

CI to Fieldglass Generic Upload



Content

1 Prerequisites	4
2 Documentation	5
3 Configuration steps on SAP Cloud Integration	6
3.1 Configure Sender Adapter	6
3.2 Configure Receiver Adapter	9

1 Prerequisites

This content is upload content to Fieldglass using Fieldglass REST APIs according to [SAP Fieldglass REST API Integration](#). You might also involve an SAP Fieldglass expert for this set up.

This iflow is to be used in along with calling iflow that is used to extract the data from the source system and call this generic iflow using PROCESS DIRECT sender adapted.

2 Documentation

The iflow in the package can be used to upload data to Fieldglass from any source using the Fieldglass Rest API's that use connectors . The input to these connectors is csv files. Input to the iflow can be either in xml, Json or csv format. The iflow takes care of the conversion to CSV format and uploading the data to Fieldglass. Please refer to [SAP Fieldglass REST API Integration](#) for more information on the uploading data to Fieldglass using connectors.

Other advantages of the iflow is, it can be used to call one time or multiple times in the same calling program to upload different sets of data.

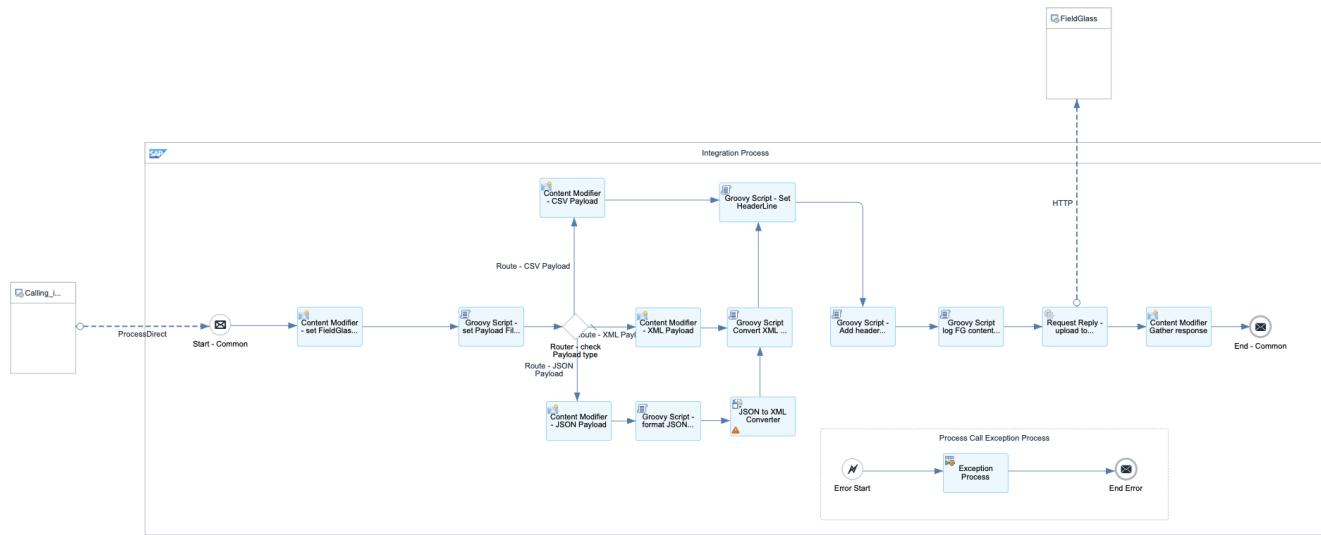


Figure 1: iFlow overview

3 Configuration steps on SAP Cloud Integration

Integration Procedure:

FieldGlass uses the upload mechanism to load both master data and transactional data using SOAP or HTTP protocol.

Here, we use HTTP protocol using OAuth2 Client Credentials.

- 1.1 Cloud Integration - How to Call the Integration Flow
- 1.2 Configuration of the Integration Flow in Cloud Integration

3.1 Configure Sender Adapter

Sender Adapter is a Process Direct adapter. Which mean we need a calling iflow to call this generic upload iflow.

Most of the parameter used in the generic upload iflow are passed as externalized parameters from the calling iflow.

Below are steps to configure the parameters in the calling iflow:

1 Cloud Integration - How to Call the Integration Flow

To work with this generic upload API/package, we will first need to create a custom integration flow and then call this standard generic integration flow that would then process the data.

Step 1: Create a custom Integration flow.

- I. to extract the data from source system
- II. API supports 3 different forms of data. XML, CSV and JSON.
 - for format XML- transform the payload to XML format like the fields to be uploaded for any specific FieldGlass upload API.
 - For CSV format – pass the CSV payload and set the Header parameter “FG_HeaderLine” as given in the CSV content header.
 - For JSON format – the fields sequence in the file should match the input to the Header parameter “FG_HeaderLine”

Sample Fieldglass file to upload “Supplier Plant Association Upload”

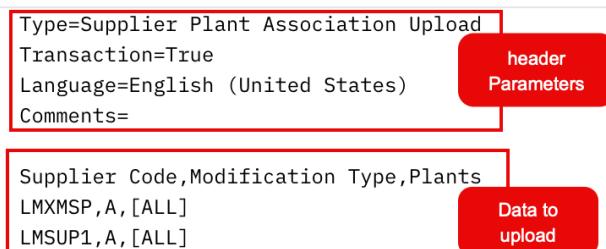


Figure 2: Sample Fieldglass upload file

Step 2: Add a Content Modifier.

- I. In the **Message Header** tab create the below mentioned variables **exactly** as shown in the screenshot below. (**Note:** these variables are passed to the generic Integration Flow and the valued are used as part of it for further processing.)

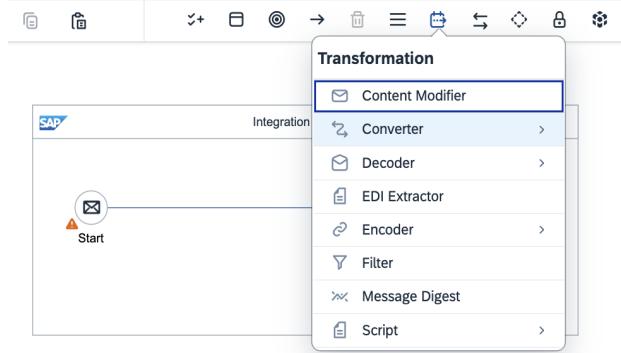


Figure 3: Content Modifier

Action	Name	Source Type	Source Value	Data Type
Create	UsedForPayloadHeader	Constant	[Number Format, Trans...	
Create	Type	Constant	Pick List Value Upload	
Create	Transaction	Constant	False	
Create	Number Format	Constant	"#,##9.99 (Example: 1,...	
Create	FG_HeaderLine	Constant	Modification Type,PO Li...	
Create	FG_Host_URL	Constant	int75.fgvmms.com/api/vc...	
Create	APIMethod	Constant	POST	
Create	FG_Credentials	Constant	FieldGlass_OAUTH	
Create	ENABLE_LOGGING	Constant	YES	
Create	Send_Exception_Email	Constant	YES	
Create	Payload_Type	Constant	CSV	
Create	calling_iflow	Expression	\${camelId}	

Figure 4: Content Modifier: Header parameters

Variables to create: These can be categorized into 4 types

1. CSV file Header properties -

- a. **UsedForPayloadHeader** – all the header parameters required as part of the CVS file to be uploaded are included within Square braces ("[" and "]").

(note: These fields can vary from connector type used)

Ex: in the sample file above. **Type, Transaction, Number Format** as show in the above screenshots for **sample file** and the **Header parameters**

- b. CSV file parameters:

- CSV Header Fields: Fields to be added as required by the Fieldglass Connector used. These need to be exactly as mentioned in the [SAP Fieldglass REST API Integration](#)
 - **Type** – A constant value to set the connector name used to upload the data to FieldGlass. Example: Supplier Plant Association Upload”.
 - **Transaction** - used as part of the FG payload header(to be configurations as required)
 - **Number Format** : used as part of the FG payload header(to be configurations as required)
- **FG_HeaderLine** : A constant value with the header fields of the CSV file. Example: For upload of “Supplier Plant Association Upload”, header fields are Supplier Code, Modification Type, Plants

2. HTTP parameters:
 - a. FG_Host_URL : FieldGlass host url
 - b. FG_Credentials: FieldGlass OAuth2 credentials as create in the Monitoring -> **Manage Security Material**
 - c. APIMethod: GET / POST / DELETE or any other as supported by HTTP adapter.
3. Fields used as part of the processing
 - a. Enable_Logging: Yes / No.
 - b. Send_Exception_Email: Yes / No. used to send exception details as email.
 - c. Payload_Type: XML / CSV / JSON. Use one of the three as per your requirements.
 - d. Calling_Iflow: this is used as part of the exception body and is optional.

Step 3: Add a Request-Reply and a receiver to your flow

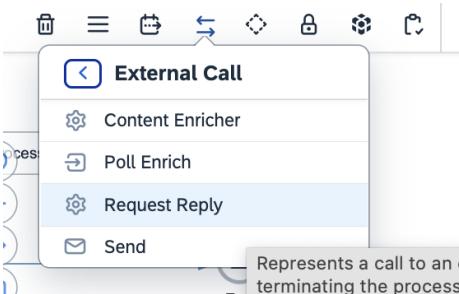


Figure 5: Request-Reply

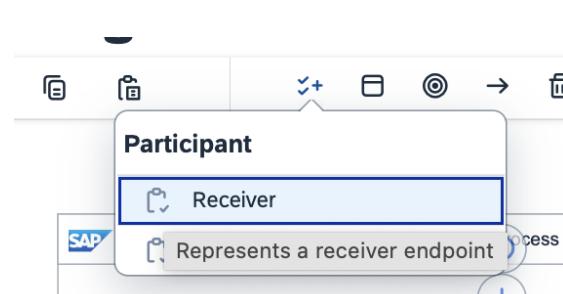


Figure 6: Receiver

Step 4: Connect the Request-Reply to the Receiver using a ProcessDirect adapter type.

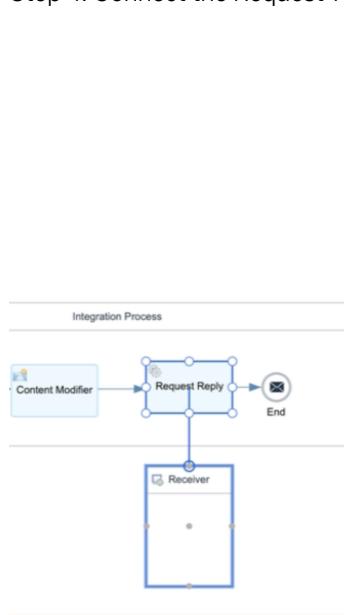


Figure 7: Connectivity

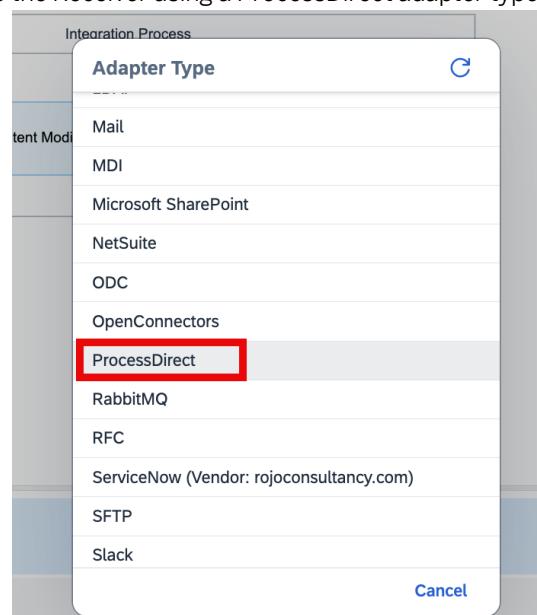


Figure 8: Receiver adapter: ProcessDirect

Step 5: In the connection tab of the ProcessDirect adapter enter the following address, **/FieldGlass_upload** (note: validate it by checking it in the downloaded iflow) as shown in figure below:

ProcessDirect

General Connection

CONNECTION DETAILS

Address: * /FieldGlass_upload

Figure 9: ProcessDirect adapter connection

3.2 Configure Receiver Adapter

We have two receiver systems.

1. Fieldglass – to upload Fieldglass data.

All the configurations required are passed from the calling iflow. There is an option to configure some of these if required.

The screenshot shows the 'Receiver' configuration page. Under 'Connection', the 'Adapter Type' is set to 'HTTP'. Other settings include 'Query', 'Proxy Type' (set to 'Internet'), 'Method' (set to 'Dynamic'), 'Send Body' (unchecked), 'Authentication' (set to 'OAuth2 Client Credentials'), 'Timeout (in ms)' (set to 60000), and checkboxes for 'Throw Exception On Failure' (checked), 'Attach Error Details on Failure' (checked), and 'Retry Failed Requests' (unchecked). Request and Response Headers are also defined.

Figure 10: HTTP Receiver adapter configurations

Configure "CI to Fieldglass Generic Upload"

The screenshot shows the 'More' tab configuration for the 'CI to Fieldglass Generic Upload' receiver. The 'Type' dropdown is set to 'All Parameters' and the 'ExceptionPayload' field contains 'ExceptionPayload'.

Figure 11: other parameters

2. Mail Server – to send exception emails.

The option to send emails is set as part of the calling iflow. **Send_Exception_Email** Header parameter determines if an email is sent or not.

Exception handling section can be used to forward the error by email:

Below are the parameters to be configured.

Configure "CI to Fieldglass Generic Upload"

Receiver	More
Receiver:	SMTP
Adapter Type:	Mail
Connection	
Address:	smtp.mail.com
Proxy Type:	Internet
Timeout (in ms):	30000
Protection:	STARTTLS Optional
Authentication:	None
Processing	
From:	test@test.com
To:	test@test.com

Figure 12: Mail Receiver adapter configurations

As part of the Exception handling the following information is captured:

- the Tenant name
- calling Integration Flow name
- current standard Integration Flow name
- response status code,
- the error message as returned by FieldGlass

The error can be caught in the calling program for further processing, like logging, sending it by an email, writing to a message queue, etc.

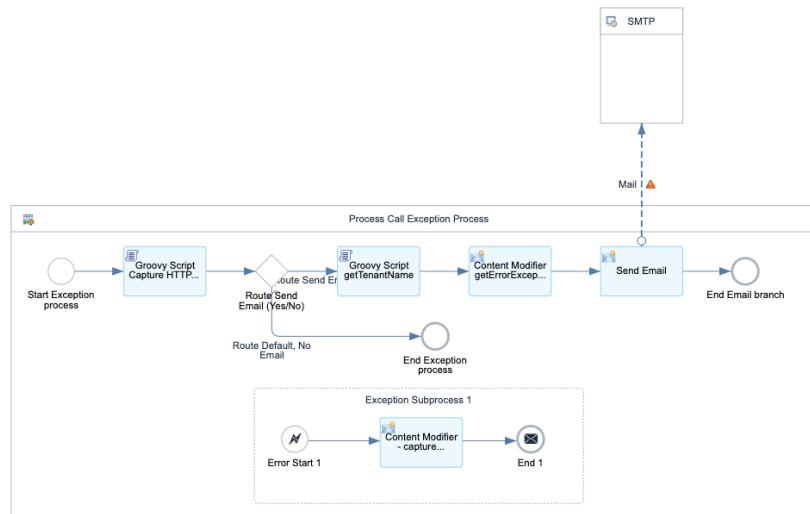


Figure 13: Exception Handling process flow

```

Hi Team,

Message failed in CI [ Environment:it-accd003 ]
calling iFlowName: FieldGlass_F _CallingiFlow
iFlowName: FieldGlass_Upload_ ...
Message processing Log ID : ...

Time:2024:10:23 05:53:42
Error Message : HTTP operation failed invoking https://.../api/vc/connector/Supplier%20Plant%20Association%20Upload with statusCode: 400
Response Body:
<xml version="1.0" encoding="UTF-8"?>
<Status>
<TransactionID> ... </TransactionID>
<ReturnCode>106</ReturnCode>
<Message>This file has been uploaded more than 5 times within 24 hours. Retry later. Previous Transaction ID: ... </Message>
</Status>

Thanks,
CI Team

```

Figure 14: Sample FieldGlass Error message