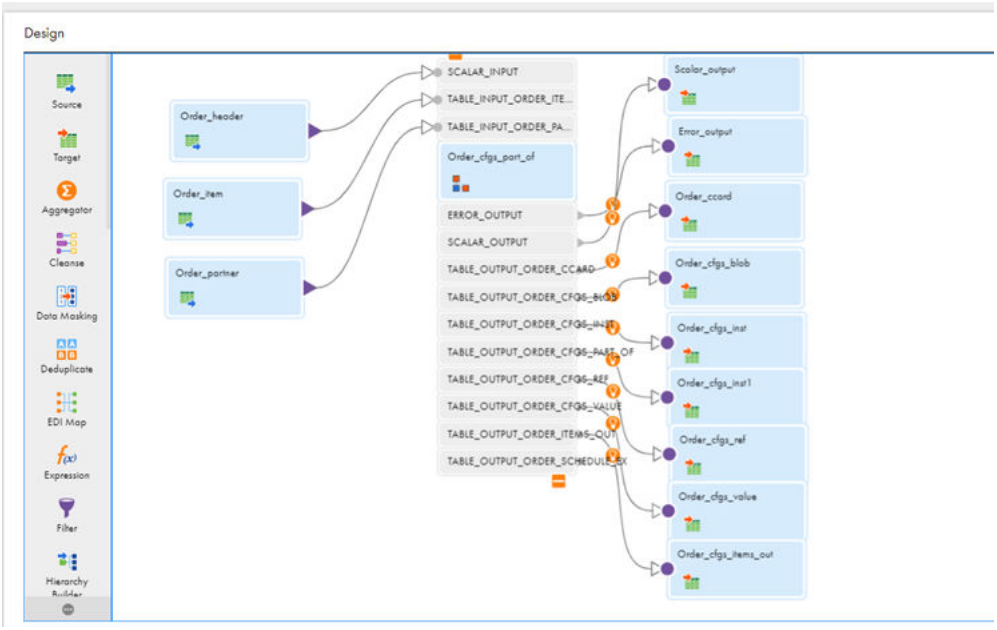
    The **Target Object** dialog box appears.
    d. Select the flat file object, and click **OK**.
    e. Optionally, you can choose to forward rejected rows in the Advanced Target options.
    f. Repeat the above steps to add all the flat file target objects.

    The following image shows the target transformation details:



11. Draw a link to connect the output ports in the Mapplet transformation to the flat file Target transformation.

    The following image shows the mapping configured for the bapi_salesorder_createfromdat1 mapplet:



12. Save and run the mapping.

SAP mapplet properties

You can commit data, cache data, and handle errors for the target in an SAP BAPI/RFC mapping task.

To configure the SAP mapplet properties for the target, click the **Runtime Options** tab of the SAP BAPI/RFC mapping task. Expand the **SAP Mapplet Properties** section, and configure the mapplet properties.

The following table describes the commit properties you can configure in an SAP BAPI/RFC mapping task:

Property	Description
Perform Commit	Issues a commit with a BAPI commit function. If you disable this option, Data Integration does not call a commit function. It also ignores the commit interval and does not wait for a commit to continue to run the mapping task. Default is enabled.
Perform Commit Wait	Waits until the commit is complete to continue to run the mapping task. SAP sends a return call to Data Integration when the commit is complete. If you disable this option, Data Integration continues to process data during the commit. Default is enabled.

Property	Description
Commit Interval	Number of BAPI/RFC calls after which Data Integration issues a commit. To issue a commit, Data Integration makes a BAPI/RFC commit call. If you disable the **Perform Commit** option, Data Integration ignores this value. Default is 1000.
Commit Function	Determines how Data Integration makes a BAPI/RFC call when it issues a commit. Data Integration makes the call either from outside SAP or from within BAPI based on the following options you select: - ABAP4_COMMIT_WORK. Makes a BAPI/RFC call from ABAP and then issues a commit. - BAPI_TRANSACTION_COMMIT. Makes call from outside SAP and then issues a commit. If you disable the **Perform Commit** option, Data Integration ignores this value. Default is BAPI_TRANSACTION_COMMIT.

The following table describes the data caching properties you can configure in an SAP BAPI/RFC mapping task:

Property	Description
Cache Directory	Default directory used to cache BAPI data. By default, the cache files are created in a directory specified by the variable $PMCacheDir. If you override the directory, make sure the directory exists and contains enough disk space for the cache files.
Cache Size	Total memory in bytes allocated to Data Integration to cache data prepared for BAPI/RFC mapplets. Default is 10 MB.

The following table describes the error handling properties you can configure in an SAP BAPI/RFC mapping task:

Property	Description
Duplicate Parent Row Handling	Determines how Data Integration handles duplicate parent rows in a mapping task. Select one of the following options: - First Row. Passes the first duplicate row to the target. Data Integration rejects rows with the same primary key that it processes after this row. - Last Row. Passes the last duplicate row to the target. - Error. Data Integration passes the first row to the target. Rows that follow with duplicate primary keys increment the error count. The mapping task fails when the error count exceeds the error threshold. Default is First Row.
Orphan Row Handling	Determines how Data Integration handles orphan rows in a mapping task. Select one of the following options: - Ignore. Ignores orphan rows. - Error. The mapping task fails when the error count exceeds the error threshold. Default is Ignore.
Continue on Error	Determines if you want to continue to run the mapping task when a BAPI/RFC error is encountered. Default is enabled.

Part V: Data integration using IDocs

This part contains the following chapters:

CHAPTER 9

Data integration using IDocs

You can generate an IDoc mapplet to Data Integration. You can then use the mapplet in an outbound mapping to read IDocs from SAP or inbound mapping to write IDocs to SAP.

IDocs electronically exchange data between SAP applications or between SAP applications and external programs. IDoc is a message-based integration interface that processes data asynchronously.

IDoc is a component of Application Link Enabling (ALE) module within SAP that can send and receive Intermediate Documents (IDocs) over RFC protocol.

ALE Layers

The message-based architecture of ALE comprises three layers:

- Application layer that provides ALE an interface to SAP to send or receive messages from external systems.
- Distribution layer that filters and converts messages to ensure that they are compatible between different SAP releases.
- Communications layer that enables ALE to support synchronous and asynchronous communication. You use IDocs for asynchronous communication.

The architecture of ALE provides a way to send IDocs as text files without connecting to a central database. Applications can communicate with each other without converting between formats to accommodate hardware or platform differences.

IDoc Record Types

IDocs contain three record types:

- Control record, which identifies the message type.
- Data records that contain the IDoc data in segments.
- Status records that describe the status of the IDoc. Status record names are the same for each IDoc type.

ALE Components

ALE has the following components:

Component	Description
Logical System	All systems that need to communicate using ALE/IDoc must be setup as a Logical System within SAP. An SAP administrator can setup logical systems in transaction BD54.
Distribution Model	Defines an agreement between two logical systems on the messages that can be exchanged and identifies the sender and the receiver. An SAP administrator can setup distribution models in transaction BD64.

Component	Description
Partner Profile	Stores the IDoc type and processing logic related to the distribution model. An SAP administrator can setup partner profiles in transaction WE20.
RFC Destination	Defines the protocol and access to the logical system. An SAP administrator can setup RFC destinations in transaction SM59.
Message Type	Representation of a business object.
IDoc Type	Representation of a message type. SAP uses IDoc types to support backward compatibility across various SAP releases.
IDoc	An instance of an IDoc type that contains business data.

CHAPTER 10

IDoc mapplets

You can import an IDoc as a mapplet using the **Mapplet - SAP IDoc** asset. An IDoc contains a hierarchical structure consisting of segments. Each segment is an SAP structure defined in the SAP system.

An IDoc has header and data record components. The header component contains control information, such as creation date and status. The control information is in an SAP structure called EDIDC. The data records are in an SAP structure called EDIDD.

Segments and groups

An IDoc is a hierarchical structure that contains segments. A segment can be a parent or child. A child segment depends on another segment. A parent segment contains child segments. A parent segment can be a child of another segment.

IDoc segments are organized into groups. The following rules determine the group to which a segment belongs:

- A parent segment starts a new group. For example, in the MATMAS04 IDoc, the E1MARCM segment contains a child and therefore starts a group.
- A child segment that is not a parent belongs to the group that is started by its immediate parent. For example, in the MATMAS04 IDoc, the E1MARA1 segment does not contain a child and therefore belongs to the group of its parent E1MARAM.
- A group can also be a parent or a child.

IDocs properties

When you fetch an IDoc in **Mapplet - SAP IDoc**, you can view all the IDoc segments. Select a segment to view the fields in the segment.

IDocs Properties

The following table describes the IDocs properties that you can view and specify in the **Mapplet - SAP IDoc** asset:

Property	Description
Transformation Type	Indicates the type of transformation you want to use to generate the IDoc mapplet. Select one of the following options: - IDoc Prepare. Select to generate a mapplet that writes source data as an IDoc message. - IDoc Interpreter. Select to generate a mapplet that reads IDoc messages.
Source	The source by which you fetch the IDoc. Select one of the following options: - Connection. Select to fetch the IDoc with an SAP Table Connector connection using the **Saprfc.ini Path** and **Destination** advanced properties. For more information about how to create an SAP Table connection, see the *SAP Table Connector* documentation. - Local file. Select to fetch the IDoc by using a file from your local drive. For more information about importing IDoc metadata for an SAP/ALE IDoc transformation from file, see "Generating IDoc metadata to local file" on page 71.
Transformation Scope	Indicates how the Secure Agent applies the transformation logic to incoming data. Select one of the following options: - Transaction. Select Transaction to apply the transformation logic to all rows in a transaction. Select Transaction when the results of the transformation depend on all rows in the same transaction, but not on rows in other transactions. When you select Transaction, associated mappings can run in real time. - All Input. Select All Input to apply the transformation logic to all incoming data, and to drop the incoming transaction boundaries. Select All Input when the results of the transformation depend on all rows of data in the source. Default is **All Input**.
Message Type	The type of application messages that classify categories of data. For example, ORDERS and MATMAS (Material Master).
Basic IDoc Type	The type of data structure associated with the message type. For example, MATMAS01, MATMAS02 for MATMAS. IDocs contain the data associated with the message type.
Extended Type	Extended IDoc type name. IDoc extension is extension of basic type and contains additional custom IDoc segments and fields that are not available in the standard basic type.
IDoc Document	Displays the IDoc segments associated with the message type and IDoc type.
Segment Name	Segment names of the IDoc type.
Description	Description of the segments.
Min Occurs	Minimum number of occurrences of the segment in an IDoc.

Property	Description
Max Occurs	Maximum number of occurrences of the segment in an IDoc.
Select	Selects the data segments to include in the transformation. When you select a segment, the parent segments and all required child segments are also selected. When you clear a segment, all child segments are also cleared.
Control Record	Displays the control record segments in the IDoc, their values, and precision. The Designer provides values for some of the segments when you create a mapplet using the IDoc prepare transformation. You can also provide values for other segments.

Segment Definition

You can view the segment field details from the segment definition section. To view the field names of the segment, select a segment name.

The following table describes the segment field details:

Field	Description
Name	Field name of a segment.
SAP Datatype	SAP data type of the field.
Precision	Precision of the field.
Scale	Scale of the field.
Description	Description of the field.

Generating IDoc metadata to local file

You can import IDoc metadata for an SAP/ALE IDoc transformation from a local file.

To import IDoc metadata from a local file, run the RSEIDoc3 program from the SAP client to generate the metadata. When you run the program, select the IDoc type and range for the IDoc metadata you want to generate. The program exports the metadata it generates into a metadata file. For example, you can export the metadata to a file with the *.idc* extension. You can then use the metadata file to import the metadata into the Designer for use in the SAP/ALE IDoc transformation.

To generate IDoc metadata using the RSEIDoc3 program:

1. Enter transaction se38 from the SAP client.
2. Execute the RSEIDoc3 program.
3. Select the basic IDoc type and range.
4. If you are using SAP version 5.0 or later, click **Control Record**.
5. Optionally, select extended IDoc type and range, if available.
6. Optionally, select extended grammar, if available.
7. Click **Parser**.
8. Click **System** > **List** > **Save** > **Local File**.
9. On the Save List in File dialog box, select **Unconverted**.

10. Enter the path and file name where you want to save the metadata file, and then save the file with the .idc extension.

Outbound mapplet

You can capture changes to the master data or transactional data in the SAP application database in real time.

When data in the application database changes, the SAP system creates IDocs to capture the changes and sends the IDocs to Data Integration. You can use the IDoc Reader connection to read the IDoc message in real time as they are generated by the SAP system.

If the Secure Agent is not running when the SAP system sends outbound IDocs, the Secure Agent does not receive the IDocs. However, the SAP system stores the outbound IDocs in EDI tables, which are a staging area for guaranteed message delivery. You can configure the SAP system to resend the IDocs by configuring the tRFC port used to communicate with the Secure Agent. When you configure the port, you can enable background processes in SAP that try to resend the IDocs to the Secure Agent a set number of times.

To generate the outbound mapplet to read IDoc messages from SAP system, use the IDoc Interpreter when you import the IDoc metadata. Import the outbound mapplet to Data Integration and configure an outbound mapping.

Outbound mapplet ports

An outbound IDoc mapplet contains predefined ports. You cannot edit the ports.

The following table describes the mapplet ports:

Port Name	Description
Basic IDoc Type	Basic IDoc type name. Basic IDoc type defines the structure of an IDoc. Each basic type describes standard IDoc segments, format of data fields, and size. Basic type contains all the standard fields that are necessary for carrying out a business transaction.
Extended IDoc Type	Extended IDoc type name. IDoc extension is extension of basic type and contains additional custom IDoc segments and fields that are not available in the standard basic type.
IDocRecord	IDoc message data.
DocumentNumber	Unique message number of the IDoc.

Target object for outbound mapplet error output

You can configure an outbound IDoc mapping to write IDocs that are not valid to a relational or flat file target. To write IDocs that are not valid to a relational or flat file target, connect the IDoc_Interpreter_Error_Output_Group port in the outbound mapplet to a relational or flat file target object. You must also configure the error log type session property in the **Schedule** page. Based on your requirement, you can choose to configure multiple target objects for each segment in the IDoc and for the IDoc_Interpreter_Error_Output_Group.

Inbound mapplet

You can synchronize transactional data in a legacy application with the data in the SAP application database.

Use an inbound SAP IDoc mapping to send the transactional data from the legacy application database to the SAP system. Data Integration extracts the data from the legacy application data source, prepares the data in SAP IDoc format, and sends the data to the SAP system as inbound IDocs using ALE. You can use the IDoc Writer connection to write inbound SAP IDoc messages to SAP systems.

To generate the inbound mapplet to write IDocs to SAP systems, use the Prepare transformations when you import the IDoc metadata. Import the inbound mapplet to Data Integration and configure an inbound mapping.

Key fields and control record fields

An IDoc mapplet includes a primary key (GPK) and a foreign key (GFK) in each segment. When you configure a mapping, integration template, or mapping task, map the primary key field, foreign key field, and any control record fields.

Note: You can enable control record fields when you import the IDoc metadata.

The Prepare transformation in the SAP IDoc Writer mapplet can have primary key and foreign key fields and other input fields, including control record fields.

The Prepare transformation has the following output fields:

- IDoc Data. Map this field to an IDoc target.
- Error IDoc Data. Map this field to see error messages about IDoc syntax/data conversion.

When you import IDoc metadata, you can add fields to an **IDoc Prepare** transformation. In **Mapplet - SAP IDoc**, on the **Control Record** tab, you can add the control record keys as key-value pairs. For example, you can add the following key-value pairs for the sender partner type and the sender partner number:

- Key: SNDPRT and Value: LS for Logical System
- Key: SNDPRN and Value ICS

In an inbound mapping, you can pass the sender partner number to SAP. You can pass a value to the CONTROL_SNDPRN port in the control input group of the Prepare transformation. If you do not connect this port to an upstream transformation, the Secure Agent uses the partner number value of SNDPRN key you specify in the **Mapplet - SAP IDoc** asset.

IDoc primary and foreign keys

An IDoc message is organized hierarchically with one top-level parent segment and one or more second-level child segments. Second-level child segments can also have one or more third-level child segments.

To maintain the structure of the IDoc data, the Prepare transformation in the SAP IDoc Writer mapplet uses primary and foreign keys. The top-level parent segment has a primary key. Each child segment has a primary key and a foreign key. The foreign key of each child segment references the primary key of its parent segment. For example, the foreign key of a second-level child segment references the primary key of the top-level parent segment. Similarly, the foreign key of a third-level child segment references the primary key of the second-level child segment.

The Prepare transformation groups incoming IDoc data based on the values in the primary and foreign key fields. The Control Input group of the Prepare transformation represents the parent segment. All other groups of the Prepare transformation except the ErrorIDocData group represent second-level or third-level child segments. The ErrorIDocData group is used for processing invalid IDocs.

The following table shows the groups of the Prepare transformation and the fields used for the primary and foreign keys:

Groups	Field	Description
Control Input Group	GPK_DOCNUM	Primary key of the parent segment.
Child Segment 1	GPK_<Child1_name>	Primary key of Child Segment 1.
Child Segment 1	GFK_DOCNUM_<Child1_name>	Foreign key of Child Segment 1 references the primary key of the parent segment.
Child Segment A of Child Segment 1	GPK_<Child1A_name>	Primary key of Child Segment A of Child Segment 1.
Child Segment A of Child Segment 1	GFK_<Child1_name>_<Child1A_name>	Foreign key of Child Segment A of Child Segment 1 references the primary key of Child Segment 1.
Child Segment 2	GPK_<Child2_name>	Primary key of the IDoc child segment.
Child Segment 2	GFK_DOCNUM_<Child2_name>	Foreign key of Child Segment 2 references the primary key of the parent segment.
Child Segment B of Child Segment 2	GPK_<Child2B_name>	Primary key of Child Segment B of Child Segment 2.
Child Segment B of Child Segment 2	GFK_<Child2_name>_<Child2B_name>	Foreign key of Child Segment B of Child Segment 2 references the primary key of Child Segment 2.

Each value for the GPK_<name> field needs to be unique. Each GFK_<parent_name>_<group_name> field needs to reference the primary key of its parent segment.

For example, the following table shows the relationship of primary and foreign keys in an IDoc message named ABSEN1 with four child segments:

Group	Field	Primary/Foreign Keys
CONTROL_INPUT_ABSEN1	GPK_DOCNUM	P1
E2ABSE1	GPK_E2ABSE1	C1
-	GFK_DOCNUM_E2ABSE1	P1
E2ABSE2	GPK_E2ABSE2	C2
-	GFK_DOCNUM_E2ABSE2	P1
E2ABSE2A	GPK_E2ABSE2A	C2A
-	GFK_E2ABSE2_E2ABSE2A	C2
E2ABSE3	GPK_E2ABSE3	C3
-	GFK_DOCNUM_E2ABSE3	P1
E2ABSE3B	GPK_E2ABSE3B	C3B

Group	Field	Primary/Foreign Keys
-	GFK_E2ABSE2_E2ABSE2A	C3
E2ABSE4	GPK_E2ABSE4	C4
-	GFK_DOCNUM_E2ABSE4	P1

The Prepare transformation uses these primary and foreign key relationships to maintain the structure of the IDoc data. Any foreign key field that does not match the primary key of its parent segment results in an orphan row. Any primary key field that is not unique results in a duplicate row.

Verify that each IDoc message has a unique primary key for the top-level parent segment, each child segment, and that each foreign key matches the primary key of its parent.

Configuring IDoc mapplets

You can configure IDoc mapplets using the **Mapplet - SAP IDoc** asset from Data Integration. You can use IDoc mapplets to the inbound or outbound mappings.

1. In Data Integration, click **New** > **Mapplets** > **Mapplet - SAP IDoc**, and then click **Create**.
2. Enter a unique name for the mapplet.
3. Specify the location where you save the mapplet.
4. Optionally, enter a description for the mapplet.
5. Select the transformation type. You can select one of the following options:
   - Select **IDoc Interpreter** to generate outbound mappings to read IDocs from the SAP system.
   - Select **IDoc Prepare** to generate inbound mappings to write IDocs to the SAP system.
6. Select the source by which you fetch the IDoc. You can select one of the following options:
   - Select **Connection** to fetch the IDoc with an SAP Table Connector connection using the **Saprfc.ini Path** and **Destination** advanced properties.

     Note: Ensure that you add the following entries in the `sapnwrfc.ini` file when you create the SAP Table connection:

     ```
     DEST=PRD
     ASHOST=sapr3
     SYSNR=01
     ```

     The SAP Table connection information is available in the SAP Table Connector documentation. For more information about the SAP Table connection, see the *SAP Table Connector* documentation.
   - Select **Local File** to fetch the IDoc using a file from your local drive.
7. Select the scope of the transformation. You can select one of the following options:
   - Select **Transaction** to apply the transformation logic to all rows in a transaction. Select **Transaction** when the results of the transformation depend on all rows in the same transaction, but not on rows in other transactions. When you select **Transaction**, associated mappings can run in real time. For outbound mappings, select **Transaction**.

- Select **All Input** to apply the transformation logic to all incoming data, and to drop the incoming transaction boundaries. Select **All Input** when the results of the transformation depend on all rows of data in the source.

8. Enter the message type and the basic IDoc type, and then click **Fetch IDoc**.

   The segment details of the IDoc appears.

9. On the **IDoc Document** tab, in the **Select IDoc Segments** section, select the segments you want to include in the mapplet.
10. Optionally, in an inbound mapping, to add a control record field to the mapplet, on the **Control Record** tab, select the checkbox for the field you want to add. This enables you to map the selected control record fields when you configure a mapping, integration template, or mapping task.
11. Click **Save**.

CHAPTER 11

Mapping and mapping tasks using IDocs

To read and write IDocs, Data Integration integrates with SAP applications using Application Link Enabling (ALE).

ALE is an SAP proprietary technology that enables data communication between SAP systems. ALE also enables data communication between SAP and external systems.

You can configure outbound mappings to read IDocs from SAP and inbound mappings to write IDocs to SAP.

In advanced mode, the Mapping Designer updates the mapping canvas to include transformations and functions that enable advanced functionality.

To configure an outbound mapping to read IDocs from SAP, perform the following tasks:

1. Configure an IDoc mapplet using the **Mapplet - SAP IDoc** asset from Data Integration. Verify that you selected the segments and groups you want to include in the IDoc. In addition, select the **IDoc Interpreter** transformation.
2. Configure an outbound mapping using the generated IDoc mapplet. Add a Source transformation to read data from the SAP system, add the generated IDoc mapplet, and add a Target transformation to write the IDoc in the target object.

To configure an inbound mapping to write IDocs to SAP, perform the following tasks:

1. Create an IDoc mapplet using the **Mapplet - SAP IDoc** asset from Data Integration. Verify that you selected the segments and groups you want to include in the IDoc. In addition, select the **IDoc Prepare** transformation.

   Note: Do not configure multiple IDoc Prepare transformations for an IDoc mapplet in an inbound mapping on Linux. Otherwise, the mapping fails with the DTM error.
2. Configure an inbound mapping using the generated IDoc mapplet. Add a Source transformation to read data from the source system, add the generated IDoc mapplet, and add a Target transformation to write the IDoc to the SAP system.

IDoc reader sources in mappings

To read IDocs from an SAP application, use an SAP IDoc Reader connection and configure the IDoc Reader source properties in the Source transformation in a mapping.

Specify the name and description of the IDoc Reader source. Configure the source and advanced properties for the source object.

The following table describes the IDoc Reader source properties that you can configure in a Source transformation:

Property	Description
Connection	Name of the source connection.
Source Type	Source type. Select **Single** for a single source object. When you select an SAP IDoc Reader as connection, the source type can be a single object and source object is the IDoc Reader Object. The source object has the generic structure of an IDoc message.
Object	Source object.

The following table describes the SAP IDoc Reader advanced source properties:

Property	Description
Idle Time	Indicates the number of seconds the Secure Agent waits for IDocs to arrive before it stops reading from the SAP source. For example, if you enter 30 seconds for idle time, the Secure Agent waits 30 seconds after reading from the SAP source. If no new IDocs arrive within 30 seconds, the Secure Agent stops reading from the SAP source. Default is 300.
Packet Count	Controls the number of packets the Secure Agent reads from SAP before stopping. For example, if you enter 10 for Packet Count, the Secure Agent reads the first 10 packets from the SAP source and then stops. The packet Size property in the ALE configuration determines the number of IDocs the Secure Agent receives in a packet. If you enter packet count as -1, you can read infinite number of packets. Default is -1.
Realtime Flush Latency	Determines, in seconds, how often the Secure Agent flushes data from the source.
Reader Time Limit	Sets a period of time, in seconds, during which the Secure Agent reads IDocs from the SAP source. For example, if you specify 10 as the reader time limit, the Secure Agent stops reading from the SAP source after 10 seconds. If you enter reader time limit as 0, the Security Agent continues to read IDocs from SAP for an infinite period of time. Default is 0.
Recovery Cache Folder	Specifies the location of the recovery cache folder. **Note:** This property is not applicable when you configure the SAP IDoc Reader Source transformation.
Number of Retries for Connection Resiliency	Determines the number of times Cloud Data Integration to attempt to connect to the SAP system. Default is 1.
Delay between Retries for Connection Resiliency	Determines the time interval in seconds between the connection retries to connect to the SAP system if it could not connect on a previous attempt. Default is 5.
Tracing Level	Sets the amount of detail that appears in the log file. You can choose terse, normal, verbose initialization or verbose data. Default is normal.

IDoc writer targets in mappings

You can use an SAP IDoc Writer connection and configure the IDoc Writer target properties in the Target transformation in a mapping.

Specify the name and description of the IDoc Writer target. Configure the target and advanced properties for the target object in a Target transformation.

The following table describes the IDoc Writer target properties that you can configure in a Target transformation:

Property	Description
Connection	Name of the target connection.
Target Type	Type of the target object.
Object	Name of the target object.

The following table describes the SAP IDoc Writer advanced target properties:

Property	Description
Packet Size	Number of IDocs that you want to send in a packet to SAP.
Number Of Retries	Not applicable.
Delay Between Retries	Not applicable.
Send IDocs Based On	The commit method of IDocs to SAP based on the value you set for the **Packet Size** property or commit properties at every commit point. Select one of the following options: - Packet Size. Commits IDocs to SAP based on the value you set for the **Packet Size** property. Data Integration collects IDoc messages until the total count reaches the packet size. It then sends messages as a packet to SAP. - Commit Call. Commits IDocs to SAP based on the commit properties at every commit point.
Generate Request ID	Not applicable.
Forward Rejected Rows	Determines whether the transformation passes rejected rows to the next transformation or drops rejected rows. By default, the agent forwards rejected rows to the next transformation.

Configuring IDoc mapplets

You can configure IDoc mapplets using the **Mapplet - SAP IDoc** asset from Data Integration. You can use IDoc mapplets to the inbound or outbound mappings.

1. In Data Integration, click **New** > **Mapplets** > **Mapplet - SAP IDoc**, and then click **Create**.
2. Enter a unique name for the mapplet.
3. Specify the location where you save the mapplet.

4. Optionally, enter a description for the mapplet.
5. Select the transformation type. You can select one of the following options:
   - Select **IDoc Interpreter** to generate outbound mappings to read IDocs from the SAP system.
   - Select **IDoc Prepare** to generate inbound mappings to write IDocs to the SAP system.
6. Select the source by which you fetch the IDoc. You can select one of the following options:
   - Select **Connection** to fetch the IDoc with an SAP Table Connector connection using the **Saprfc.ini Path** and **Destination** advanced properties.

     Note: Ensure that you add the following entries in the `sapnwrfc.ini` file when you create the SAP Table connection:

     ```
     DEST=PRD
     ASHOST=sapr3
     SYSNR=01
     ```

     The SAP Table connection information is available in the SAP Table Connector documentation. For more information about the SAP Table connection, see the *SAP Table Connector* documentation.
   - Select **Local File** to fetch the IDoc using a file from your local drive.
7. Select the scope of the transformation. You can select one of the following options:
   - Select **Transaction** to apply the transformation logic to all rows in a transaction. Select **Transaction** when the results of the transformation depend on all rows in the same transaction, but not on rows in other transactions. When you select **Transaction**, associated mappings can run in real time. For outbound mappings, select **Transaction**.
   - Select **All Input** to apply the transformation logic to all incoming data, and to drop the incoming transaction boundaries. Select **All Input** when the results of the transformation depend on all rows of data in the source.
8. Enter the message type and the basic IDoc type, and then click **Fetch IDoc**.

   The segment details of the IDoc appears.
9. On the **IDoc Document** tab, in the **Select IDoc Segments** section, select the segments you want to include in the mapplet.
10. Optionally, in an inbound mapping, to add a control record field to the mapplet, on the **Control Record** tab, select the checkbox for the field you want to add. This enables you to map the selected control record fields when you configure a mapping, integration template, or mapping task.
11. Click **Save**.

Configuring an outbound mapping to read IDocs from SAP

You can configure an outbound mapping to read IDocs from SAP.

1. To create a mapping, click **Data Integration** > **New** > **Mappings**. Select **Mapping** and click **Create**.
2. Enter a name and description for the mapping, and click **OK**.

   You can use alphanumeric characters and underscores (_) in the mapping name.
3. To configure an SAP source, on the **Transformation** palette, click **Source**.
4. In the **Properties** panel, on the **General** tab, enter a name and description.

5. Click the **Source** tab and select an SAP IDoc Reader connection.

   When you select an SAP IDoc Reader as connection, the source type is a single object and source object is the IDoc Reader Object.

   You can also parameterize the SAP IDoc Reader connection.
6. If required, configure the advanced source properties.
7. To add an IDoc mapplet transformation, on the **Transformation palette**, click Mapplet.
   a. On the **General** tab, enter a name and description for the mapplet.
   b. Draw a link to connect the previous transformation to the transformation.
   c. On the **Mapplet** tab, click **Select**.

      The **Select Mapplet** dialog box appears.
   d. Specify an IDoc mapplet that you imported using the Interpreter transformation and click **OK**.
8. To add any other transformation, on the **Transformation palette**, click the transformation name. Or, drag the transformation onto the mapping canvas.
   a. On the **General** tab, enter a name and description for the transformation.
   b. Draw a link to connect the previous transformation to the transformation.

      When you link transformations, the downstream transformation inherits the incoming fields from the previous transformation.
      For a Joiner transformation, draw a master link and a detail link.
   c. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.
   d. Configure additional transformation properties, as needed.

      The properties that you configure vary based on the type of transformation that you create.
   e. To add another transformation, repeat these steps.
9. To add a Target transformation, on the **Transformation palette**, click **Target**.
   a. On the **General** tab, enter a name and description.
   b. Draw a link to connect the previous transformation to the Target transformation.
   c. Click the **Target** tab and configure target details. If required, configure the advanced target properties.

      Target details and advanced target properties appear based on the connection type.
   d. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.
   e. Click **Field Mapping** and map the fields that you want to write to the target.
   f. To add another Target transformation, repeat these steps.
10. Save and run the mapping. or save and create a mapping task.

For information about the source and target transformations, see *Transformations*.

Configuring an inbound mapping to write IDocs to SAP

You can configure an inbound mapping to write IDocs to SAP.

1. To create a mapping, click **Data Integration** > **New** > **Mappings**. Select **Mapping** and click **Create**.

2. Enter a name and description for the mapping, and click **OK**.

   You can use alphanumeric characters and underscores (_) in the mapping name.
3. To configure a source, on the **Transformation** palette, click **Source**.
4. In the **Properties** panel, on the **General** tab, enter a name and description.
5. Click the **Source** tab and configure the source details.
6. To add an IDoc mapplet transformation, on the **Transformation palette**, click Mapplet.
   a. On the **General** tab, enter a name and description for the mapplet.
   b. Draw a link to connect the previous transformation to the transformation.
   c. On the **Mapplet** tab, click **Select**.

      The **Select Mapplet** dialog box appears.
   d. Specify an IDoc mapplet that you imported using the Prepare transformation and click **OK**.
   e. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.

      Note: You must link the DOCNUM port of the mapplet to the source transformation. The DOCNUM port represents a unique number for each IDoc and the SAP system does not accept inbound IDocs without a unique document number.
7. To add any other transformation, on the **Transformation palette**, click the transformation name. Or, drag the transformation onto the mapping canvas.
   a. On the **General** tab, enter a name and description for the transformation.
   b. Draw a link to connect the previous transformation to the transformation.

      When you link transformations, the downstream transformation inherits the incoming fields from the previous transformation.
      For a Joiner transformation, draw a master link and a detail link.
   c. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.
   d. Configure additional transformation properties, as needed.

      The properties that you configure vary based on the type of transformation that you create.
   e. To add another transformation, repeat these steps.
8. To add a Target transformation, on the **Transformation palette**, click **Target**.
   a. On the **General** tab, enter a name and description.
   b. Draw a link to connect the previous transformation to the Target transformation.
   c. Click the **Target** tab and configure SAP target details.
   d. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.
   e. Click **Field Mapping** and map the fields that you want to write to the target.
   f. To add another Target transformation, repeat these steps.
9. Save and run the mapping or save and create a mapping task.

Outbound mapping to read MATMAS IDocs from SAP example

You can read material master (MATMAS) IDocs from SAP and write it to a relational database object.

In this example, to read the MATMAS IDocs, perform the following steps:

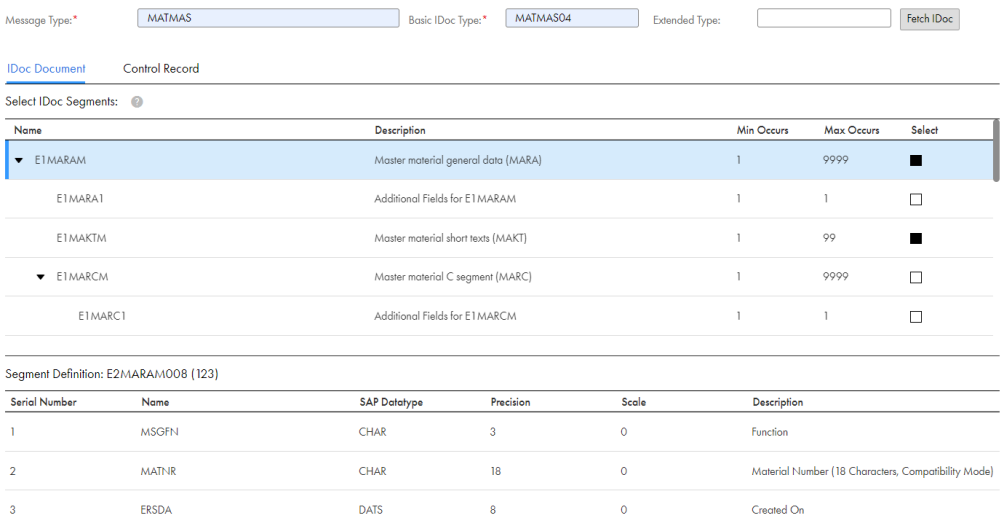
1. Configure an IDoc mapplet with the MATMAS IDoc.
2. Configure a mapping using the generated mapplet.

Step 1: Configuring an IDoc mapplet with the MATMAS IDoc

Configure an IDoc mapplet with the MATMAS IDoc from Data Integration. Use the configured mapplet to an outbound mapping.

1. In Data Integration, click **New** > **Mapplets** > **Mapplet - SAP IDoc**, and then click **Create**.
2. Enter a unique name for the mapplet.
3. Specify the location where you save the mapplet.
4. Optionally, enter a description for the mapplet.
5. Select **IDoc Interpreter** as the transformation type.
6. From the **Connection** list, select the source connection by which you fetch the IDoc.
7. From the **Transformation Scope** list, select **Transaction**.
8. In the **Message Type** field, enter MATMAS.
9. In the **Basic IDoc Type** field, enter MATMAS04.
10. Click **Fetch IDoc**.

    The segment details of the IDoc appears in the **Select IDoc Segments** section.



11. On the **IDoc Document** tab, in the **Select IDoc Segments** section, select the segments you want to include in the mapplet.
12. Click **Save**.

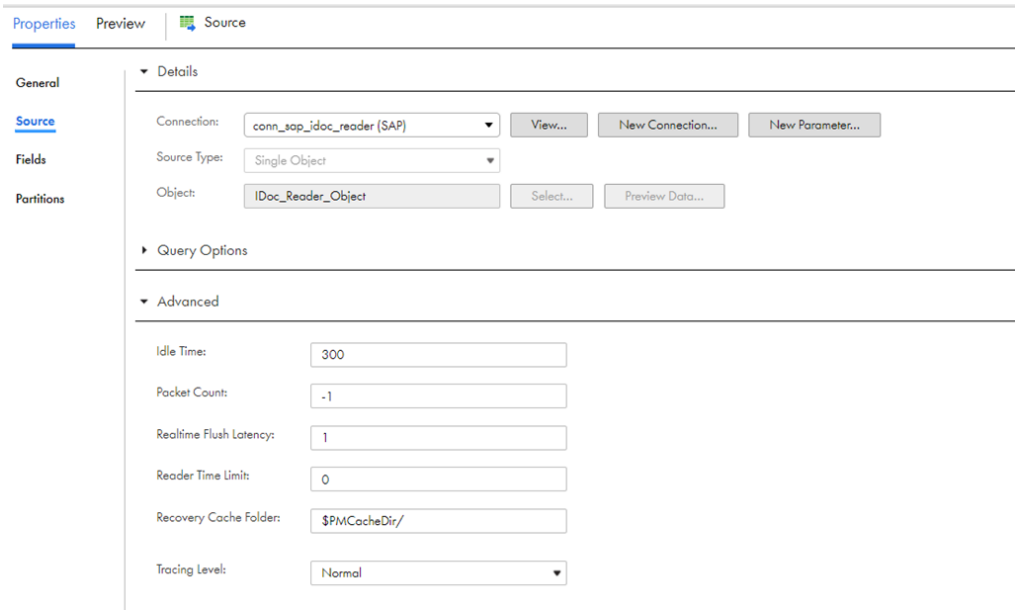
Step 2: Configuring an outbound mapping with the MATMAS IDoc

Configure an outbound mapping that uses the IDoc mapplet to read the MATMAS IDoc from SAP.

1. To create a mapping, click **Data Integration** > **New** > **Mappings**. Select **Mapping** and click **Create**.
2. Enter a name and description for the mapping, and click **OK**.
3. To configure an SAP source, on the **Transformation** palette, click **Source**.
4. In the **Properties** panel, on the **General** tab, enter a name and description.
5. Click the **Source** tab and select an SAP IDoc Reader connection.

   When you select an SAP IDoc Reader as connection, the source type is a single object and source object is the IDoc Reader Object.
6. If required, configure the advanced source properties.

   The following image shows the SAP source details:

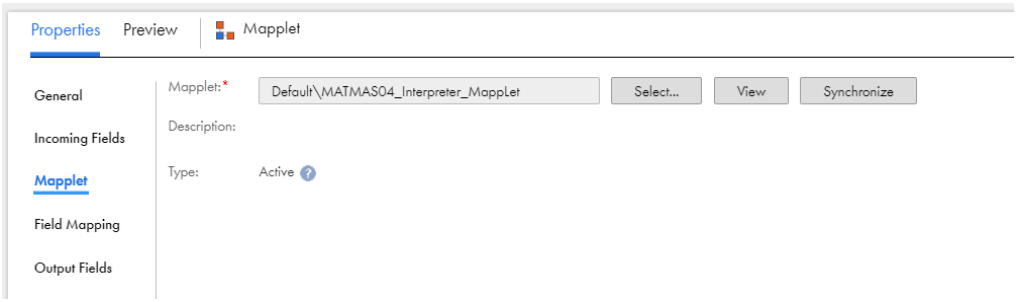


7. To add an IDoc mapplet transformation, on the **Transformation palette**, click Mapplet.
8. On the **General** tab, enter a name and description for the mapplet.
9. Draw a link to connect the Source transformation to the Mapplet transformation.
10. On the **Mapplet** tab, click **Select**.

    The **Select Mapplet** dialog box appears.

11. Specify an IDoc mapplet that you imported using the Interpreter transformation and click **OK**.

    The following image shows the MATMAS mapplet details:



12. To preview the incoming single IDoc Reader Object, click **Incoming Fields**.

    The following image shows the incoming single IDoc Reader object:

13. Click **Field Mappings** to map the incoming IDoc Record field with the IDocData field in the mapplet.
14. To add a Target transformation, on the **Transformation palette**, click **Target**.
15. On the **General** tab, enter a name and description.
16. Click the **Target** tab, and configure the required target database details.

    Note: When you use IDoc Listener as a source, you can specify only those targets that honor commit. Contact Global Customer Support for the list of supported targets.
17. Draw a link to connect the mapplet groups to the target database object.
18. Save and run the mapping.

Inbound mapping to write IDocs to SAP example

You can create material master (MATMAS) IDocs in SAP using the MATMAS IDoc type.

In this example, to write MATMAS IDoc to SAP, perform the following steps:

1. Configure an IDoc mapplet with the MATMAS IDoc.
2. Configure a mapping using the generated mapplet.

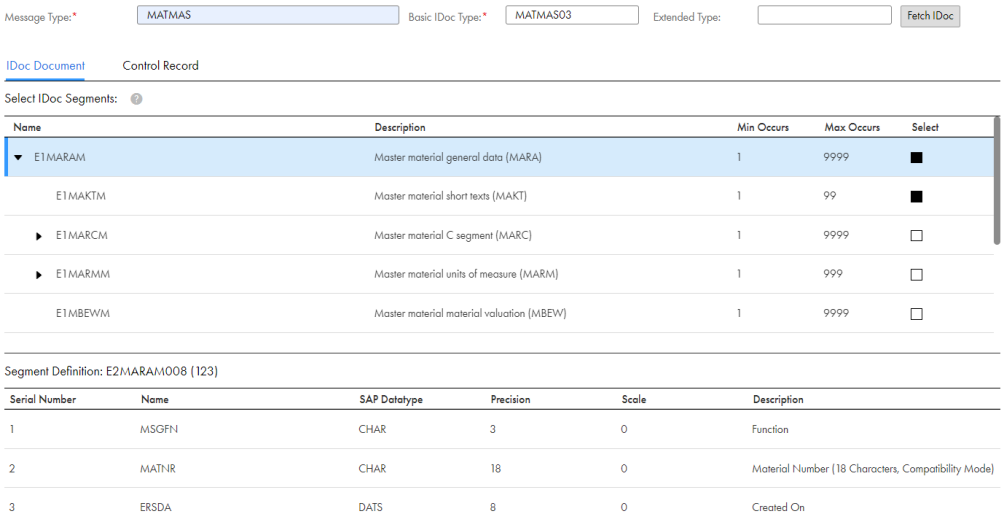
Step 1: Configuring an IDoc mapplet with the MATMAS IDoc

Configure an IDoc mapplet with the MATMAS IDoc from Data Integration. Use the configured mapplet to an inbound mapping.

1. In Data Integration, click **New** > **Mapplets** > **Mapplet - SAP IDoc**, and then click **Create**.
2. Enter a unique name for the mapplet.
3. Specify the location where you save the mapplet.
4. Optionally, enter a description for the mapplet.
5. Select **IDoc Prepare** as the transformation type.

   Note: Do not configure multiple IDoc Prepare transformations for an IDoc mapplet in an inbound mapping on Linux. Otherwise, the mapping fails with the DTM error.
6. From the **Connection** list, select the source connection by which you fetch the IDoc.
7. From the **Transformation Scope** list, select **All Input**.
8. In the **Message Type** field, enter MATMAS.
9. In the **Basic IDoc Type** field, enter MATMAS03.
10. Click **Fetch IDoc**.

    The segment details of the IDoc appears in the **Select IDoc Segments** section.



11. On the **IDoc Document** tab, in the **Select IDoc Segments** section, select the segments you want to include in the mapplet.
12. Optionally, to add a control record field to the mapplet, on the **Control Record** tab, select the checkbox for the field you want to add. This enables you to map the selected control record fields when you configure a mapping, integration template, or mapping task.
13. Click **Save**.

Step 2: Configuring an inbound mapping with the MATMAS IDoc

Configure an inbound mapping that uses the IDoc mapplet to write the MATMAS IDoc from SAP.

1. To create a mapping, click **Data Integration** > **New** > **Mappings**. Select **Mapping** and click **Create**.
2. Enter a name and description for the mapping, and click **OK**.

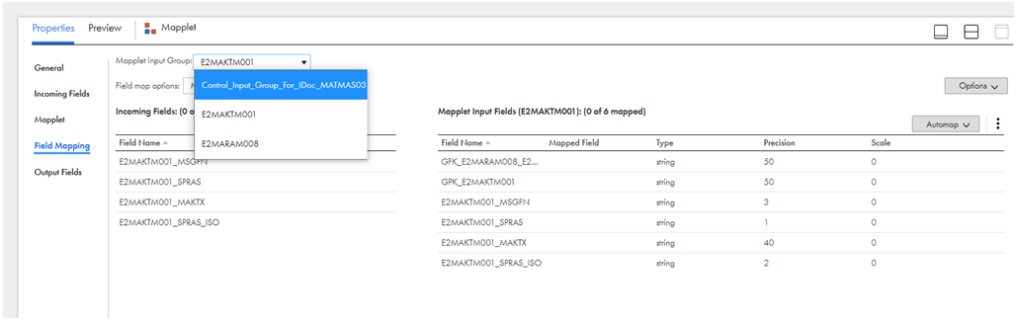
3. To configure a flat file source, on the **Transformation** palette, click **Source**.
4. In the **Properties** panel, on the **General** tab, enter a name and description.
5. Click the **Source** tab and select an flat file connection.
6. Add separate flat file sources for the control_input group and segments in the MATMAS IDoc.
7. To add an IDoc Mapplet transformation, on the **Transformation palette**, click Mapplet.
8. On the **General** tab, enter a name and description for the mapplet.
9. On the **Mapplet** tab, click **Select**.

   The **Select Mapplet** dialog box appears.
10. Specify an IDoc mapplet that you imported using the Prepare transformation and click **OK**.
11. Draw a link to connect the Source transformations to the Mapplet transformation. For example, connect the Control_Input source object to the Control_Input_Group for the MATMAS IDoc.
12. To preview the incoming fields, click **Incoming Fields**.

    The following image shows the incoming fields:

13. Click **Field Mappings** to map the incoming fields with the mapplet input fields.

    The following image shows the field mapping in the mapplet:



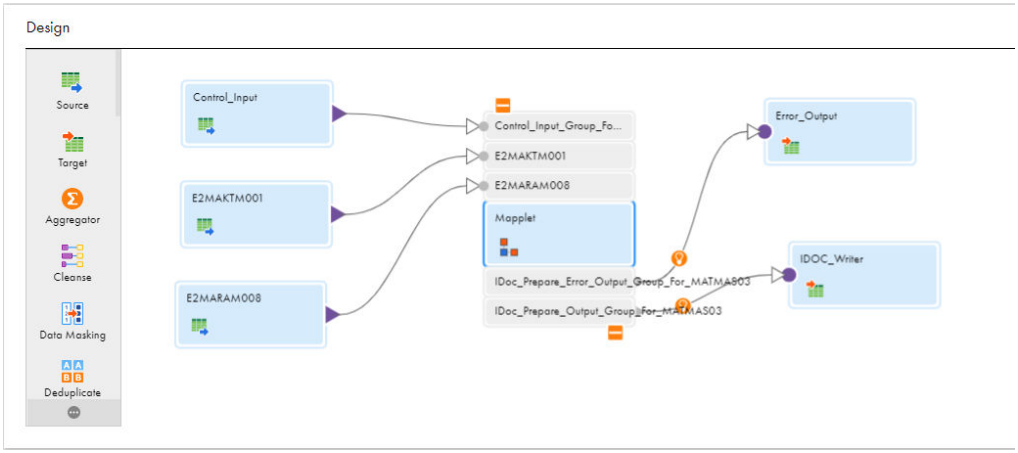
14. To add a Target transformation for writing to SAP, on the **Transformation palette**, click **Target**.
    a. On the **General** tab, enter a name and description.
    b. Click the **Target** tab and configure SAP target details.

    You can also parameterize the SAP target details.
15. To add a flat file Target transformation for the error output, on the **Transformation palette**, click **Target**.
    a. On the **General** tab, enter a name and description.

b. Click the **Target** tab and configure flat file target details.

16. Draw a link to connect the `IDoc_Prepare_Output_Group_For_MATMAS03` to the IDocWriter object. Draw another link to connect the `IDoc_Prepare_Error_Output_Group_For_MATMAS03` to the Error_Output flat file object.

    The following image shows the mapping for the inbound mapping for the MATMAS IDoc:



17. Save and run the mapping.

SAP mapplet properties

You can configure SAP mapplet properties in an outbound or inbound mapping task when you read IDocs from or write IDocs to SAP.

To configure the SAP mapplet properties, click the **Runtime Options** tab of the outbound or inbound mapping task. Expand the **SAP Mapplet Properties** section, and configure the mapplet properties.

The following table describes the SAP mapplet properties you can configure in an outbound or inbound mapping task:

Property	Outbound / Inbound	Description
Duplicate Parent Row Handling	Outbound and Inbound	Determines how Data Integration handles duplicate parent rows in a mapping task. Select one of the following options: - First Row. Data Integration passes the first duplicate row to the target. Data Integration rejects rows with the same primary key that it processes after this row. - Last Row. Data Integration passes the last duplicate row to the target. - Error. Data Integration passes the first row to the target. Rows that follow with duplicate primary keys increment the error count. The mapping task fails when the error count exceeds the error threshold. Default is First Row.
Orphan Row Handling	Outbound and Inbound	Determines how Data Integration handles orphan rows in a mapping task. Select one of the following options: - Ignore. Data Integration ignores orphan rows. - Error. The mapping task fails when the error count exceeds the error threshold. Default is Ignore.
Extended Syntax Check	Outbound and Inbound	Checks for IDocs that are not valid. If you use this property to validate IDocs, the performance of the mapping task might degrade.
Row Level Processing	Outbound	Data Integration processes each row of an outbound IDoc according to the IDoc metadata and passes it to a downstream transformation. If you enable row-level processing, Data Integration does not validate IDocs. Select this property to increase performance of the mapping task.
NULL Field Representation	Inbound	Determines how Data Integration handles fields with a null value in the IDoc prepare transformation. Select one of the following options: - Blank. Data Integration inserts all blanks for the field. - Slash (/). Data Integration inserts a single slash (/) for the field.
Cache Directory	Inbound	The directory used to cache inbound IDoc or DMI data. By default, the cache files are created in a directory specified by the variable $PMCacheDir. If you override the directory, make sure the directory exists and contains enough disk space for the cache files.
Cache Size	Inbound	Total memory in bytes allocated to Data Integration to cache data prepared for IDoc prepare transformations. Default is 10 MB.

Part VI: SAP BW data extraction

This part contains the following chapters:

CHAPTER 12

SAP BW data extraction

You can use SAP BW Connector to connect to the SAP BW system from Data Integration.

Use the SAP BW Connector connection to read data from SAP BW objects such as InfoCubes, InfoSets, MultiProviders, and DataStore objects.

You can use the SAP BW objects as sources in synchronization tasks, mappings, and mapping tasks.

CHAPTER 13

Synchronization tasks with SAP BW

You can configure a synchronization task to synchronize data between an SAP BW source and a target. You can use the Synchronization Task wizard to configure the synchronization task.

You can use SAP BW objects as sources. You can read data from InfoCubes, InfoSets, MultiProviders, and DataStore objects. You can use expressions to transform the data according to your business logic and use data filters to filter data before writing it to targets.
Note: You cannot perform a lookup on an SAP BW object.

When you create a task, you can associate it with a schedule to run it at specified times or at regular intervals. You can also manually run a task. You can monitor tasks and view logs about completed tasks.

SAP BW sources in Synchronization tasks

When you configure a synchronization task to use an SAP BW source, you can configure the source properties. The source properties appear on the **Source** page of the Synchronization Task wizard.

The following table describes the SAP BW source properties:

Property	Description
Connection	SAP BW Reader connection that you want to use to read data.
Source Type	Source type. You can read data from a single SAP BW object. You cannot edit this field.
Source Object	SAP BW source object from which you want to read data.
Display technical field names instead of labels	When selected, displays technical names instead of business names of the fields in the specified source object.
Display source fields in alphabetical order	When selected, displays source fields in alphabetic order. By default, fields appear in the order returned by the source system.
Data Preview	Displays the first 10 rows of the first five columns in the object, and the total number of columns in the object.
Preview All Columns	Previews all source columns in a file.

You can also configure advanced source properties when you schedule the synchronization task. Advanced source properties appear on the **Schedule** page of the Synchronization Task wizard.

The following table describes the SAP BW advanced source properties:

Property	Description
Packet Size in MB	Size of the HTTP packet that SAP sends to the Secure Agent. The unit is MB. Default is 10 MB.
Package Size in ABAP in Rows	Number of rows that are read and buffered in SAP at a time. Default is 1000 rows.
Enable Compression	When selected, the ABAP program compresses the data in the gzip format before it sends the data to the Secure Agent. If the Secure Agent and the SAP system are not on the same network, you might want to enable the compression option to optimize performance. Default is not selected.

Configuring a Synchronization task with a single SAP BW object as the source

1. To create a synchronization task, click **Data Integration** > **New** > **Tasks**. Select **Synchronization Task** and click **Create**.
2. Enter a name for the synchronization task.

   The names of synchronization tasks must be unique within the organization. Synchronization task names are not case sensitive. Synchronization task names can contain alphanumeric characters, spaces, and the following special characters: `_ . + -`
3. Enter a description for the synchronization task.

   The description can have a maximum length of 255 characters.
4. Select the task operation that you can perform on the target.

   You can select one of the following options:

   - Insert
   - Update
   - Upsert
   - Delete
5. Click **Next** to enter the source details.
   a. Select an SAP BW Reader connection from the list or create a new SAP BW Reader connection.
   b. Select an SAP BW source object from the list or click **Select** to search for an SAP BW source object.

      The **Select Source Object** dialog box appears. The dialog box displays up to 200 objects. If the objects that you want to use do not appear, enter a search string to search for the object based on its name, label, description, or type, and then click **Search**.
   c. Select the SAP BW source object from which you want to read data, and then click **Select**.

The **Data Preview** area displays the first 10 rows of the first five columns in the SAP BW object, and the total number of columns in the object. To preview all source columns in a file, click **Preview All Columns**.

d. To display technical names instead of business names, select the **Display technical field names instead of labels** option.

e. To display source fields in alphabetic order, select the **Display source fields in alphabetical order** option.

By default, fields appear in the order returned by the source system.

6. Click **Next** to specify the target connection and target objects.
7. Click **Next** to specify the data filter criteria.
   a. Click **New** to create a data filter. You can specify a simple data filter based on key characteristics.

      Note: You cannot create an advanced data filter. You cannot specify a simple data filter based on key figures.

   b. Select the source object, key characteristic source field, and operator.

      You can select one of the following operators for the filter condition:

      - Equals
      - Not Equals
      - Less Than
      - Less Than or Equals
      - Greater Than
      - Greater Than or Equals

   c. Enter the value based on which you want to filter the data and click **OK**.
8. Click **Next** to configure the field mappings. Perform any of the following steps based on your requirements:
   a. Click **Edit Types** in the Source column to edit the precision and scale of the SAP BW object.
   b. Click **Automatch** to match source and target fields with similar names.
   c. Click **Refresh Fields** to update the cache and view the latest field attributes.
   d. Click **Edit Types** in the Target column to edit the data type, precision, and scale of the target object. Note that this option is not available for all target types.
   e. Select a source field and drag it to the target field to map the source and target fields. Repeat for all the fields that you want to map.
   f. Click the **Add or Edit Expression** icon to define a field expression to transform data.
   g. Click **Validate Mapping** to validate all the field mappings.
   h. Click **Clear Mapping** to clear all the field mappings.
9. Click **Next** to configure a schedule and advanced options. Perform any of the following steps based on your requirements:
   a. Click **Run this task on schedule** and specify the schedule that you want to use. Alternatively, you can click **Do not run this task on a schedule** to run the task manually.
   b. Configure the email notification options.
   c. Configure a parameter file, if needed.
   d. Configure the advanced source properties.

e. Specify the execution mode.

10. Save the synchronization task. You can choose one of the following save options:
    - Click **Save and Close** to save and close the task.
    - Click **Save and Continue** to save the task and continue with configuring the task.
    - Click **Save and Run** to save and run the task immediately.

Monitoring a Synchronization task

After you run a synchronization task, you can monitor the task and view the logs.

In **Monitor**, you can monitor the status of the logs after you run the task.

You can also monitor the progress of the task by calling Transaction SM37 from SAP. You can view the actual job duration in SAP. The job duration listed in the Data Integration activity log is a higher value because it also includes time required to complete processing in Data Integration.

You can view the HTTP and HTTPS log files in the SMICM transaction. Optionally, you can increase trace level to 3 to view the detailed logs.

CHAPTER 14

Mappings and Mapping tasks with SAP BW

Use a mapping to define data flow logic that is not available in synchronization tasks, such as specific ordering of logic. Use the Data Integration Mapping Designer to configure mappings.

You can create a mapping to read data from an SAP BW source. You cannot perform a lookup on an SAP BW object. In advanced mode, the Mapping Designer updates the mapping canvas to include transformations and functions that enable advanced functionality.

When you configure a mapping to describe the flow of data from a source to a target, you can also add transformations to transform data. A transformation includes field rules to define incoming fields. Links visually represent how data moves through the data flow.

After you create a mapping, you can run the mapping or you can deploy the mapping in a mapping task. You can use the Mapping application to process data based on the data flow logic that you defined in the mapping.

Use the Mapping Task wizard to create a mapping task. When you create a mapping task, you can select the mapping based on which you want to create the task.

If you configured parameters, which are placeholders for information, in a mapping, you can define the parameters in the mapping task. You can define parameters for additional flexibility and to use the same mapping in multiple mapping tasks. For example, you can use a parameter for a source connection in a mapping, and then define the source connection when you configure the mapping task.

When you create a mapping task, you can associate the task with a schedule to run it at specified times or at regular intervals. Or, you can run it manually. You can also configure advanced session properties. You can monitor tasks and view details about completed tasks.

SAP BW sources in mappings

To read data from an SAP BW source, configure an SAP BW object as the Source transformation in a mapping.

Specify the name and description of the SAP BW source. Configure the source and advanced properties for the source object.

The following table describes the SAP BW source properties that you can configure in a Source transformation:

Property	Description
Connection	SAP BW Reader connection that you want to use to read data.
Source Type	Source type. Select one of the following types: - Single. Select to specify a single SAP BW object. - Parameter. Select to specify a parameter name. You can configure the source object in a mapping task associated with a mapping that uses this source transformation.
Object	SAP BW source object from which you want to read data.

The following table describes the SAP BW advanced source properties:

Property	Description
Packet Size in MB	Size of the HTTP packet that SAP sends to the Secure Agent. The unit is MB. Default is 10 MB.
Package Size in ABAP in Rows	Number of rows that are read and buffered in SAP at a time. Default is 1000 rows.
Enable Compression	When selected, the ABAP program compresses the data in the gzip format before it sends the data to the Secure Agent. If the Secure Agent and the SAP system are not on the same network, you might want to enable the compression option to optimize performance. Default is not selected.
Tracing Level	Amount of detail that appears in the log file. You can select one of the following options: - Terse - Normal - Verbose Initialization - Verbose Data Default is normal.

Configuring a mapping with a single SAP BW source

Use the Data Integration Mapping Designer to configure a mapping.

1. To create a mapping, click **Data Integration** > **New** > **Mappings**. Select **Mapping** and click **Create**.
2. Enter a name and description for the mapping, and click **OK**.

   You can use alphanumeric characters and underscores (_) in the mapping name.
3. To configure an SAP BW source, on the **Transformation** palette, click **Source**.
4. In the **Properties** panel, on the **General** tab, enter a name and description for the SAP BW source.
5. Click the **Source** tab and configure the SAP BW source details.

6. Select an SAP BW Reader connection from the list or click **New Connection** to create a new SAP BW Reader connection.

   You can also parameterize the connection.
7. Specify the source type. You can choose one of the following options:
   - Select **Single Object** to select a single SAP BW object.
   - Select **Parameter** to configure the SAP BW source objects in a mapping task associated with this mapping.
8. Enter the SAP BW object name or click **Select** to search for an object based on its name, label, description, or type.
9. Select the SAP BW object from which you want to read data and click **OK**.
10. Click **Query Options** in the **Source** tab to specify data filters for the SAP BW object.

    You can specify a simple data filter based on key characteristic fields.

    Note: You cannot create an advanced data filter.

    a. Click **Configure** and then click the **+** icon to add a data filter.

    b. Select the source object, key characteristic source field, and operator.

       You can select one of the following operators for the filter condition:
       - Equals
       - Not Equals
       - Less Than
       - Less Than or Equals
       - Greater Than
       - Greater Than or Equals

    c. Enter the value that you want to use and click **OK**.

       You can also parameterize the filter condition.
11. Click **Advanced** to specify the advanced source properties.
12. To add or remove source fields, update field metadata, or synchronize fields with the source, click the **Fields** tab.
13. To add a transformation, on the **Transformation palette**, click the transformation name. Or, drag the transformation onto the mapping canvas.

    a. On the **General** tab, you can enter a name and description for the transformation.

    b. Draw a link to connect the previous transformation to the transformation.

       When you link transformations, the downstream transformation inherits the incoming fields from the previous transformation.
       For a Joiner transformation, draw a master link and a detail link.

    c. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.

    d. Configure additional transformation properties, as needed.

       The properties that you configure vary based on the type of transformation you create.

    e. To add another transformation, repeat these steps.
14. To add a Target transformation, on the **Transformation palette**, click **Target**.

    a. On the **General** tab, enter a name and description for the target.

    b. Draw a link to connect the previous transformation to the Target transformation.

c. Click the **Target** tab and configure target details. If necessary, configure the advanced target properties.

   Target details and advanced target properties appear based on the connection type. For more information, see *Transformations*.

d. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.

e. Click **Field Mapping** and map the fields that you want to write to the target.

f. To add another Target transformation, repeat these steps.

15. Save the mapping. You can choose one of the following save options:

- Click **Save and Close** to save and close the mapping.
- Click **Save and Run** to save and run the mapping immediately.
- Click **Save and New Mapping Task** to save and create a mapping task.

Creating a Mapping task

You can create a mapping task based on a valid mapping.

1. To create a mapping task, click **Data Integration** > **New** > **Tasks** and then complete one of the following steps:
   - To create a mapping task based on a mapping, select **Mapping Task** and click **Create**.
   - To create a mapping task using a template, expand the appropriate template category and select the template you want to use, and then click **Create**.

   To edit a mapping task, on the **Explore** page, navigate to the mapping task. In the row that contains the task, click **Actions** and select **Edit**.
2. Enter a name for the task.

   Task names must be unique within the organization. Task names are not case sensitive. Task names can contain alphanumeric characters, spaces, and the following special characters: _ . + -
3. Select the runtime environment that contains the Secure Agent that you want to use to access the SAP BW objects.
4. Select **Mapping** as the task based on which you want to create the mapping task.
5. Click **Select** to specify a mapping.

   The **Select a Mapping** dialog box appears.
6. Select a mapping. You can also search for a mapping based on its name and description.
7. Click **OK**.

   The image of the selected mapping appears.
8. Click **Next**.

   If you specified any parameters in the source or target details in the mapping, the **Source** or **Target** page appears. If not, the **Schedule** page appears.
9. Configure a schedule and advanced options. Perform any of the following steps based on your requirements:

   a. Click **Run this task on schedule** and specify the schedule that you want to use.

b. Configure the email notification options.
c. Configure advanced options for the task.
d. Specify the execution mode.

10. Optionally, add advanced session properties.
    a. Click **Add**.
    b. Select a session property.
    c. Configure the value of the session property.
11. Save the mapping task. You can choose one of the following save options:
    - Click **Save and Close** to save and close the task.
    - Click **Save and Continue** to save the task and continue with configuring the task.
    - Click **Save and Run** to save and run the task immediately.

Monitoring a Mapping task

After you run a mapping task, you can monitor the task and view the logs.

In **Monitor**, you can monitor the status of the logs after you run the task.

You can also monitor the progress of the task by calling Transaction SM37 from SAP. You can view the actual job duration in SAP. The job duration listed in the Data Integration activity log is a higher value because it also includes time required to complete processing in Data Integration.

You can view the HTTP and HTTPS log files in the SMICM transaction. Optionally, you can increase trace level to 3 to view the detailed logs.

Rules and guidelines for SAP BW sources

Consider the following rules and guidelines when you configure the SAP BW sources:

- When you establish a communication from the SAP system with the Secure Agent using the IP address of the NAT gateway, you must include the DTM property named SapTableReaderNatIpAddress for the Secure Agent and specify the NAT IP address as the value.
- When you create a mapping that contains an InfoObject with the **Attribute Only** option selected, the mapping fails.

Part VII: Data integration for SAP ADSO

This part contains the following chapters:

CHAPTER 15

Data integration for SAP ADSO

You can use the SAP ADSO Writer connection to load large volumes of data to the target ADSO objects in the SAP BW/4HANA application. To read data from the SAP ADSO objects, you can use the SAP Table Connector connection.

ADSO is the central object for data storage and consolidation in the SAP BW system. ADSO is the primary data persistency object in BW/4HANA that simplifies data modeling and the support environment. ADSOs are commonly stored in 2-dimensional transparent database tables.

CHAPTER 16

Data Integration for SAP ADSO sources using SAP Table Reader

You can use SAP Table Reader Connector to connect to the SAP BW/4HANA application and read from SAP ADSOs.

SAP Advanced DataStore Object (ADSO) is the central object for data storage and data consolidation in the SAP BW system. ADSO is the primary persistent object in BW/4HANA and simplifies the modeling and support environment. ADSOs are commonly stored in 2-dimensional transparent database tables.

You can use SAP Table Reader Connector in Data Integration to read large volumes of data from the SAP BW/4HANA application.

Data Integration imports the following SAP ADSO information:

- ADSO name
- Description
- Type
- Info area
- Transfer structure
- Field information such as the name, type, description, precision, scale, SAP type, SAP length, SAP output length, and SAP decimals.

Importing an SAP ADSO object

To read from an SAP ADSO, use SAP Table Reader Connector. Import an ADSO object from SAP as a source in Data Integration mappings.

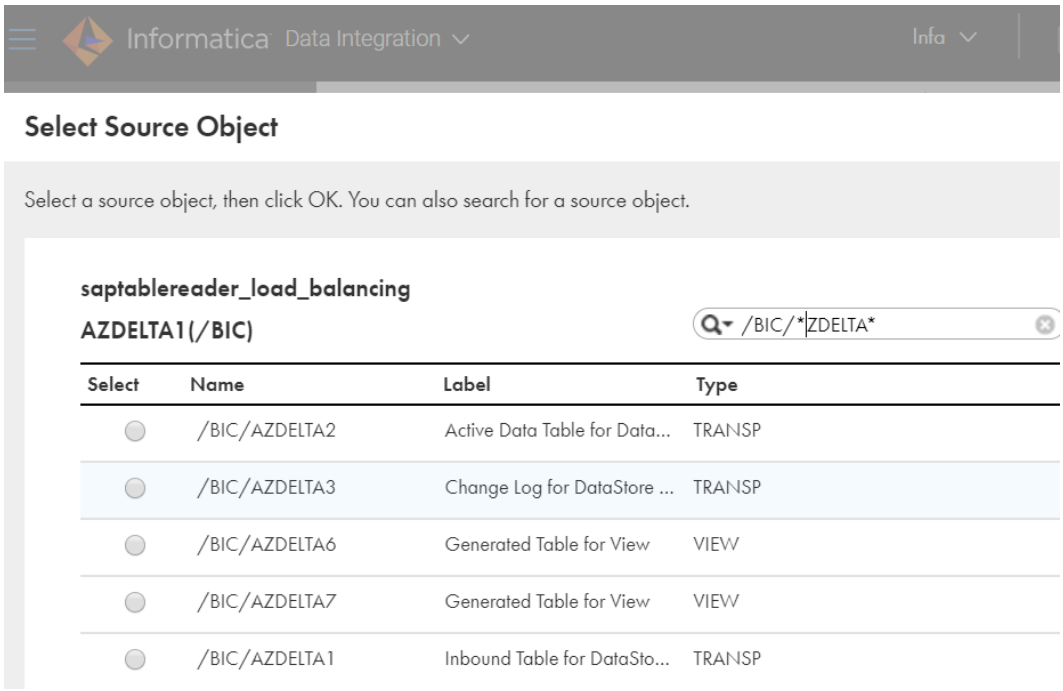
The naming convention of an ADSO is: `/BIC/A<technical_name of ADSO>*`, where * represents the table number associated with an ADSO.

When you import an ADSO into Cloud Data Integration, the Secure Agent imports the following five tables associated with each ADSO:

- `/BIC/A<technical_name of ADSO>1`. Inbound table for data store <technical_name of ADSO>.
- `/BIC/A<technical_name of ADSO>2`. Active data table for data store <technical_name of ADSO>.
- `/BIC/A<technical_name of ADSO>3`. Change log for data store <technical_name of ADSO>.
- `/BIC/A<technical_name of ADSO>6`. View for extraction from data store <technical_name of ADSO>.

- /BIC/A<technical_name of ADSO>7. View for reporting for datastore <technical_name of ADSO>.
- /BIC/A<technical_name of ADSO>8. View for external access (introduced in BW/4HANA 2.0 <technical_name of ADSO>.

The following image shows the five tables of an ADSO with technical name ZDELTA that you can import from BW/4HANA 1.0 into Cloud Data Integration:



CHAPTER 17

Data Integration for SAP ADSO targets using SAP ADSO Writer

You can use SAP ADSO Writer Connector to connect to the SAP BW/4HANA application and write to ADSOs.

SAP Advanced DataStore Object (ADSO) is the central object for data storage and data consolidation in the SAP BW system. ADSO is the primary persistent object in BW/4HANA and simplifies the modeling and support environment. ADSOs are commonly stored in 2-dimensional transparent database tables.

You can use SAP ADSO Writer Connector in Data Integration to load large volumes of data to the target ADSO objects in the SAP BW/4HANA application.

Data Integration imports the following SAP ADSO information:

- ADSO name
- Description
- Type
- Info area
- Transfer structure
- Field information such as the name, type, description, precision, scale, SAP type, SAP length, SAP output length, and SAP decimals.

Rules and guidelines for SAP ADSO targets

Use the following rules and guidelines when you configure SAP ADSO targets:

- Only the activated ADSOs appear in the browser list of the SAP ADSO Writer Connector. Ensure that the required ADSO is activated in the SAP system before you can use in Data Integration.
- If the ADSO name has namespace `(“/”)` characters at the beginning, the Secure Agent replaces those characters with `T__`. Every subsequent namespace character in the ADSO name is replaced by `“__”`. For example, if the ADSO name is `/IMO/CMSD17`, it appears in the browser list as `T__IMO__CMSD17`.
- Do not edit the field metadata as it might corrupt the extracted data.
- If the field name of the selected ADSO has a namespace character (“/”), the Secure Agent replaces it with an underscore “_” character.
- When you define a mapping, you must select the operation as Insert. While activating, SAP BW/4HANA uses its algorithm to handle loaded data. You do not need to set any additional property in Data

Integration to perform update or delete operation. Operations such as update and delete are handled at the SAP BW/4HANA layer itself.

- Ensure that data with the different data types that you load to BW/4HANA is correct. The activation of the loaded packet might fail if the inserted data is incorrect.
- The Secure Agent does check the validity of the data according to the supported transformation data type. However, if the data is incorrect, the Secure Agent loads the data to SAP BW/4HANA, but fails during the data activation. Ensure that the data is valid according to the data type of the fields.
- The default packet size to write data is 8 MB. When you set the packet size value equal to greater than 8 MB and you encounter out-of-memory issues, you can increase the Java heap memory for the Secure Agent. In the **System Configuration Details** section of the Secure Agent properties, select **DTM** and set the **JVMOption1** property to the following value: *Xmx512m*. Click **OK** to save the change and then restart the Secure Agent. Adjust the value for the JVMOption1 property based on the amount of records you want to write and the available memory on the Secure Agent machine.
- The default precision for variable length data types such as STRING and RAWSTRING is set to a higher value of 1333. To manage the optimal record length, you must, however, set the length for these data types in the Edit Metadata Section of the mapping based on the expected length of the actual data.

CHAPTER 18

Mappings and mapping tasks with SAP ADSO Writer

Use a mapping to define the data flow logic. Use the Data Integration Mapping Designer to configure mappings. In advanced mode, the Mapping Designer updates the mapping canvas to include transformations and functions that enable advanced functionality.

When you configure a mapping to describe the flow of data from source and target, you can also add transformations to transform data. A transformation includes field rules to define incoming fields. Links visually represent how data moves through the data flow.

After you create a mapping, you can run the mapping or you can deploy the mapping in a mapping task. The Mapping Configuration application allows you to process data based on the data flow logic defined in a mapping or integration template.

Use the Mapping task wizard to create a mapping task. When you create a mapping task, you select the mapping or integration template for the task to use.

If you configured parameters, which are placeholders for information, in a mapping, you can define the parameters in the mapping task. Defining parameters provides additional flexibility and allows you to use the same mapping in multiple mapping tasks. For example, you can use a parameter for a target connection in a mapping, and then define the target connection when you configure the mapping task.

When you create a mapping task, you can associate the task with a schedule to run it at specified times or on regular intervals. Or, you can run it manually. You can also configure advanced session properties. You can monitor tasks that are currently running and view details about completed tasks.

SAP ADSO Extractor targets in mappings

To write data to an SAP BW/4HANA application, configure an SAP ADSO object as the Target transformation in a mapping.

Specify the name and description of the SAP ADSO target. Configure the target and advanced properties for the target object.

The following table describes the SAP ADSO Writer target properties that you can configure in a Target transformation:

Property	Description
Connection	Name of the source connection.
Target Type	The source type. Select from one of the following types: - Single Object. Select to specify a single SAP ADSO object. - Parameter. Select to specify the SAP ADSO object as a parameter.
Object	Target object for the task. Click **Select** and then select the target object.
Operation	The type of operation for the task. Only Insert is applicable.
Preview Data	Not applicable.

The following table describes the SAP ADSO Writer advanced target properties:

Property	Description
Activate Option Mode	Loads the data into the target ADSO object and then sends data activation requests to SAP BW/4HANA based on the activate options you select. You can select from the following activate options: - **0-Do Not Activate**. Does not send an activate request. - **1-Activate**. Creates a single activation request for all load requests in the queue. - **2-Activate Request by Request**. Creates a separate activation request for each of the load requests in the queue. - **Parameter**. Use this option to parameterize the activate options. Specify a valid activate option value in the **Parameter Name for Activate Option Mode** field. Default is Do Not Activate.
Parameter Name for Activate Option Mode	The parameter name that you defined for the **Activate Option Mode** in the parameter file. You can specify the following values for the data fetch mode: - 0 for Do Not Activate - 1 for Activate - 2 for Activate Request by Request If you pass any other value, the jobs fail when you run the mapping,
Packet size in MB	The packet size of each SAP data packet while writing data to SAP BW4/HANA. Default value is 10 MB.
Tracing	Sets the amount of detail that appears in the log file. You can choose terse, normal, verbose initialization, or verbose data. Default is normal

Activate option mode for SAP ADSO Writer mappings

You can select the activate option mode when you load data to the ADSO target in SAP BW/4HANA.

When you specify the activate mode, the Secure Agent first loads the data into the target ADSO object and then sends the data activation requests to SAP BW/4HANA based on the following activate options you select:

- **0-Do Not Activate**. Does not send an activate request and the activation precessing does not occur.
- **1-Activate**. Creates a single activation request for all load requests in the queue.
- **2-Activate Request by Request**. Creates a separate activation request for each of the load requests in the queue.

The activation request activates all the previous load requests in the queue that are pending for activation. Activation occurs per load request in a chronological sequence, starting from the earliest pending request to the most recent load request. The Secure Agent waits for the activation to be fully processed in SAP BW/4HANA system. The wait time depends upon the number of load requests in the queue that were available earlier to the load request associated with the current mapping.

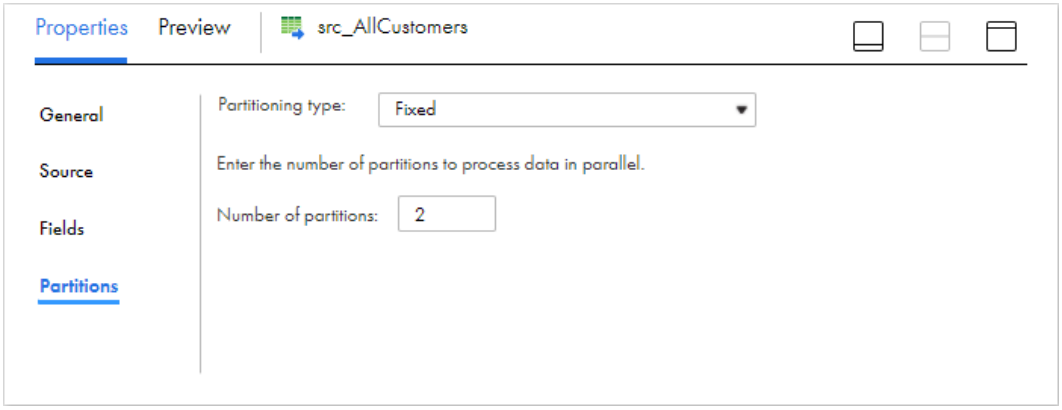
During the activation process, if the activation of a specific load request fails, the whole activation process fails and the rest of the load requests are also not activated. In such cases, the mapping fails even though the Secure Agent loaded the data correctly based on the enabled activate mode.

Fixed partitioning for SAP ADSO targets

You can configure fixed partitioning in a mapping that writes to SAP ADSO targets.

Enable partitioning when you configure the Source transformation in the Mapping Designer. You do not have to enable fixed partitioning in the Target transformation. The Secure Agent considers the same number of partitions you specified for the source as the target partition number.

On the **Partitions** tab for the Source transformation, you select fixed partitioning and enter the number of partitions, as shown in the following image:



When you run the mapping, the Secure Agent distributes rows of data for the target based on the number of partitions that you specify in the Source transformation. You can specify up to 64 partitions.

Configuring a mapping with an SAP ADSO target

Use the Data Integration Mapping Designer to configure a mapping.

1. To create a mapping, click **Data Integration** > **New** > **Mappings**. Select **Mapping** and click **Create**.
2. Enter a name and description for the mapping, and click **OK**.

   You can use alphanumeric characters and underscores (_) in the mapping name.
3. To configure a source, on the **Transformation** palette, click **Source**.
4. In the **Properties** panel, on the **General** tab, enter a name and description.
5. Click the **Source** tab and configure source details.
6. Specify the source type. You can choose one of the following options:
   - Select **Single Object** to select a single SAP object.
   - Select **Parameter** to configure the source objects in a mapping task associated with this mapping.
7. Click **Query Options** in the **Source** tab to specify any filter for the SAP ADSO object.
8. Click **Advanced** to specify the advanced source properties.

   Note: You cannot add or remove source fields or edit the field metadata.
9. To add a transformation, on the **Transformation palette**, click the transformation name. Or, drag the transformation onto the mapping canvas.
   a. On the **General** tab, you can enter a name and description for the transformation.
   b. Draw a link to connect the previous transformation to the transformation.

      When you link transformations, the downstream transformation inherits the incoming fields from the previous transformation.
      For a Joiner transformation, draw a master link and a detail link.
   c. To preview fields, configure the field rules, or rename fields, click **Incoming Fields**.
   d. Configure additional transformation properties, as needed.

      The properties that you configure vary based on the type of transformation you create.
   e. To add another transformation, repeat these steps.
10. To add an ODSO object to the Target transformation, on the **Transformation palette**, click **Target**.
    a. On the **General** tab, you can enter a name and description.
    b. Draw a link to connect the previous transformation to the Target transformation.
    c. Click the **Target** tab, configure the target details. Specify the target type:
       - Select **Single Object** to select an ADSO object.
       - Select **Parameter** to configure the target objects in a mapping task associated with this mapping.
       - Select **Operation** as **Insert**.
       - Click **Advanced** and specify the advanced target properties.
    d. Click **Field Mapping** and map the fields that you want to write to the target.
    e. To add another Target transformation, repeat these steps.
11. Click the **Field Mapping** to map the Source and Target fields.

    You can use the Automatic option if the fields are similar between the source and target.
12. Save and run the mapping or save and create a mapping task.

Creating a mapping task

You can create a mapping task based on a valid mapping or integration template on the Mappings page.

1. To create a mapping task, click **Data Integration** > **New** > **Tasks**.
2. Enter a name for the task.

   Task names must be unique within the organization. Task names can contain alphanumeric characters, spaces, and the following special characters:_ . + -Task names are not case sensitive.
3. Select the runtime environment that contains the Secure Agent that you want to use to access the SAP ADSO Writer connection.
4. Select **Mapping** as the task based on which you want to create the mapping task.
5. Click **Select** to specify a mapping.

   The **Select a Mapping** dialog box appears.
6. Select a mapping or search for the required mapping and select **OK**.

   The image of the selected mapping appears.
7. Click **Next**.

   If you specified any parameters in the source or target details in the mapping, the Source or Target page appears. If not, the Schedule page appears.
8. Click **Next** to configure a schedule and advanced options. Perform any of the following steps based on your requirements.
   a. Click **Run this task on schedule** and specify the schedule you want to use.
   b. Configure the email notification options.
   c. Configure advanced options for the task.
   d. Configure the advanced target properties.
   e. Specify the execution mode.
9. Optionally, add the advanced session properties.
   a. Click **Add**.
   b. Select a session property.
   c. Configure the value of the session property.
10. Save and run the mapping task.

APPENDIX A

SAP data type reference

Data Integration uses the following data types in mappings, synchronization tasks, and mapping tasks with SAP:

Native data types

Native data types are data types specific to the source and target databases or flat files. They appear in non-SAP sources and targets in the mapping.

SAP data types

SAP data types appear in the Fields tab for Source and Target transformations when you choose to edit metadata for the fields. SAP performs any necessary conversion between the SAP data types and the native data types of the underlying source database tables.

Transformation data types

Set of data types that appear in the remaining transformations. They are internal data types based on ANSI SQL-92 generic data types, which Data Integration uses to move data across platforms. Transformation data types appear in all remaining transformations in a mapping, synchronization task, or mapping task.

When Data Integration reads source data, it converts the native data types to the comparable transformation data types before transforming the data. When Data Integration writes to a target, it converts the transformation data types to the comparable native data types.

SAP and transformation data types

The following table lists the SAP data types that SAP IDoc Reader, IDoc Writer, and RFCs/BAPI Connector support along with the corresponding transformation data types:

SAP Data Type	Transformation Data Type	Range for Transformation Data Type
ACCP	Date/time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D.
CHAR	String	1 to 104,857,600 characters Fixed-length or varying-length string.
CLNT	String	1 to 104,857,600 characters Fixed-length or varying-length string.

SAP Data Type	Transformation Data Type	Range for Transformation Data Type
CUKY	String	1 to 104,857,600 characters Fixed-length or varying-length string.
CURR	Decimal	Precision 1 to 28 digits, scale 0 to 28
DATS	Date/time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. Precision to the nanosecond.
DEC	Decimal	Precision 1 to 28 digits, scale 0 to 28
DF16_DEC	Decfloat16	Range of 1-15 and scaling of maximum 14. Decimal floating point number stored in BCD format. You can use the DF16_DEC data type in the following scenarios: - When you configure SAP BAPI Connector as a business service within a mapping or a mapping task to integrate with SAP - When you use SAP BAPI/RFC mapplets to integrate with SAP
DF34_DEC	Decfloat34	Range of 1-31 and scaling of maximum 30. Decimal floating point number stored in BCD format. You can use the DF34_DEC data type when you configure SAP BAPI Connector as a business service within a mapping or a mapping task to integrate with SAP.
DF16_RAW	Double	Maximum of 16 positions with floating decimal. Decimal floating point number stored in binary format. You can use the DF16_RAW data type in the following scenarios: - When you configure SAP BAPI Connector as a business service within a mapping or a mapping task to integrate with SAP - When you use SAP BAPI/RFC mapplets to integrate with SAP
DF34_RAW	Double	Maximum of 34 positions with floating decimal. Decimal floating point number stored in binary format. You can use the DF34_RAW data type in the following scenarios: - When you configure SAP BAPI Connector as a business service within a mapping or a mapping task to integrate with SAP - When you use SAP BAPI/RFC mapplets to integrate with SAP
FLTP	Double	Precision 15, scale 0
INT1	Small Integer	Precision 5, scale 0
INT2	Small Integer	Precision 5, scale 0
INT4	Integer	Precision 10, scale 0
INT8	Int8	8-byte integer between -9,223,372,036,854,775,808 and +9,223,372,036,854,775,807. The length is set at 19 positions.
LANG	String	1 to 104,857,600 characters Fixed-length or varying-length string.
LCHR	String	1 to 104,857,600 characters Fixed-length or varying-length string.

SAP Data Type	Transformation Data Type	Range for Transformation Data Type
LRAW	Binary	Uninterrupted sequence of bytes with a maximum length of 255 positions.
NUMC	String	1 to 104,857,600 characters Fixed-length or varying-length string.
PREC	Binary	Uninterrupted sequence of bytes with a maximum length of 255 positions.
QUAN	Decimal	Precision 1 to 28 digits, scale 0 to 28
RAW	Binary	Uninterrupted sequence of bytes with a maximum length of 255 positions.
RAWSTRING	Binary	Uninterrupted byte string. You can use the RAWSTRING data type in the following scenarios: - When you use IDocs to integrate with SAP - When you use SAP BAPI/RFC mapplets to integrate with SAP
SSTRING	String	Small Character string. You can use the SSTRING data type when you use SAP BAPI/RFC mapplets to integrate with SAP.
STRING	String	Character string. You can use the STRING data type in the following scenarios: - When you use SAP BAPI/RFC mapplets to integrate with SAP - When you use IDocs to integrate with SAP
TIMS	Date/time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. Precision to the nanosecond.
UNIT	String	1 to 104,857,600 characters Fixed-length or varying-length string.
VARC	String	1 to 104,857,600 characters Fixed-length or varying-length string.

SAP BW and transformation data types

The following table lists the SAP data types that SAP BW Connector supports along with the corresponding transformation data types:

SAP Data Type	Transformation Data Type	Range for Transformation Data Type
CHAR	String	1 to 104,857,600 characters Fixed-length or varying-length string.
CUKY	String	1 to 104,857,600 characters Fixed-length or varying-length string.

SAP Data Type	Transformation Data Type	Range for Transformation Data Type
CURR	Decimal	Precision 1 to 28 digits, scale 0 to 28
DATS	Date/time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. Precision to the nanosecond.
DEC	Double	Precision 1 to 28 digits, scale 0 to 28
FLTP	Double	Precision 15, scale 0
INT4	Integer	Precision 10, scale 0
LANG	String	1 to 104,857,600 characters Fixed-length or varying-length string.
NUMC	String	1 to 104,857,600 characters Fixed-length or varying-length string.
QUAN	Decimal	Precision 1 to 28 digits, scale 0 to 28
TIMS	Date/time	Jan 1, 0001 A.D. to Dec 31, 9999 A.D. Precision to the nanosecond.
UNIT	String	1 to 104,857,600 characters Fixed-length or varying-length string.

INDEX

O

P

R

S

T

U