TIBCO ActiveMatrix™ BusinessWorks™ Plug-in for EDI User's Guide

Software Release 6.3

March 2016

Two-Second Advantage®

TIBCO®

Important Information

Contents

TIBCO Documentation and Support Services

Documentation for this and other TIBCO products is available on the TIBCO Documentation site:

https://docs.tibco.com

Documentation on the TIBCO Documentation site is updated more frequently than any documentation that might be included with the product. To ensure that you are accessing the latest available help topics, please visit us at https://docs.tibco.com.

Product-Specific Documentation

The following documents for this product can be found on the TIBCO Documentation site:

- TIBCO ActiveMatrix™ BusinessWorks™ Plug-in for EDI Installation
- TIBCO ActiveMatrix™ BusinessWorks™ Plug-in for EDI Release Notes
- TIBCO ActiveMatrix™ BusinessWorks™ Plug-in for EDI User's Guide

Other TIBCO Product Documentation

When working with the TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI, you may find it useful to read the documentation for the following TIBCO products:

- TIBCO ActiveMatrix BusinessWorks™
- TIBCO Foresight® Instream®
- TIBCO Foresight® Translator

How to Contact TIBCO Support

For comments or problems with this manual or the software it addresses, contact TIBCO Support:

- For an overview of TIBCO Support, and information about getting started with TIBCO Support, visit this site:

  http://www.tibco.com/services/support

- If you already have a valid maintenance or support contract, visit this site:

  https://support.tibco.com

  Entry to this site requires a user name and password. If you do not have a user name, you can request one.

How to Join TIBCOmmunity

TIBCOmmunity is an online destination for TIBCO customers, partners, and resident experts. It is a place to share and access the collective experience of the TIBCO community. TIBCOmmunity offers forums, blogs, and access to a variety of resources. To register, go to:

https://www.tibcommunity.com

EDI Plug-in Overview

This chapter gives an overview of TIBCO Foresight products and TIBCO ActiveMatrix™ BusinessWorks Plug-in for EDI.

Introduction

TIBCO ActiveMatrix™ BusinessWorks Plug-in for EDI is available in two versions: Standard Edition and Healthcare Edition. HIPAA code table validation is only available in the Healthcare Edition and its bundled Healthcare Editions of TIBCO Foresight® Instream® and TIBCO Foresight® Translator.

TIBCO ActiveMatrix™ BusinessWorks is an easy to use integration product suite for enterprise, web, and mobile applications. It uses the Eclipse graphical user interface (GUI) for defining business processes and the process engine to execute them. TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI plugs into TIBCO ActiveMatrix™ BusinessWorks, and provides access to TIBCO Foresight tools for validation and translation of Electronic Data Interchange (EDI).

TIBCO ActiveMatrix™ BusinessWorks supports plug-ins that extend palette functionality. After installing the plug-in, an EDI Palette becomes available in TIBCO Business Studio™. You can add the plug-in activities to the business processes you are designing, and integrate them into the process flow. At runtime, the plug-in activities are executed as part of the TIBCO ActiveMatrix™ BusinessWorks process execution.

The EDI plug-in allows users to:

- validate data with Instream®
- generate response documents with Instream's Response Generator functionality
- split data into valid and invalid types with Instream's Document Splitter functionality
- translate data with Foresight® Translator
- utilize additional capabilities to refine output such as prescanning data, incorporating errors, swapping data, modifying errors, and more.

Refer to TIBCO Foresight Overview for more detailed information about these activities.

TIBCO Foresight Overview

This document focuses on the transaction validation, response generation, document splitting, translation and other functions provided by TIBCO Foresight products as part of the EDI Plug-in.

Before using this document, familiarize yourself with Electronic Data Interchange (EDI). See Electronic Data Interchange for more information.

TIBCO Foresight products benefit customers by connecting partners, validating transactions, reducing administrative inefficiencies and addressing mandates such as HIPAA 5010. The TIBCO Foresight family of EDI management solutions allows organizations to trade electronic transactions with commonly-used formats (EDI, XML, and proprietary Flat File) and standards (HIPAA, X12, EDIFACT, and so on).

The EDI Plug-in provides access to the following TIBCO Foresight tools:

- Validation with Instream, including:
  - ResponseGenerator
  - DocumentSplitter
  - DataSwapper
- Translation with Translator

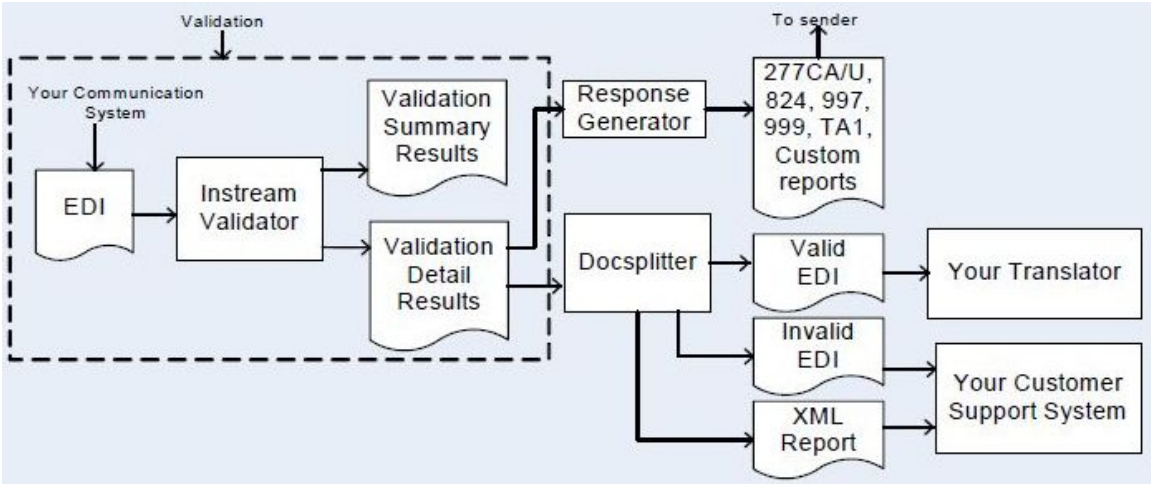
- Additional functions:
  - pre-scanning of EDI and EDIFACT data to parse document header information using X12 or EDIFACTPrescanners
  - modification of error lines (also called DTL lines) in an Instream results file using ISErrorRefiner
  - incorporation of error records and other identifying information into data using Shuffler.

Transaction Validation with Instream

Instream ensures the compliance of inbound and outbound data using the fastest and most thorough transaction validation engine.

You can use Instream to automate transaction flow throughout your organization and validate transactions according to industry standards, organizational guidelines, and specific business rules.

Instream is designed to be highly configurable, with several components that work together. The following figure shows Instream, Response Generator, and Document Splitter working together to process inbound EDI.



Instream validation can be run as a command line, usually from within a batchfile or script. Or you can integrate Instream validation into other applications, either statically (C/C++) or dynamically (C/C++, C#, and Java).

Reference

See *Instream Validation Technical Manual* PDF document for more information about the Instream validation function.

ResponseGenerator

Response documents (also called acknowledgments) are used to give feedback to the sender of a transaction on the status of the acceptance of the transaction by its recipient. After validation, the Response Generator function can be used to generate a response document for transmission to the sender.

Response Generator generates the following response types:

- For X12 data
  - 277CA Health Care Claim Acknowledgment (277CA)
  - 277U Unsolicited Health Care Claim Status Notification (277U)

  - 824 Application Advice
  - 997 Functional Acknowledgment
  - 999 Implementation Acknowledgment
  - TA1 Interchange Acknowledgment
  - Custom report
- For EDIFACT data
  - CONTRL document

References

- See Response Documents (Acknowledgments) for more information on the response documents available through Response Generator.
- See *Response Generator Technical Manual* PDF document for more information about the response generation function.

DocumentSplitter

After validation, the Document Splitter function can be used to split data into two types: valid and invalid. For example, Docsplitter can sift out invalid claims in an 837 so that you can process the valid ones and notify the sender of the invalid ones.

Document Splitter can be used with the following data types:

- HIPAA EDI
- X12
- EDIFACT
- Flat File
- XML file

Reference

See *Document Splitter Technical Manual* PDF document for more information about the document splitting function.

DataSwapper

After validation, Data Swapping can be used to substitute values in EDI data.

Dataswapper can perform the following functions:

- Replace the value in a specified data element with another value in the EDI
- Insert a new segment into the EDI
- Delete a segment from the EDI
- Create a report of the data that was changed.

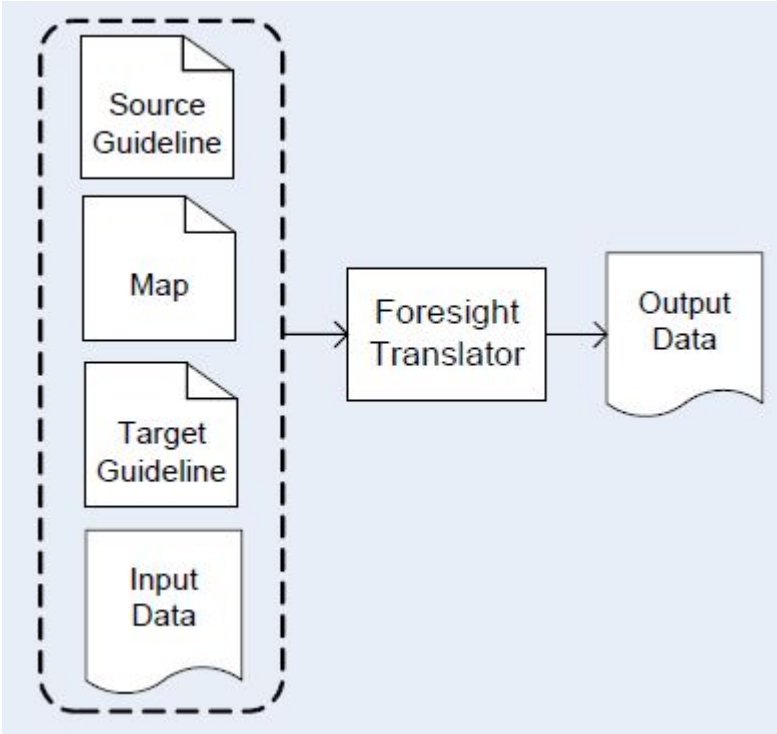
Reference

See *Dataswapper* PDF document for more information about the data swapping function.

Translator

Translator is a specialized, high-speed transformation engine that enables the mass conversion of transaction files based on pre-built or custom maps, providing direct translation to and from EDI, XML, and Flat File formats with no interim staging required.

You can use Translator to translate the input data in one format to a new file in another format. The following figure shows the translation workflow.



Reference

See *Translator* PDF document for more information about the translation function.

Additional Functions

The EDI Plug-in provides additional activities useful for processing EDI.

Prescanning

The EDI Plugin provides activities that perform prescanning on X12 and EDIFACT data. This allows information to be gathered such as transaction type and trading partner group, which that can then be used to proceed with processing the data in an appropriate manner.

X12Prescanner

The X12Prescanner parses the X12 document header ISA, GS, and ST segments.

Example

If the X12 data contains a GS08 value of 005010X279A1 **AND** a GS01 value of HE, Instream should use the 271-X279 guideline for validation.

EDIFACTPrescanner

The EDIFACTPrescanner parses the EDIFACT document header UNA, UNB, UNG, and UNH segments.

Example

If the EDIFACT data contains a UNG07-01+02 (or, if there is no UNG, the UNH02-02+03) value of D:96A **AND** a UNB1-1 value of 3, Instream should use the D96A guideline for validation.

CMS Compliance

The EDI Plugin provides two activities for use by Centers for Medicare and Medicaid Services (CMS) customers utilizing data that has been validated by Instream to convert EDI to and from flat file format for compliance with the CMS-supplied Claims Edit Module (CEM) requirements for electronic processing of claims.

The Shuffler and ISErrorRefiner activities are available **only** to customers who have purchased the TIBCO Foresight® Instream® MAC Adapter. Contact your TIBCO Foresight Account Representative for more information about Instream® MAC Adapter.

Shuffler

Shuffler enhances detail results from Instream by "shuffling" STC records into 837 and 276 flat file documents to identify errors found in front-end validation. For each error found in the original EDI file, Shuffler adds STC records after the flat file segment in error. The STC information added for each error is as defined in the CMS 837 and 276 spreadsheets.

See the *TIBCO Foresight® Instream® MAC Adapter Introduction* PDF document for more information about Shuffler.

ISErrorRefiner

Errors in Instream are assigned a severity, a HIPAA Type, various response codes, and more that apply to the error wherever it occurs. However, a customer may prefer that an error have different values for these parameters, depending on the location of the error within the EDI.

In this case, ISErrorRefiner is used to modify error lines (i.e., DTL lines) in the results file, based on a configuration table of modification instructions.

See the *TIBCO Foresight® Instream® MAC Adapter ISErrorRefiner* PDF document for more information about ISErrorRefiner.

Getting Started

A typical workflow using the TIBCO ActiveMatrix™ BusinessWorks Plug-in for EDI includes creating a project, designing a process, and deploying the application.

TIBCO ActiveMatrix™ BusinessWorks enables users to create services and integrate applications, and deploy them at runtime. It uses the Eclipse graphical user interface (GUI) for defining business processes and the process engine to execute them.

To design a process and deploy it at runtime, refer to the following:

1. Creating a Project
2. Designing a Process
3. Testing an Application
4. Deploying an Application

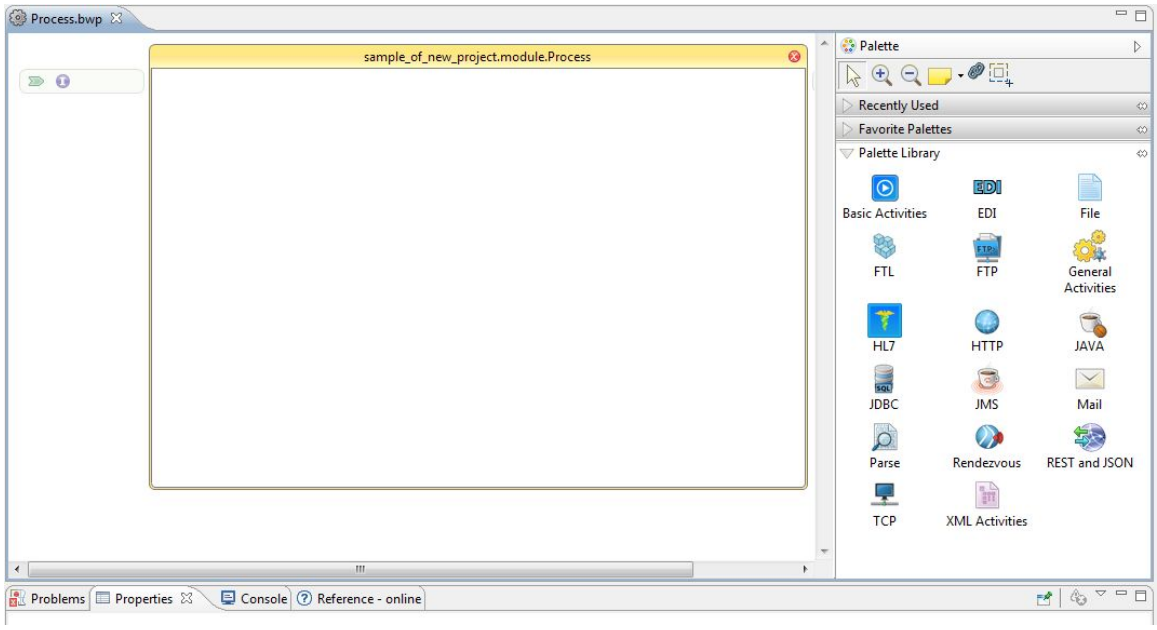
See *TIBCO ActiveMatrix BusinessWorks™ Application Deployment* for more detailed information.

Creating a Project

TIBCO ActiveMatrix™ BusinessWorks application modules are Eclipse projects that are created in TIBCO Business Studio™, which contain various resources.

Procedure

1. Start TIBCO Business Studio™.
2. Select **File** > **New** > **BusinessWorks Resources**.
3. Click the **BusinessWorks Application Module** resource in the BusinessWorks Resource. Click **Next**.
4. Type a name for the project that you are creating in the **Project name** field.
5. Select the **Create empty process** check box and type a process name in the **Name** field.
6. Select the **Create Application** check box and type an application name in the **Name** field. Click **Finish**.
   A project with the default settings is opened in the Project Explorer view.



Designing a Process

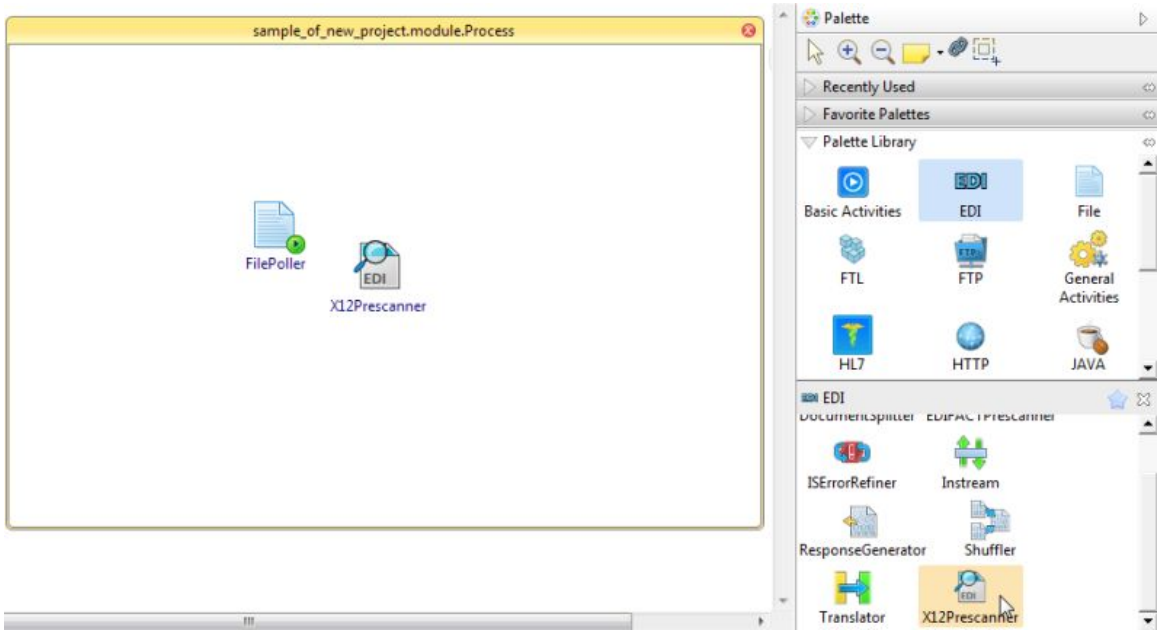
Processes capture and manage the flow of business information in an enterprise between different data sources and destinations. You can design a process by using activities and adding conditions.

By default, an empty process is created when Creating a Project with the **Create empty process** check box selected.

See *TIBCO ActiveMatrix™ BusinessWorks Application Deployment* for more details about creating processes.

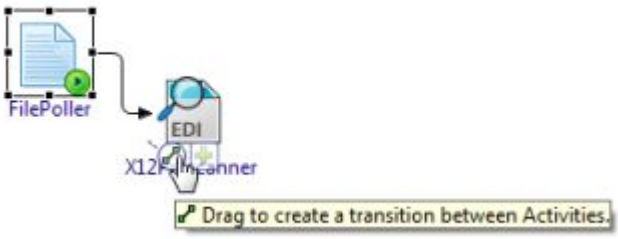
Procedure

1. In the Process editor, select and drop an activity from the Palette view.
   For example, select and drop the FilePoller activity from the **File** palette and the X12Prescanner activity from the **EDI** palette.



You can also select activities from the **Context** menu.

2. Click the icon to create links between the activities.



3. Configure the added activities.
4. Click **File** > **Save** to save the process.

Testing an Application

An application contains an application module that is defined in TIBCO Business Studio™. After you design your business process, you can run and debug the configurations.

TIBCO Business Studio™ has a built-in debugger that allows users to debug the design-time configurations.

Prerequisites

Ensure that you have created an application and designed a process before testing.

By default, a process and an application are created when Creating a Project with the **Create empty process** and **Create Application** check boxes selected.

See *TIBCO ActiveMatrix BusinessWorks™ Application Deployment* for more details about creating processes and applications.

Procedure

1. Click **Run** > **Run** to run the application.
2. Optional: Click **Run** > **Debug** to debug the application.
   The perspective changes to Debug from Modeling. You can view the job details in the Console view from the Debug perspective.

Deploying an Application

You can manage TIBCO ActiveMatrix™ BusinessWorks applications by using TIBCO® Enterprise Administrator after deploying the applications.

Prerequisites

An enterprise archive (EAR) file must be generated before deploying an application. See Generating an EAR File for details about generating an EAR file.

A complete workflow of deployment includes:

1. Building an EAR file.
2. Uploading the EAR file.
3. Deploying the EAR file.
4. Starting the application.

You can deploy an application EAR file from TIBCO Business Studio™, or by using the command-line mode with the **bwdesign** utility. See *TIBCO ActiveMatrix BusinessWorks™ Administration* for more details about how to deploy an application.

Generating an EAR File

Application archives are the enterprise archive (EAR) files that are created in TIBCO Business Studio™. An EAR file is required when deploying an application.

Prerequisites

An application project has already been created, as explained in Creating a Project.

Procedure

1. Go to File Explorer and click the Open Directory to Browse  icon.
2. Select the folder where you want to generate the EAR file and click **OK**.
   The new folder is displayed in the File Explorer view.
3. Drag the application from the Project Explorer to the new folder in the File Explorer.

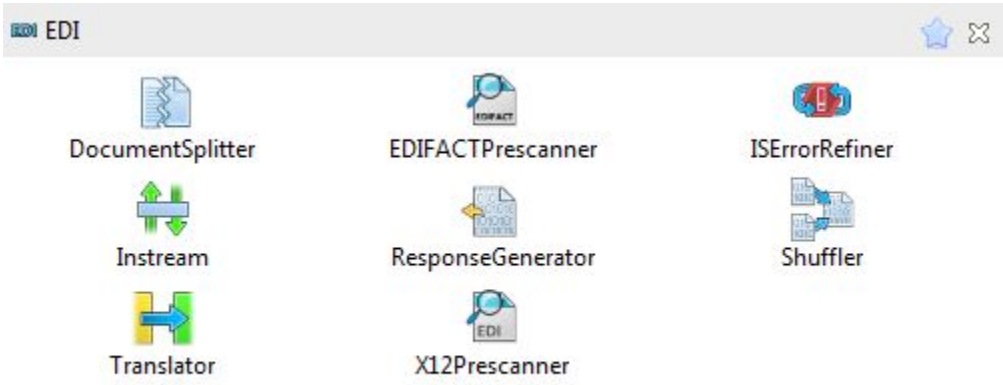
   The EAR file is generated with the name `<application>_<version>.ear`.

EDI Palette Overview

The EDI Palette contains EDI activities for TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI.

One of the key activities in integrating health care systems is to implement the Interface Engine functionality using the TIBCO ActiveMatrix BusinessWorks™ Process Engine in the course of defining process models. This process modeling is done in TIBCO Business Studio™ using the EDI activities, as well as the many features that are part of TIBCO ActiveMatrix BusinessWorks™. These include FTP, the file poller, and logging support.

The EDI palette, found on the Palettes panel, includes eight activities.



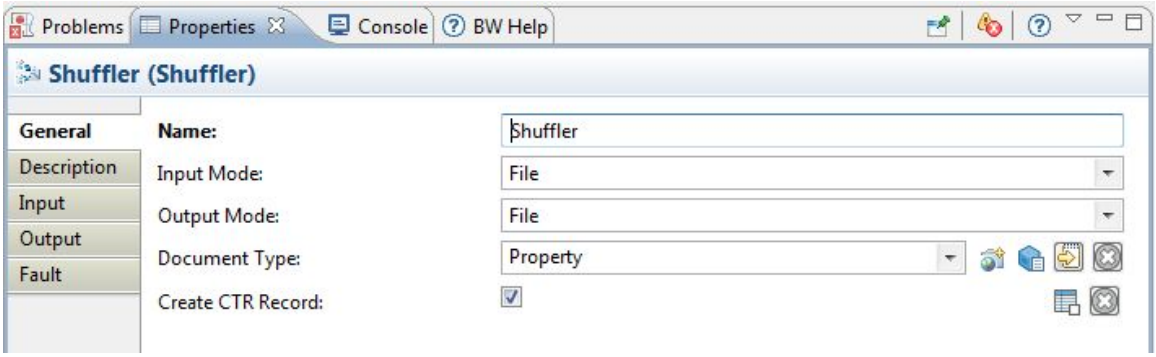
The following activities are available on your EDI integration project:

- DocumentSplitter
- EDIFACTPrescanner
- ISErrorRefiner
- Instream
- ResponseGenerator
- Shuffler
- Translator
- X12Prescanner

Configuring Activities

When an activity has been added to the workspace, the activity can be configured by clicking on the activity icon.

The properties view area below the workspace is populated with fields associated with the activity. Each activity has configuration tabs on the left, and a corresponding configuration input area on the right.



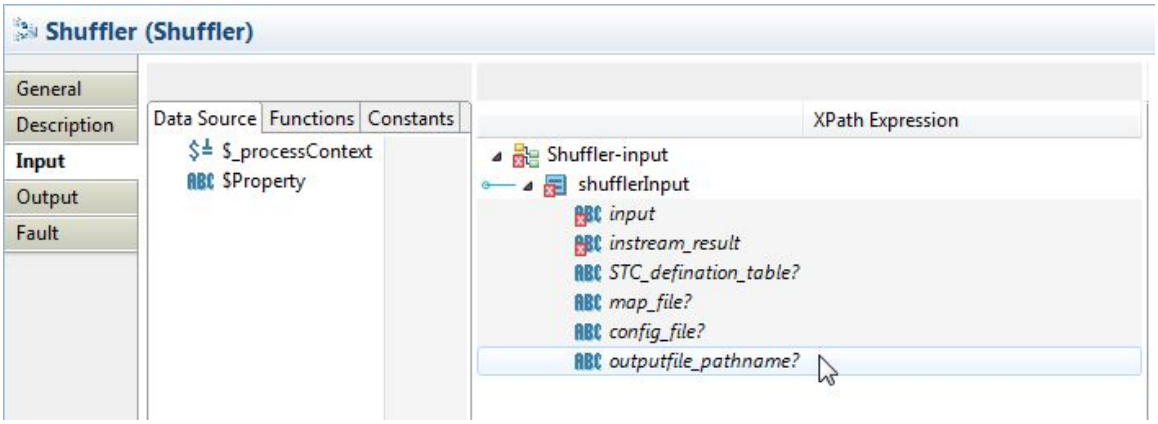
The configuration properties are grouped under tabs such as General, Description, Input, Output, and so on. For example, upon adding a Shuffler activity, you can configure it by specifying the values for the properties under the tabs General, Description, and Input, and view results using Output and Fault.

See Working with Standard Activity Features in the *TIBCO ActiveMatrix BusinessWorks™ Application Development* document for details.

Configuration Using XPath

Input, Output, Advanced, and Fault information is configured and displayed using the XPath Builder. XPath uses path expressions to navigate through XML documents. XPath also has basic manipulation functions for strings, numbers, and Booleans.

TIBCO ActiveMatrix BusinessWorks™ uses XPath as the language for defining conditions and transformations.



See XPath in the *TIBCO ActiveMatrix BusinessWorks™ Application Development* document for details.

DocumentSplitter

Use the DocumentSplitter activity to split validated data into valid and invalid data.

For more details on Document Splitter, refer to the *Document Splitter Technical Manual* PDF document.

Docsplitter Output

Use of the DocumentSplitter activity results in the following output:

- A file containing the **valid** data
- A file containing the **invalid** data

- An XML or delimited report containing the status of each claim or other data being split, plus additional information.

Note the following about DocumentSplitter output:

- When using content-based splitting or split-point grouping, DocumentSplitter will create multiple split files.
- For EDI, output includes a CR/LF after the segment terminator unless:
  - The segment terminator in the input file is a LF
  - You include the following section in a setup file

```
[Options]
OutputEDIWithCRLF=0
```

- DocumentSplitter fixes the output file's segment counts by default.

Split Points

Documents can be split at various point in the data. Basic split points are as follows:

Data Type	Split Points
HIPAA	Vary
X12 non-HIPAA	ST ISA,GS
EDIFACT	UNH UNB, UNG
Flat File	Top of layout
XML	Any repeating complex element

For a complete listing of where documents can split, see Appendix B: Split Points in the *Document Splitter Technical Manual* PDF.

General

The **General** tab contains the following fields.

Field	Literal Value/ Module Property/ Process Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
Input Mode	No	Specifies where the input data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.

Field	Literal Value/ Module Property/ Process Property?	Description
Output Mode	No	Specifies where the output data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.
Report Type	No	This drop-down list allows you to specify what report types are generated. • Valid data • Invalid data • Both (Default) • No report
Report Format	No	This drop-down list allows you to specify the format of the generated report. • XML This report contains the status of each claim or other data being split, plus other information.
Ignore Output File Name	No	DocumentSplitter outputs multiple files, resulting in large amounts of data. Select this checkbox to ignore output file names and reduce the volume of unnecessary data. (Default = Checked)

Description

The **Description** tab is used to provide a short description for the activity.

Input

The **Input** tab contains the following fields.

Field	Datatype	Description
input	string	Specifies the file path and name of the input data file.
instream_result	string	Specifies the file path and name of the Instream detail file to be split. (Required)  This file is not changed by DocumentSplitter.
output_directory	string	Specifies directory path for the output (split) files. (Required)

Field	Datatype	Description
profile	string	Full path and name to a Document Splitter INI file.  The parameters set in the Document Splitter INI file take precedence over those set in the Input tab and the DocumentSplitter tab.
tpa_file	string	Path to a Trading Partner Automation CSV file that contains a pointer to a setup file. All sections in the setup file are processed. See the *Trading Partner Automation* PDF document. If there is a conflict between the command line and the settings in the setup file selected by TPA, the last setting processed prevails.
parameters	Yes	You can specify parameters listed in the *Document Splitter Technical Manual* PDF document in this field. For example: • `-wb` Swap data in both valid and invalid output file(s) according to the rules in the validation guideline.

Output

The **Output** tab contains the following fields.

Field	Datatype	Description
report	string	Specifies the full path and name of the Document Splitter report, which contains the status of each claim or other data being split, as well as other information.
files	string	output_data – Body of report.
	boolean	is_valid_document? – Document is valid (1) or document is not valid (0).
	string	content_name – Specifies the name of the contents if content-based splitting is used.

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Error Schema Element	Datatype	Description
msg	string	Error message description.
msgCode	string	The error code. It represents TIBCO ActiveMatrix BusinessWorks Plug-in for EDI and TIBCO Foresight errors. See Error Codes for details.

EDIFACTPrescanner

Use the EDIFACTPrescanner activity to parse EDIFACT document header (UNA/UNB/UNG/UNH) segments and put each element in separate output fields.

Example

Prescanning allows data to be scanned for information such as transaction type and trading partner group. This information can then be used to proceed with processing data in an appropriate manner. For example: If the EDIFACT data contains a UNG07-01+02 (or, if there is no UNG, the UNH02-02+03) value of D:96A **AND** a UNB1-1 value of 3, Instream should use the D96A guideline for validation.

General

The **General** tab contains the following fields.

Field	Literal Value/ Module Property/ Process Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
Input Mode	No	Specifies where the input data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.

Description

The **Description** tab is used to provide a short description for the activity.

Input

The **Input** tab contains the following fields.

Field	Datatype	Description
input	string	Specifies the file path and name of the input data file. (Required)
encoding	string	Specifies encoding type: UTF-8, UTF-16, UTF-16BE, UTF-16LE, ISO-8859-1.

Output

The **Output** tab contains the following fields.

Field	Sub-Field	Additional Sub-Fields	Datatype
UNA	ComponentDataElementSeperator		string
	DataElementSeperator		string
	DecimalNotification		string
	ReleaseIndicator		string
	RepeatElementSeperator		string
	SegmentTerminator		string
UNB	SyntaxIdentifier	SyntaxIdentifier	string
		SyntaxVersionNumber	integer
		ServiceCodeListDirectoryVersionNumber	string
		CharacterEncodingCoded	string
		SyntaxReleaseNumber	string
	InterchangeSender	SenderIdentification	string
		PartnerIdentificationCodeQualifier	string
		AddressForReverseRouting	string
		InterchangeSenderInternalSubIdentification	string
	InterchangeRecipient	RecipientIdentification	string
		PartnerIdentificationCodeQualifier	string
		RoutingAddress	string
		InterchangeRecipientInternalSubIdentification	string
	DateandTime	DateOfPreparation	integer
		TimeOfPreparation	integer

Field	Sub-Field	Additional Sub-Fields	Datatype
	InterchangeControlReference		string
	RecipientsReferencePassword	Password	string
		PasswordQualifier	string
	ApplicationReference		string
	ProcessingPriorityCode		string
	AcknowledgmentRequest		integer
	CommunicationsAgreementID		string
	TestIndicators		integer
UNG	FunctionalGroupIdentification		string
	SenderIdentifiation	SenderIdentification	string
		PartnerIdentificationCodeQualifier	string
	RecipientsIdentification	RecipientsIdentification	string
		PartnerIdentificationCodeQualifier	string
	DateandTime	DateOfPreparation	integer
		TimeOfPreparation	integer
	FunctionalGroupReferenceNumber		string
	ControllingAgency		string
	MessageVersion	MessageTypeVersionNumber	string
		MessageTypeReleaseNumber	string
		AssociationAssignedCode	string

Field	Sub-Field	Additional Sub-Fields	Datatype
	ApplicationPassword		string
UNH	MessageReferenceNumber		string
	MessageIdentifier	MessageTypeIdentifier	string
		MessageTypeVersionNumber	string
		MessageTypeReleaseNumber	string
		ControllingAgency	string
		AssociationAssignedCode	string
	CommonAccessReference		string
	StatusOfTheTransfer	SequenceMessageTransferNumber	string
		FirstLastSequenceMessageTransferIndication	string
FileInfo	FileName		string
	FileSize		integer

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Error Schema Element	Datatype	Description
msg	string	Error message description.
msgCode	string	The error code. It represents TIBCO ActiveMatrix BusinessWorks Plug-in for EDI and TIBCO Foresight errors. See Error Codes for details.

Instream

Use the Instream activity to validate EDI for many different file formats, including (but not limited to) HIPAA (health care) data, other X12 formats, EDIFACT, flat files, and XML.

The Instream activity allows the plug-in to validate the input EDI data, split validation results, swap elements in data, and use validation detail results to generate responses.

Support Protocols

The following list shows the protocols supported by the Instream activity:

- X12
- HIPAA
- EDIFACT
- Flat File

Features

The following list shows the features provided by the Instream activity:

1. Validation

   - creates two files or an output stream: detail results and summary results
   - supports the Instream CallBack function.

   See *Instream Validation Technical Manual* PDF document for more information about the validation function.

2. DocumentSplitter

   - uses validation results to separate the valid EDI from the invalid
   - generates a list of files containing valid data, a list of files containing invalid data, and a report containing the status of each claim or other data being split based on the split point grouping specified in the DocumentSplitter profile.

   You can create your own profile to split the EDI into multiple documents. See *Document Splitter Technical Manual* PDF document for details about the Document Splitter function.

3. ResponseGenerator

   - uses validation results to create responses, such as 277 Acknowledgment, 824 Application Advice, 997/999 Functional Acknowledgment, TA1 Interchange Acknowledgment, and CONTRL document.

   See *Response Generator Technical Manual* PDF document for details about the Response Generator function.

4. DataSwapper

   - replaces the value in a specified data element with another value in the EDI
   - inserts a new segment into the EDI
   - deletes a segment from the EDI
   - creates a report of the data that was changed.

   See *DataSwapper* PDF document for details about the DataSwapper function.

Examples

Refer to the associated examples: EDI Plug-in Example Processes.

General

The **General** tab contains the following fields.

For more details on Instream, refer to the *Instream Validation Technical Manual* PDF document.

Field	Literal Value/ Module Property/ Process Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
Operation Type		Specify an operation type: • Instream only • Instream and ResponseGenerator • Instream and DocumentSplitter • Instream, ResponseGenerator and DocumentSplitter • Instream and DataSwapper
Input Mode	No	Specifies where the input data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.
Output Mode	No	Specifies where the output data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.
Stop Validation on IC Error	No	Check this checkbox to indicate that validation should be stopped if an error occurs in the Interchange.
Document Level Validation	No	 This functionality is not applicable when the CallBack shared resource is enabled, and it is applicable only for X12 protocol. Check this checkbox to validate the EDI data that has no enveloping (EDI does not have ISA or GS enveloping). When the incoming EDI data contains ISA or GS enveloping, the enveloping will be ignored if you check this checkbox. If selected, you must also enter Document Level Validation Separators on the Input > Functions tab, using the "document_only_parameters" field. See Electronic Data Interchange for details about EDI enveloping.

Field	Literal Value/ Module Property/ Process Property?	Description
CallBack	No	Select this checkbox to use a Java Class for callback (optional). See Instream CallBack for more complete information on Enabling Callbacks.
Class Name	No	Specify a Java Class. This field is enabled when CallBack is checked.  Create a new Java Class.  Browse to an existing Java Class using the Type selection dialog. See Instream CallBack for more complete information on Enabling Callbacks.

Description

The **Description** tab is used to provide a short description for the activity.

DocumentSplitter

The **DocumentSplitter** tab contains the following fields.

For more details on Document Splitter, refer to DocumentSplitter.

Field	Literal Value/ Module Property/ Process Property?	Description
Report Type	No	Refer to DocumentSplitter, General - Report Type Field.
Report Format	No	Refer to DocumentSplitter, General - Report Format Field.
Ignore Output File Name	No	Refer to DocumentSplitter, General - Ignore Output File Name Field.

ResponseGenerator

The **ResponseGenerator** tab contains the following fields.

For more details on Response Generator, refer to the *Response Generator Technical Manual* PDF document.

Output Item	Literal Value/ Module Property/ Process Property?	Description
Generate 277	No	Refer to ResponseGenerator, General - Generate 277 Field.
Generate 824	No	Refer to ResponseGenerator, General - Generate 824 Field.
Generate 997	No	Refer to ResponseGenerator, General - Generate 997 Field.
Generate 999	No	Refer to ResponseGenerator, General - Generate 999 Field.
Generate TA1	No	Refer to ResponseGenerator, General - Generate TA1 Field.
Generate Control	No	Refer to ResponseGenerator, General - Generate Control Field.
Generate Custom Report	No	Refer to ResponseGenerator, General - Generate Custom Report Field.
Custom Report Template	No	Refer to ResponseGenerator, General - Custom Report Template Field.
Parameters	Yes	Refer to ResponseGenerator, General - Response Parameters Field.

DataSwapper

The **DataSwapper** tab contains the following fields.

For more details on DataSwapper, refer to the *DataSwapper* PDF document.

Field	Literal Value/ Module Property/ Process Property?	Description
Setup File	Yes	Specifies the path to the DataSwapper setup file. Refer to the *DataSwapper* PDF document for information on how to create a DataSwapper setup file.
Insert After Segment	No	Check this checkbox to place inserted segments after the segment containing a GenerateFSUID or InsertSegment business rule. If unchecked, the insertion will be before the segment that contains the business rule.

Input

The **Input** tab contains the following fields.

For more details on Instream, refer to the *Instream Validation Technical Manual* PDF document.

Field	Datatype	Description
instream		
input	string	Specifies the full path and name of the input file.
guideline_name	string	The name of the guideline used for validation, or used as the input for Document Splitter.  This field will not be displayed when the CallBack shared resource is enabled, because the guideline will be extracted from the CallBack code.
output_directory	string	This field appears only when File mode is selected as the output mode in the General tab. You need to specify the output directory for the output files.
instream_profile	string	(Optional) Specifies which APF file to use. The APF file contains configuration information for the Instream activity. Each time you validate, an APF file is read to determine what is to be checked and what is written to output files. The default APF is $fsdeflt.apf and is located in the *TIBCO_FORESIGHT_HOME* / *Instream-Translator* /bin directory. **Note:** This field will not be displayed when the CallBack shared resource is enabled, because the APF file will be extracted from the CallBack code.
document_only_parameters	string	If the Document Level Validation checkbox is selected on the General tab, Separators (Segment Terminator, Element Separator, and Component Element Separator, in this order) must be specified in this field in one of the following formats: • character, for example: ~*: • integer, for example: 29, 30, 31 • hex, for example: 0x1E, 0x1F, 0x1D
documentSplitter		
profile	string	Refer to DocumentSplitter, Input - profile Field.
tpa_file	string	Refer to DocumentSplitter, Input - tpa_file Field.
parameters	string	Refer to DocumentSplitter, Input - parameters Field.

Field	Datatype	Description
output_directory	string	Refer to DocumentSplitter, Input - output_directory Field.
responseGenerator		
profile		Refer to ResponseGenerator, Input - profile Field.
tpa_file		Refer to ResponseGenerator, Input - tpa_file Field.
parameters		Refer to ResponseGenerator, Input - parameters Field.
output_directory	string	Refer to ResponseGenerator, Input - output_directory Field.
dataSwapper		
profile		Specifies the path to the DataSwapper setup file.  Refer to the *DataSwapper* PDF document for information on how to create a DataSwapper setup file.
output_directory	string	Output directory for the output files.  This field only appears when File mode is selected as the output mode in the Configuration panel. You need to specify the output directory for the output files.

Output

The result and the summary report are listed in the Output tab.

The **Output** tab contains information grouped by function and presented in the order they are displayed on the GUI.

In File output mode, when you run a process with the Instream activity twice using the same input file and output location, an error is thrown indicating that the output files already exist. Make sure that you move all output files to an appropriate location after every validation to avoid this error.

instream Output Items

Output Field	Mode Selected	Output Item	Datatype	Description
instream				

Output Field	Mode Selected	Output Item	Datatype	Description
instream_result	File output	validation_resultFile	string	Specifies the full path and name of the validation detail results file (in the Flat File format), which contains general messages and statistics, and describes errors and warnings found during validation.
	Memory output	validation_result_in _flatfile	string	Specifies the message of the validation detail results in the Flat File format.
instream_summary	File output	validation_summaryFile	string	Specifies the full path and name of the validation summary file (in the Flat File format), which contains a summary of the validation, including validation start time, end time, number of errors, warnings, and other messages.
	Memory output	validation_summary_in _flatfile	string	Specifies the message of the validation summary results in the Flat File format.
total_of_errors	N/A	total_of_errors	integer	Records the total errors with a severity level larger than level 2 (Error Count, Fatal Count, User1 Count, and User2 Count). The number of errors for each severity is recorded in the SVRTY record in the summary file. See *Instream Validation Technical Manual* PDF document for details.

Output Field	Mode Selected	Output Item	Datatype	Description
return_code	N/A	return_code	integer	Specifies the Instream return code. For example, 100 means the validation ran successfully, 133 means the database directory cannot be found, and so on. Return codes are listed in Appendix A of *Instream Validation Technical Manual* PDF document.
MessageCountBy Severity	N/A		integer	
MessageCountBy Type	N/A		integer	
Response Generator				
rg_277_filename	File output		string	Specifies the full path and name of the 227 CA/U Acknowledgment.
rg_277	Memory output		string	Provides the content of the 277 CA/U Acknowledgment.
rg_824_filename	File output		string	Specifies the full path and name of the 824 Application Advice.
rg_824	Memory output			Provides the content of the 824 Application Advice.
rg_997_filename	File output		string	Specifies the full path and name of the 997 Functional Acknowledgment.
rg_997	Memory output			Provides the content of the 997 Functional Acknowledgment.
rg_999_filename	File output		string	Specifies the full path and name of the 999 Implementation Acknowledgment.
rg_999	Memory output			Provides the content of the 999 Implementation Acknowledgment.

Output Field	Mode Selected	Output Item	Datatype	Description
rg_TA1_filename	File output		string	Specifies the full path and name of the TA1 Interchange Acknowledgment.
rg_TA1	Memory output			Provides the content of the TA1 Interchange Acknowledgment.
rg_CONTRL_DOC_ filename	File output		string	Specifies the full path and name of the CONTRL document.
rg_CONTRL_DOC	Memory output			Provides the content of the CONTRL document.
rg_custom_report_ filename	File output		string	Specifies the full path and name of the custom report.
rg_custom_report	Memory output			Provides the content of the custom report.
documentSplitter				
ds_report_filename	File output		string	Specifies the full path and name of the DocumentSplitter report, which contains the status of each claim or other data being split, as well as other information.
ds_rpt_valid_results	File output		string	Contains a list of valid files (ds_rpt_valid_filename). The ds_rpt_valid_filename output specifies the full path and name of the valid EDI file.  When splitting the EDI based on the split point grouping specified in the DocumentSplitter profile, multiple valid files will be generated if the input data contains multiple segments.

Output Field	Mode Selected	Output Item	Datatype	Description
ds_rpt_invalid_results	File output		string	Contains a list of invalid files (ds_rpt_invalid_filename). The ds_rpt_invalid_filename output specifies the full path and name of the invalid EDI file.  When splitting the EDI based on the split point grouping specified in the DocumentSplitter profile, multiple invalid files will be generated if the input data contains multiple segments.
ds_report_content	Memory output		string	Specifies the content of the DocumentSplitter report.
ds_rpt_valid_results	Memory output		string	Contains a list of ds_rpt_valid_content. The ds_rpt_valid_content output specifies the content of the valid EDI data.  When splitting the EDI based on the split point grouping specified in the DocumentSplitter profile, multiple valid files will be generated if the input data contains multiple segments

Output Field	Mode Selected	Output Item	Datatype	Description
ds_rpt_invalid_results	Memory output		string	Contains a list of ds_rpt_invalid_content. The ds_rpt_invalid_content output specifies the content of the invalid EDI data.  When splitting the EDI based on the split point grouping specified in the DocumentSplitter profile, multiple invalid files will be generated if the input data contains multiple segments
dataSwapper				
report	N/A		string	An audit report showing what data was swapped.
output	N/A		string	A changed EDI file containing the swapped data.

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Error Schema Element	Datatype	Description
msg	string	Error message description.
msgCode	string	The error code. It represents TIBCO ActiveMatrix BusinessWorks Plug-in for EDI and TIBCO Foresight errors. See Error Codes for details.

Instream CallBack

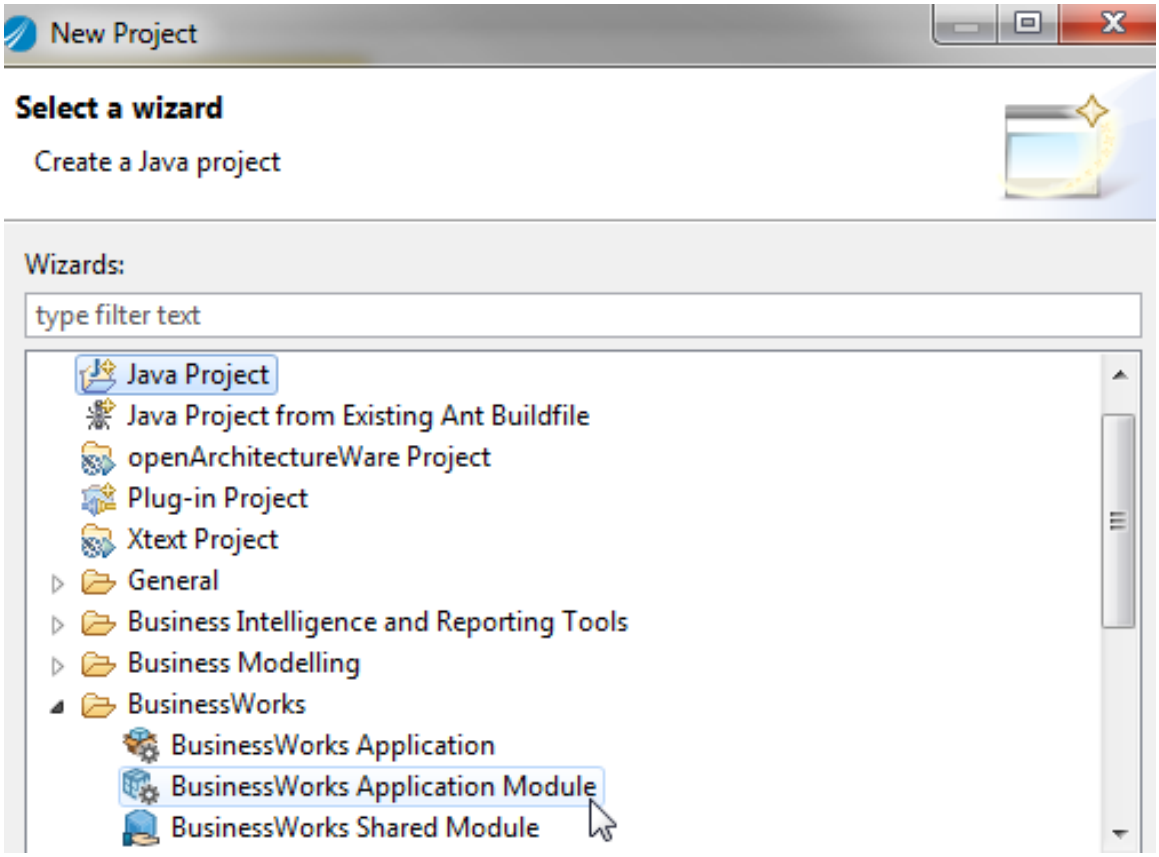
Overview

Using a Java Class for Instream Callback allows you to select validation guidelines and profiles based on the contents of the input data by modifying the Java code.

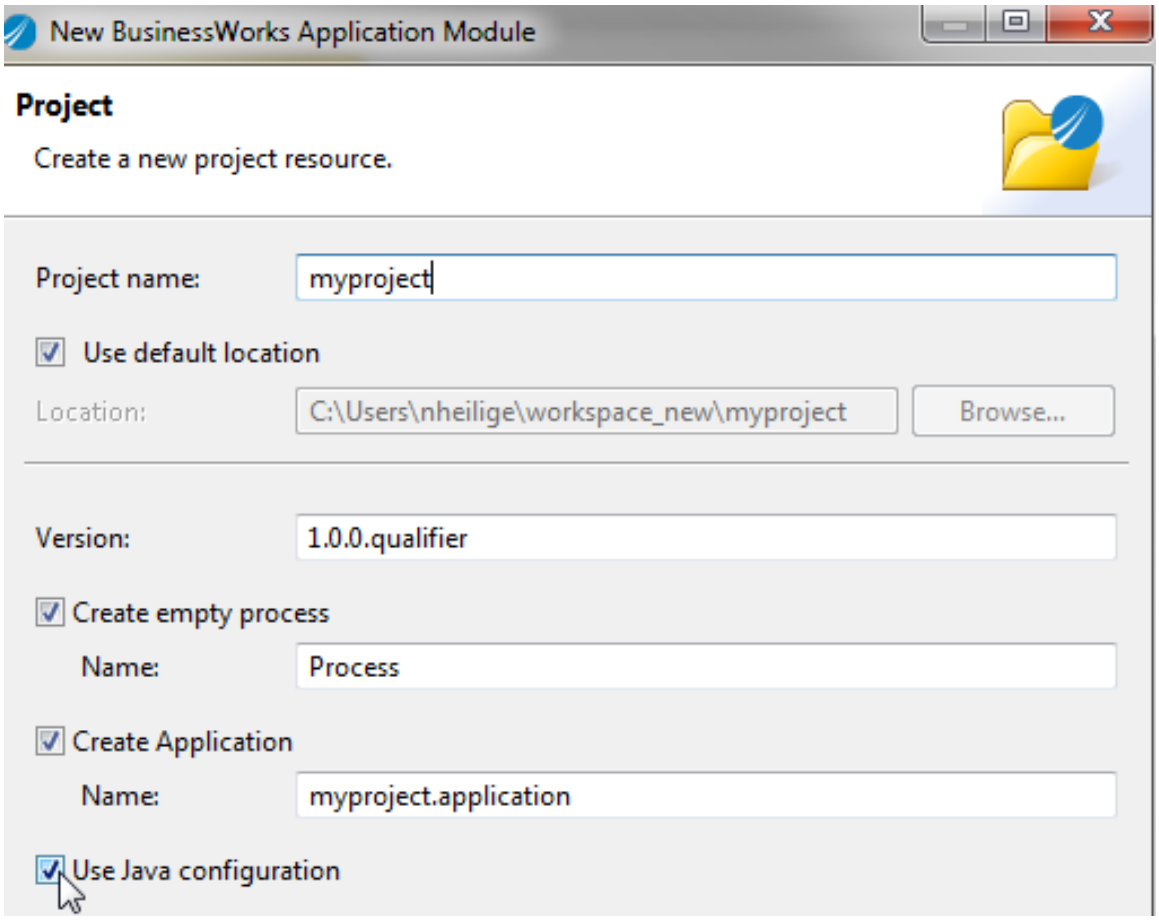
Enable Callback

To enable an Instream CallBack:

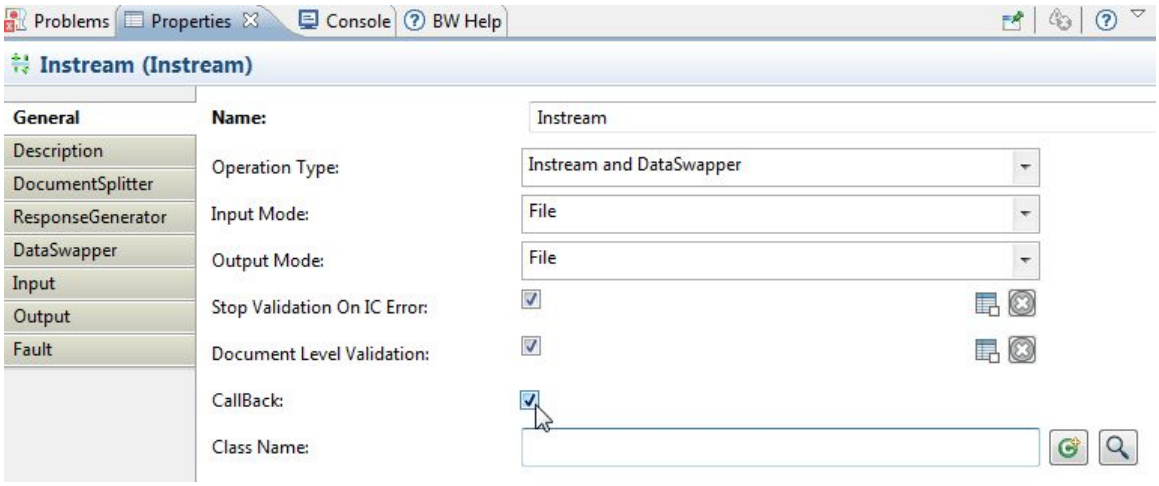
1. Select the TIBCO ActiveMatrix BusinessWorks™ Application Module when creating a new project.



2. Enter a Project name, select the **Use Java configuration** checkbox, and click **Next**.

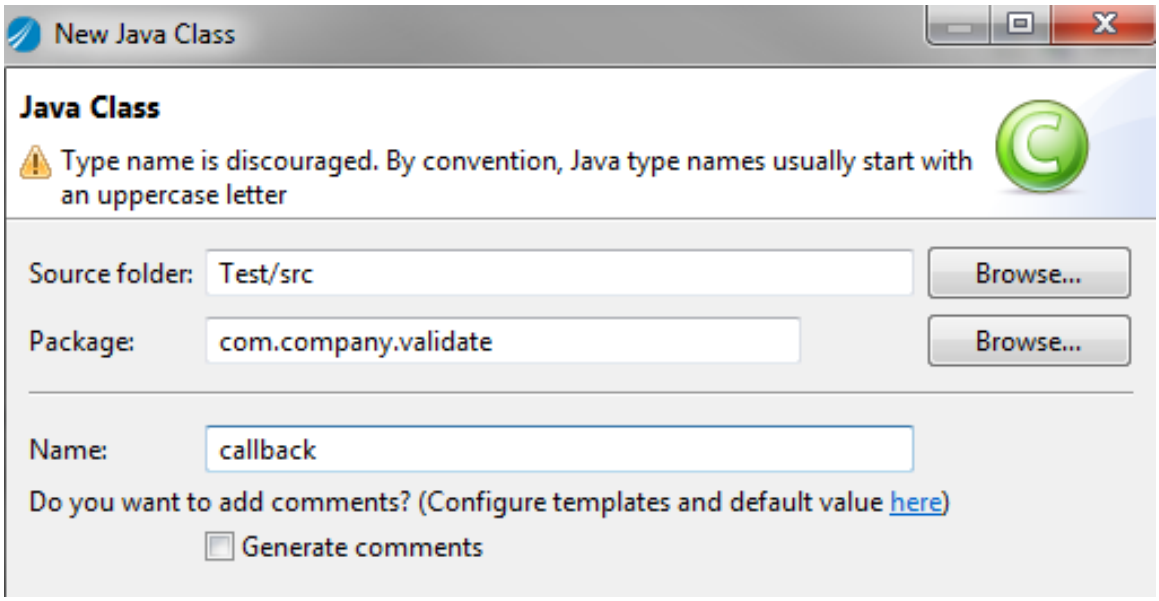


3. Select the **CallBack** checkbox in the Instream activity General tab and click the **Create a new class** icon  .



4. Fill in Class Name and click **Finish**.

   A Java class is created with default functions and variables.



Result

MSH information will set to `HashMap<String, String> infoMap`.

Use the "get" function to access the values in the map.

```
infoMap.get("Info.MessageTypeMessageCode")
```

Setting Guidelines and Profiles

You can set guidelines and profiles by modifying `public void selectGuideline()` in the Java code.

Refer to the associated examples:

- EDI Plug-in Example Processes

If Guideline Resides on Disk (Instream Database Directory)

Sign the map image to variable `mapFileBuffer`.

```
guidelineName = "VXR_V03.std";
```

If Guideline Resides in Memory (Database)

1. Sign the guideline image to variable `guidelineByteBuffer`.
   ```
   guidelineByteBuffer=(guideline in byte[]);
   ```
2. Sign the guideline name to variable `guidelineName`.
   ```
   guidelineName = "VXR_V03.std";
   ```

Setting a Profile

Directly assign the profile name to the variable `apfFileName`.

```
apfFilename="user_profile.apf";
```

ISErrorRefiner

Use the ISErrorRefiner activity to modify error lines (also called DTL lines) in an Instream results file.

The ISErrorRefiner activity is for use by Centers for Medicare and Medicaid Services (CMS) customers utilizing data that has been validated by Instream to convert EDI to and from flat file format for compliance with the CEM requirements for electronic processing of claims. Purchase of the TIBCO Foresight® Instream® MAC Adapter is required to use ISErrorRefiner

Errors in Instream are assigned a severity, a HIPAA Type, various response codes, and more that apply to the error wherever it occurs. However, a customer may prefer that an error have different values for those parameters, depending on the location of the error within the EDI. ISErrorRefiner allows you to make changes to these parameters based on a configuration table of modification instructions.

For more details on ISErrorRefiner, refer to the *Foresight® Instream® MAC Adapter ISErrorRefiner* PDF document.

Example

An example use of ISErrorRefiner is as follows.

The following data is encountered during validation:

```
DTL 30 2330AN4 1343 27 4 1 1 7 41652 2 14482 8 7024 X7 21
```

The ISErrorRefiner configuration file specified in the input field "config_file" contains the following modification instructions:

If an Instream DTL record is found for Error 41652 in Loop 2330A on the fourth element of the N4 segment, then the following fields in the DTL record are changed:

- the element position set to '2'
- the severity set to '3'

The modified Instream detail file (saved as specified in the input field "output_file") contains the changed severity and element position rather than the default settings.

The modified data appears as follows:

```
DTL 30 2330AN4 1343 27 2 1 1 7 41652 3 14482 8 7024 X7 21
```

General

The **General** tab contains the following fields.

Field	Literal Value/ Module Property/ Process Property?	Description
Name	No	User-defined name for the activity.
Input Mode	No	Specifies where the input data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.
Output Mode	No	Specifies where the output data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.

Description

The **Description** tab is used to provide a short description for the activity.

Input

The **Input** tab contains the following fields.

Field	Datatype	Description
instream_result	string	Specifies the file path and name of the Instream detail file to be modified. (Required)  This file is not changed by ISErrorRefiner.
config_file	string	Specifies the file path and name of the ISErrorRefiner configuration file containing the modification instructions that ISErrorRefiner is to use. (Required) (See ISErrorRefiner Configuration File for more information about the ISErrorRefiner Configuration Files.)
output_file	string	Specifies the file path and name for the output modified Instream detail file. (Required) It should **not** be the same as the Input file.

Output

The **Output** tab contains the following fields.

Field	Datatype	Description
instream_result_out	string	File path and name for the output modified Instream detail file.
return_code	string	Specifies the ISErrorRefiner return code. For example, 0 means ISErrorRefiner ran successfully, 2 means a specified file could not be found, and so on. Return codes are listed in Chapter 4 of the *Foresight® Instream® MAC Adapter ISErrorRefiner* PDF document

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Error Schema Element	Datatype	Description
msg	string	Error message description.
msgCode	string	The error code. It represents TIBCO ActiveMatrix BusinessWorks Plug-in for EDI and TIBCO Foresight errors. See Error Codes for details.

ISErrorRefiner Configuration File

The ISErrorRefiner Configuration File is used to specify the changes that ISErrorRefiner is to make.

For complete information on ISErrorRefiner Configuration files, see the *ISErrorRefiner* PDF document.

ISErrorRefiner Configuration File Format

The ISErrorRefiner Configuration File is a text file containing two sections, which can occur in any order.

[Settings] Section

The Settings Section starts with the line [Settings], and contains the optional StopErrors entry.

StopErrors is used to identify errors that, when encountered, cause the severity of any and all other errors at the same level (i.e., loop/segment/element) to be set to zero (ignore); in effect, causing just the Stop Error to appear.

This results in fewer extraneous errors such as those generated by unused loops or segments, where the contents of those loops and segments contain additional errors that are unimportant when compared to the main error.

The StopErrors entry is a comma-separated list of one or more five-digit error numbers.

StopErrors=*Error#*{,*Error#*,*Error#*,...}

Example

```
StopErrors=11003,11302,11323,11324,10808,10811,10812,95171,95111,95236,95237,95238,9
5120,95121,95239,95240,95241,95242,95235,95211,95233
```

[Extended Warning Levels] Section

The Extended Warning Levels Section contains instructions about the modifications ISErrorRefiner is to make. Each line in this section corresponds to one Error/Location, and is structured as follows:

Error+Location=New Data

When ISErrorRefiner encounters an error (DTL) line in the input Detail File, it consults the table for an entry that has the same Error Number and Location (Loop, Segment, Element). If it finds a match, it replaces the fields in the DTL line with the data from the New Data portion of the instructions.

- Error+Location

  Error+Location is comprised of four fields separated by the vertical bar character '|':

  - Field 1 = Error Number
  - Field 2 = Loop ID
  - Field 3 = Segment ID
  - Field 4 = Element Number (followed by a dash and sub-element number for Composite sub-elements)
- New Data

  New Data is comprised of 12 fields separated by the comma character, and represents the data that is to replace various fields in the Input DTL record. Each field is optional, and should contain data only when that field is to be changed.

New Data Replacement Fields

Field #	Description	Valid Values
1	Severity	Single digit 0 through 6
2	HIPAA Type	Single digit 0 through 8
3	997 AK3-04 Code	One or two character string
4	997 AK4-03 Code	One or two character string
5	824 TED-01 Code	One, two, or three character string
6	824 TED-02 Code	One, two, or three character string
7	277 STC-01.02 Code	One through five character string
8	999 IK3-04 Code	One, two, or three character string
9	999 IK4-03 Code	One, two, or three character string
10	Application Data	String no longer than 20 characters
11	(Not Used)	
12	Additional Replacement Values	String of additional replacement fields separated by periods. See the Table *Additional New Data Replacement Fields*.

Additional New Data Replacement Fields

Field #	Description	Valid Values
1	(Not Used)	
2	Loop ID	1 - 6 characters
3	Segment ID	1 - 4 characters
4	Element ID	1 - 4 characters
5	Composite ID	1 - 4 characters

Field #	Description	Valid Values
6	Segment Position	1 - 10 digits
7	Element Position	1 - 6 digits or "none"
8	Sub-Element Position	1 - 2 digits or "none"
9 - 15	(Not Used)	
16	Segment Ordinal Number	1 - 5 digits

Example

This example shows an ISErrorRefiner Configuration File entry.

```
41652|2330A|N4|4=3,,,,,,,,,,,,,,,.....2.................
```

If a DTL record is found for Error 41652 in Loop 2330A on the fourth element of the N4 segment, then the following fields in the DTL record are changed:

- Severity set to '3'
- Element Position set to '2'.

ResponseGenerator

Use the ResponseGenerator activity to generate a response to EDI validated by the Instream activity.

Response documents (also called acknowledgments) are used to give feedback to the sender of a transaction on the status of the acceptance of the transaction by its recipient. See Types of Response Documents for details.

Additionally, custom reports can be generated for internal use. See Response Generator Custom Reports for details.

General

The **General** tab contains the following fields.

Field	Literal Value/ Module Property/ Process Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
Input Mode	No	Specifies where the input data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.

Field	Literal Value/ Module Property/ Process Property?	Description
Output Mode	No	Specifies where the output data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.
Generate 277	Yes	Check this checkbox to generate a 277 CA/U Acknowledgment. A 277 CA/U Acknowledgment is generated when responding to HIPAA 837 transactions. See Types of Response Documents for details.
Generate 824	Yes	Check this checkbox to generate an 824 Application Advice. An 824 Application Advice is generated when responding to X12 EDI transactions. See Types of Response Documents for details.
Generate 997	Yes	Check this checkbox to generate a 997 Functional Acknowledgment. A 997 Functional Acknowledgment is generated when responding to X12 EDI transactions. See Types of Response Documents for details.
Generate 999	Yes	Check this checkbox to generate a 999 Functional Acknowledgment. A 999 Functional Acknowledgment is generated when responding to X12 EDI transactions. See Types of Response Documents for details.
Generate TA1	Yes	Check this checkbox to generate a TA1 Interchange Acknowledgment. A TA1 Interchange Acknowledgment is generated when responding to X12 EDI transactions. See Types of Response Documents for details.
Generate Control	Yes	Check this checkbox to generate a CONTRL document. A CONTRL document is generated when responding to EDIFACT validation (except Inbound CONTRL). See Types of Response Documents for details.
Generate Custom Report	Yes	Check this checkbox to generate a custom report. See Response Generator Custom Reports for details.
Custom Report Template	Yes	Full path and name to a Custom Report Template. (Active only if Generate Custom Report is selected.) Refer to *Response Generator Technical Manual* PDF document for information about Custom Report Templates.

Field	Literal Value/ Module Property/ Process Property?	Description
Parameters	Yes	You can specify parameters listed in the *Response Generator Technical Manual* PDF document in this field. For example: • `-ge-y` indicates that group enveloping is to be included in the response documents. The Response Generator function can overwrite files if they already exist. • `-dav string` indicates that the string, for example, 005010 , is set as the application version for the outbound GS. • `-fver_pass` indicates that the ISA and GS version numbers for the source document will not be changed.

Description

The **Description** tab is used to provide a short description for the activity.

Input

The **Input** tab contains the following fields.

Field	Datatype	Description
instream_result	string	Specifies the file path and name of the Instream detail file for which the response will be generated. (Required)  This file is not changed by ResponseGenerator.
output_directory	string	Specifies directory path for the output files. (Required)  This field only appears when File mode is selected as the output mode in the Configuration panel. You need to specify the output directory for the output files.
profile	string	(Optional) Specifies which APF file to use. The APF file contains configuration information for the Instream activity. Each time you validate, an APF file is read to determine what is to be checked and what is written to output files. The default APF is $fsdeflt.apf and is located in the *TIBCO_FORESIGHT_HOME* / *Instream-Translator* /bin directory.  This field will not be displayed when the CallBack shared resource is enabled, because the APF file will be extracted from the CallBack code.

Field	Datatype	Description
tpa_file	string	Path to a Trading Partner Automation CSV file that contains a pointer to a setup file. All sections in the setup file are processed. See the *Trading Partner Automation* PDF document. If there is a conflict between the command line and the settings in the setup file selected by TPA, the last setting processed prevails.
parameters	string	Specifies the response generator parameters to be used. (Optional)

Output

The **Output** tab contains the following fields.

Field	Datatype	Description
return_code	string	Specifies the ResponseGenerator return code. For example, 100 means ResponseGenerator ran successfully, 110 means ResponseGenerator failed, and so on. Return codes are listed in Appendix A of *Response Generator Technical Manual* PDF document.
rg_277<_filename>	string	Provides the content (memory output) or specifies the full path and name (file output) of the 227 CA/U Acknowledgment.
rg_824<_filename>	string	Provides the content (memory output) or specifies the full path and name (file output) of the 824 Application Advice.
rg_997<_filename>	string	Provides the content (memory output) or specifies the full path and name (file output) of the 997 Functional Acknowledgment.
rg_999<_filename>	string	Provides the content (memory output) or specifies the full path and name (file output) of the 999 Implementation Acknowledgement.
rg_TA1<_filename>	string	Provides the content (memory output) or specifies the full path and name (file output) of the TA1 Interchange Acknowledgment.
rg_CONTRL_DOC<_filename>	string	Provides the content (memory output) or specifies the full path and name (file output) of the CONTRL document.
rg_custom_report<_filename>	string	Provides the content (memory output) or specifies the full path and name (file output) of the custom report document.

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Error Schema Element	Datatype	Description
msg	string	Error message description.
msgCode	string	The error code. It represents TIBCO ActiveMatrix BusinessWorks Plug-in for EDI and TIBCO Foresight errors. See Error Codes for details.

Shuffler

Use the Shuffler activity to enhance detail results from Instream by "shuffling" STC records into 837 and 276 flat file documents to identify errors found in front-end validation.

The Shuffler activity is for use by Centers for Medicare and Medicaid Services (CMS) customers utilizing data that has been validated by Instream to convert EDI to and from flat file format for compliance with the CMS-supplied Claims Edit Module (CEM) requirements for electronic processing of claims. Purchase of the TIBCO Foresight® Instream® MAC Adapter is required to use Shuffler.

Shuffler can be used with the following types of input and output:

- EDI input and EDI output - utilizing EDI detail file output from Instream to generate EDI data
- Flat file input and flat file output - utilizing flat file input from Translator to generate flat file data.

Shuffler should be placed **after** validation has occurred and **before** translation.

For more details on Shuffler, refer to the *TIBCO Foresight® Instream® MAC Adapter Introduction* PDF document.

Examples

Example uses of Shuffler are as follows.

EDI input and EDI output

1. Instream validates good EDI and creates a Detail file (DTL).
2. Shuffler inserts additional data into the EDI and Detail file received from Instream.
   - For each ISA, a CTRD record is added (optional).
   - After checking data against a user-specified STC Definition Table file, STC records are added after the segment in error for each error found in the original EDI file.
3. Shuffler outputs data in EDI format.

Flat file input and flat file output

1. Instream validates good EDI and creates a Detail file (DTL).
2. Translator converts the EDI data received from Instream into flat file format.
3. Shuffler inserts additional data into the flat file received from Translator:
   - For each ISA flat file, a CTRD record is added (optional).

- After checking data against a user-specified STC Definition Table file, STC records are added after the flat file segment in error for each error found in the original EDI file.

4. Shuffler outputs data in flat file format.

General

The **General** tab contains the following fields.

Field	Literal Value/ Module Property/ Process Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
Input Mode	No	Specifies where the input data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.
Output Mode	No	Specifies where the output data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.
Document Type	No	Select TEXT (if using flat-file data) or EDI (if using an Instream detail file).
Create CTR Record	No	Check this check box to create CTRD records.

Description

The **Description** tab is used to provide a short description for the activity.

Input

The **Input** tab contains the following fields.

Field	Datatype	Description
input	string	Input file path and name.
instream_result	string	Specifies the file path and name of the Instream detail file to be modified. (Required)  This file is not changed by Shuffler.
STC_definition_table	string	Specifies the path to a comma-separated STC definition table describing whether or not an STC record should be included in the resulting file. Refer to STC Definition Table for more information.

Field	Datatype	Description
map_file	string	This field specifies the name and extension of a translation map file. TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI supports map files with the .map and .xml file name extensions. A map file specifies the source and target guidelines for each translation map. The source guideline describes the data before translation. The target guideline describes the data after translation. See Translator for more information about map files
config_file	string	Specifies the file path and name of the Shuffler configuration file. (Required)
outputfile_pathname	string	Specifies the file path and name for the output modified file. (Required) It should **not** be the same as the Input file.

Output

The **Output** tab contains the following fields.

Field	Datatype	Description
output	string	File path and name for the output modified Instream detail file.
error_message	string	Error message description.
return_code	string	Specifies the Shuffler return code. For example, 100 means the activity ran successfully, 0 means there was a parameter error, and so on. Return codes are listed in Appendix A of *TIBCO Foresight® Instream® MAC Adapter Introduction.*

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Error Schema Element	Datatype	Description
msg	string	Error message description.
msgCode	string	The error code. It represents TIBCO ActiveMatrix BusinessWorks Plug-in for EDI and TIBCO Foresight errors. See Error Codes for details.

STC Definition Table

By default, STC segments appear in validation results in a certain way. To utilize different and/or multiple STC codes when you are using Shuffler, you can create an STC Definition table to use as an override file.

For complete information on STC Override files, see Appendix G: 277CA STC Override in the *Response Generator Technical Manual* PDF document.

STC Definition Table File Format

The STC Definition Table is a comma delimited file containing error numbers and information for Shuffler to use when creating STC segments.

When omitting values in a line, include the commas to maintain correct positioning.

Each line contains these, in order:

```
Error#,LoopID,Sement,Element,CategoryCode,StatusCode,EntityIdentifierCode,InsertWith
Shuffler
```

Value	Description
Error#	Error number that should have new STC behavior.
LoopID	Apply the STC override to a certain loop only.
Segment	Apply the STC override to a certain segment only. If omitted, the override applies to all loops.
Element	Apply the STC override to a certain element only.
CategoryCode	Value to be used for the STC01.01.
StatusCode	Value to be used for the STC01.02.
EntityIdentifierCode	Value to be used for the STC01.03.
InsertWithShuffler	Indicates if STC information should be inserted into the data by Shuffler. If left blank, the default (yes) is used. 0 – No 1 – Yes (default)

Example STC Definition Table File

This example shows an STC Definition Table File.

```
[Version]
ValidatorVersion=7.8.0
```

Error#	LoopID	Sement	Element	CategoryCode	StatusCode	EntityIdentifierCode	InsertWithShuffler
32666	2000A	REF	2	5	249	1	1
32667	2000B	NM1	9	6	206	2	0
32669	2010AA	REF	1	8	222	3	1
40600	1000A	PER	2	A8	561		0

Translator

Use the Translator activity to translate a file from one format to another format.

See *TIBCO Foresight® Translator Using Translator* PDF document for more information about the translation function.

Examples

Refer to the associated examples: EDI Plug-in Example Processes.

Translated Data Formats

Translation for the following data formats is supported by this activity.

From (Source)	To (Target)
EDI	XML
XML	EDI
EDI	Flat File
Flat File	EDI
Flat File	XML
XML	Flat File
EDI	EDI
Flat File	Flat File

General

The **General** tab contains the following fields.

Field	Literal Value/ Module Property/ Process Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
Operation Type	No	This drop-down list allows you to specify the operation type. Refer to the Translated Data Formats section above.
Input Mode	No	Specifies where the input data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.
Output Mode	No	Specifies where the output data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.
CallBack Buffer Size		Select a number.
CallBack	No	Select this checkbox to use a Java Class for callback (optional).  This function is not available for XML input. See to Translator CallBack for more complete information on Enabling Callbacks.
Class Name	No	Specify a Java Class. This field is enabled when CallBack is checked.  Create a new Java Class.  Browse to an existing Java Class using the Type selection dialog. See to Translator CallBack for more complete information on Enabling Callbacks.

Description

The **Description** tab is used to provide a short description for the activity.

Input

The **Input** tab contains the following fields.

Field	Datatype	Description
input	string	Specifies the full path and name of the input file.

Field	Datatype	Description
map_filename	string	This field specifies the name and extension of a map file. The EDI Plug-in supports map files with the .map and .xml file name extensions. A map file specifies the source and target guidelines for each translation map. The source guideline describes the data before translation. The target guideline describes the data after translation.  All involved XSD, STD, and MAP files must be in the `TIBCO_FORESIGHT_HOME /Translator/Database` directory. If you want to share guidelines with the `TIBCO_FORESIGHT_HOME /Instream/Database` directory, refer to the Share Guidelines and Maps (Translator) section below.
output_directory	string	Specifies the full path and name of the directory for the output files.
input encoding	string	(Optional) The input encoding used for translation. This is used when translating to ISO-8859-1 only. Valid values are blank (no encoding) or ISO-8859-1.
output encoding	string	(Optional) The output encoding used for translation. Valid values are blank (no encoding), UTF-8, UTF-16, or ISO-8859-1.

separater_group	string	Specifies that translator should overwrite the target EDI delimiters using this string of four characters (in this order):
		
		`seg/elm/comp/repeat`
		
		For example, entering
		
		`!*:^`
		
		will cause the target EDI to use the '`!`' character as the segment terminator, the '`*`' character as the element delimiter, the '`:`' character as the composite sub-element delimiter, and the '`^`' character as the repeating element delimiter.

Share Guidelines and Maps (Translator)

If you install Translator and Instream in different directories, you can share their Database directories by modifying the `TIBCO_FORESIGHT_HOME /Instream/bin/$dir.ini` or `TIBCO_FORESIGHT_HOME /Translator/bin/$dir.ini` file.

For example, if you want to use the `TIBCO_FORESIGHT_HOME /Instream/Database` directory as shared, perform the following steps:

1. Open the `TIBCO_FORESIGHT_HOME /Translator/bin/$dir.ini` file.
2. Modify the database path and point it to the `TIBCO_FORESIGHT_HOME /Instream/Database` directory.

   The following is an example:

```
:* FORESIGHT Supplied .STD Standard Files
FSFACTORY ="C:\Foresight\Instream\Static"
```

```
:*User-defined .STD Standard Files
ALLUSERSSHARED ="C:\Foresight\Instream\Database"
```

The value must be enclosed with double quotation marks.

3. After making the modification, you can put the involved XSD, STD, and MAP files either in the `TIBCO_FORESIGHT_HOME /Translator/Database` directory or in the `TIBCO_FORESIGHT_HOME / Instream/Database` directory.

Output

The **Output** tab contains the following fields.

Output Item	Datatype	Description
output_msg	string	The output of this activity is the translated message in the specified format, which can be the EDI, XML, or Flat File format. The generated message format is specified in the Operation Type field on the Configuration tab.
return_code	integer	Specifies the Translator return code. For example, 100 means the transaction ran successfully, 158 means the map file cannot be opened, and so on. Translator return codes are listed in the *TIBCO Foresight® Translator Using Translator* PDF document.
encoding	string	The encoding used for translation.

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Error Schema Element	Datatype	Description
msg	string	Error message description.
msgCode	string	The error code. It represents TIBCO ActiveMatrix BusinessWorks Plug-in for EDI and TIBCO Foresight errors. See Error Codes for details.

Translator CallBack

Overview

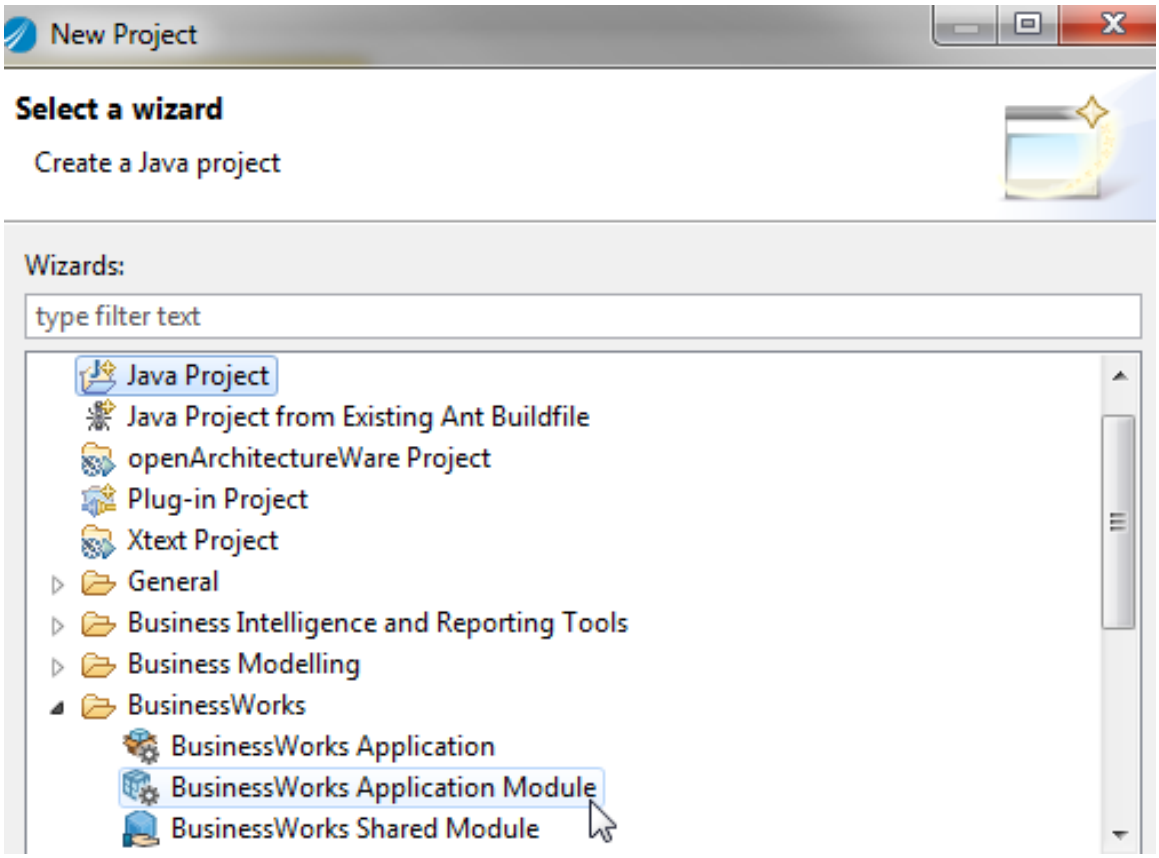
Using a Java Class for Translator Callback allows you to allows you to select a translation map based on the contents of the input data by modifying the Java code.

This function is not available for XML input.

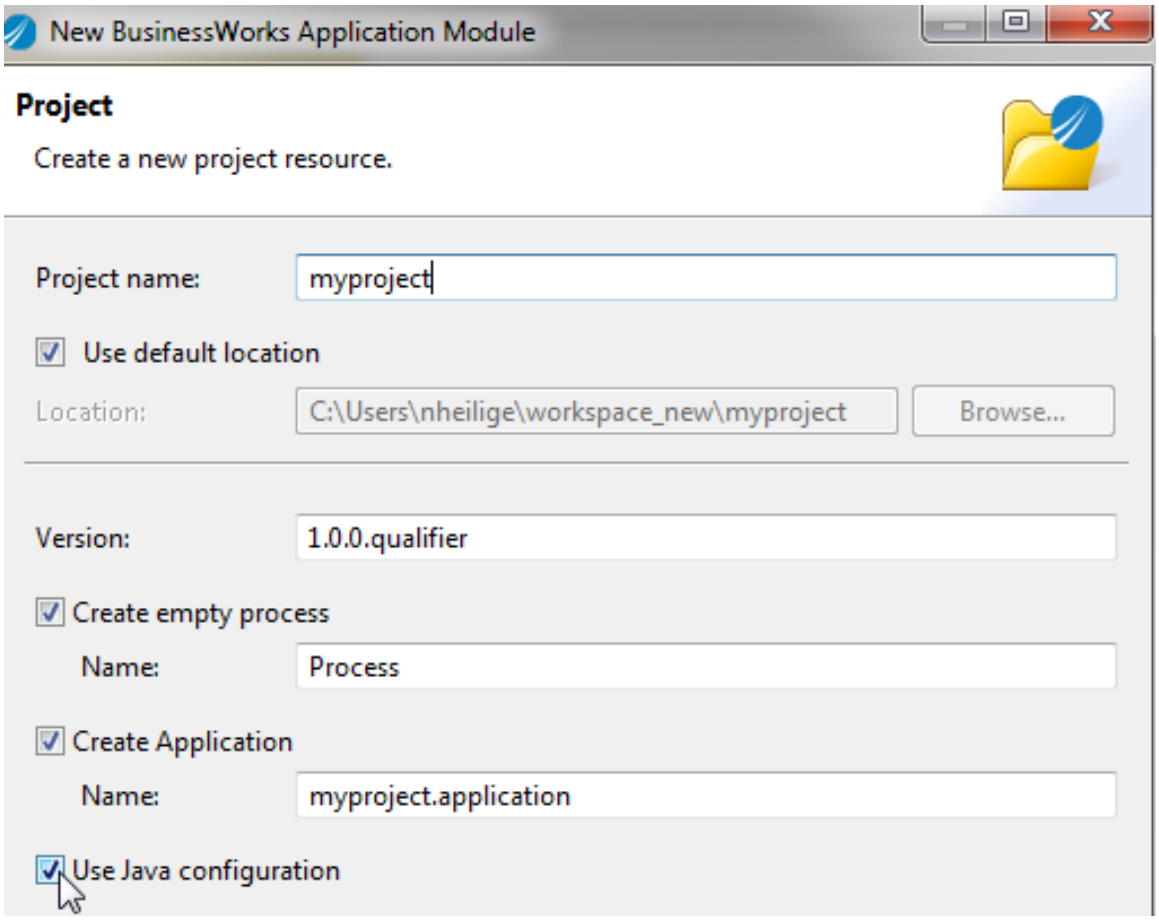
Enable Callback

To enable Translator CallBack:

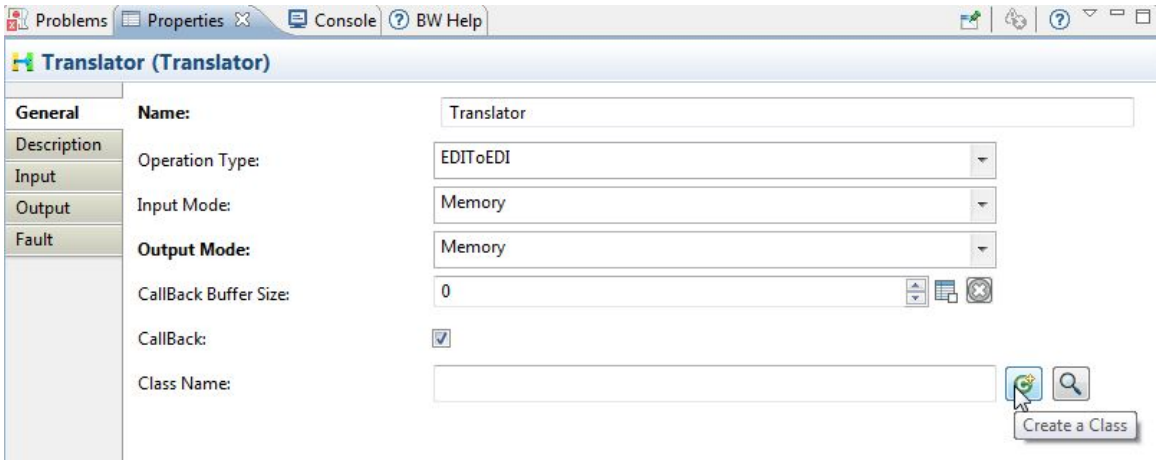
1. Select the TIBCO ActiveMatrix BusinessWorks™ Application Module when creating a new project.



2. Enter a project name, select the **Use Java** configuration checkbox, and click **Next**.

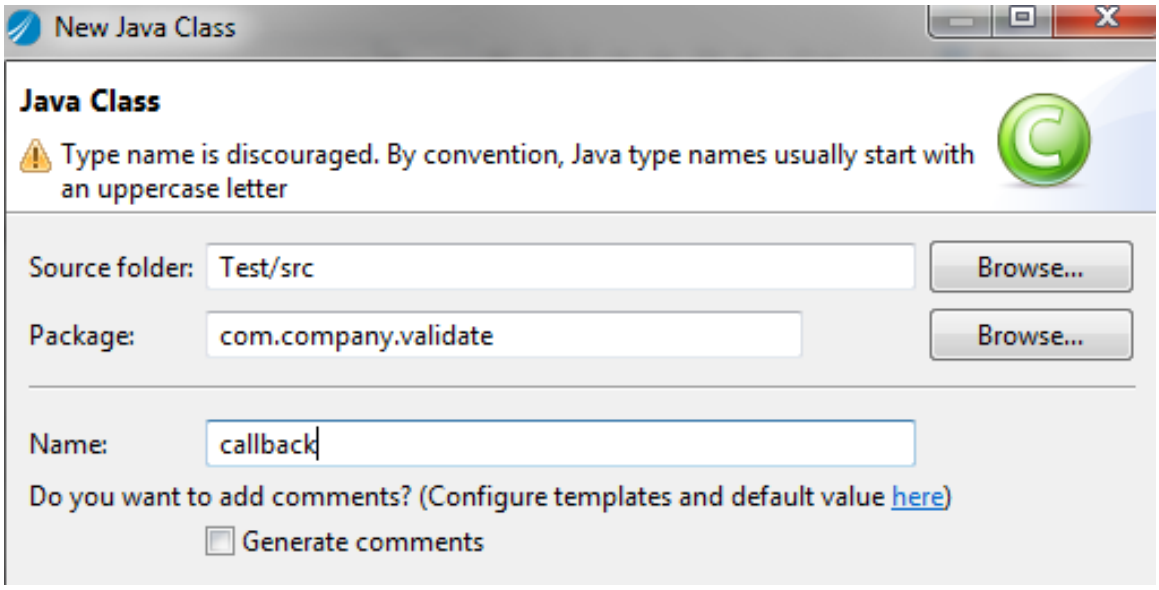


3. Select the **CallBack** checkbox in the Translator activity General tab and click the **Create a new class** icon  .



4. Enter a class name and click **Finish**.

   A Java class is created with default functions and variables.



Result

MSH information will be set to `HashMap<String, String> infoMap`

Use the "get" function to access the values in the map.

```
infoMap.get("Info.MessageTypeMessageCode")
```

Setting the Map, Source Guideline and Target Guideline

Refer to the associated examples:

- EDI Plug-in Example Processes

If Map Resides on Disk (Translator Database Directory)

Set up the Map file name.

1. Directly assign the map name to variable `mapFileName`.
   ```
   mapFileName = "VXR_V03_VXR_V03_EX.map";
   ```
2. Update the configName variable with the format `configName=mapname||`.
   ```
   configName="VXR_V03_VXR_V03_EX.map||"
   ```

If Map Resides in Memory (Database)

1. Update the configName variable with the format `configName = MapName|SourceGuidelineName|TargetGuidelineName;`
2. Assign the map image to variable `mapFileBuffer =(map image in byte[])`.
3. Specify the source guideline variable with the format `sourceGuidelineBuffer = (source guideline image in byte[]);`
4. Specify the target guideline variable with the format `targetGuidelineBuffer = (target guideline image in byte[]);`

X12Prescanner

Use the X12Prescanner activity to parse X12 document header (ISA/GS/ST) segments and put each element into a separate output field.

Example

Prescanning allows data to be scanned for information such as transaction type and trading partner group. This information can then be used to proceed with processing data in an appropriate manner. For example: If the X12 data contains a GS08 value of 005010X279A1 **AND** a GS01 value of HE, Instream should use the 271-X279 guideline for validation.

General

The **General** tab contains the following fields.

Field	Literal Value/ Module Property/ Process Property?	Description
Name	No	The name to be displayed as the label for the activity in the process.
Input Mode	No	Specifies where the input data is stored. There are two options: File and Memory. Memory is normally the most efficient way of connecting, but large documents should be stored as files.

Description

The **Description** tab is used to provide a short description for the activity.

Input

The **Input** tab contains the following fields.

Field	Datatype	Description
input	string	Specifies the file path and name of the input data file. (Required)
encoding	string	Specifies encoding type: UTF-8, UTF-16, UTF-16BE, UTF-16LE, ISO-8859-1.

Output

The **Output** tab contains the following fields.

Field	Sub-Field	Datatype
ISA	AuthorizationInformationQualifier	string
	AuthorizationInformation	string
	SecurityInformationQualifier	string
	SecurityInformation	string
	InterchangeIDQualifier	string
	InterchangeSenderID	string
	InterchangeIDQualifier	string
	InterchangeReceiverID	string
	InterchangeDate	string
	InterchangeTime	string
	RepetitionSeperator	string
	ControlVersionNumber	string
	ControlNumber	string
	AcknowledgmentRequested	string
	InterchangeUsageIndicator	string
	CompomentElementSeperator	string
GS	IdentifierCode	string
	ApplicationSenderCode	string
	ApplicationReceiverCode	string
	Date	string
	Time	string
	GroupControlNumber	string
	ResponsibleAgencyCode	string
	Version	string
ST	TransactionSetIdentfierCode	string

Field	Sub-Field	Datatype
	TransactionSetControlNumber	string
	ImplementationConventionReference	string
	HierarchicalStructureCode	string
	DescriptionTransactionSetPurposeCode	string
FileInfo	FileName	string
	FileSize	integer

Fault

The **Fault** tab lists exceptions that are thrown by this activity.

Error Schema Element	Datatype	Description
msg	string	Error message description.
msgCode	string	The error code. It represents TIBCO ActiveMatrix BusinessWorks Plug-in for EDI and TIBCO Foresight errors. See Error Codes for details.

Managing Logs

Logs are used to trace and troubleshoot exceptions. The plug-in allows users to set up log levels and export logs. Additionally, the plug-in also allows users to enable TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI logging.

- See Managing Plug-in Logs for more details about setting up log levels and exporting logs.
- See Enabling TIBCO EDI Plug-in Logging for more details about enabling TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI logging.

Managing Plug-in Logs

The plug-in allows users to set up log levels and export logs by modifying the `logback.xml` file.

TIBCO Business Studio™ provides a built-in debugger, which allows users to debug a project. When starting debugging or running a project, the logs are displayed in the Console view. You can set up the log level to define the log amount. To set up the console log level, see Setting Up Log Level.

The plug-in also allows users to export logs of the defined log levels to a file. To export logs to a file, see Exporting Logs.

Exporting Logs

You can set up the log level and export logs to a file by modifying the `logback.xml` file.

Procedure

1. Navigate to the *TIBCO_HOME*`\bw\6.3\config\design\logback` directory and open the `logback.xml` file.

   When deploying the application in TIBCO® Enterprise Administrator, you need to navigated to the *TIBCO_HOME*`\bw\domains\mydomain\appnodes\myspace\mynode` directory to find the `logback.xml` file.

2. Add the following node to specify the file location.

```
<appender name="FILE" class="ch.qos.logback.core.FileAppender">
   <file>c:/bw6-asbw.log</file>
      <encoder>
        <pattern>%d{HH:mm:ss.SSS} [%thread] %-5level %logger{36}-%msg%n</pattern>
      </encoder>
</appender>
```

   The `file` tag defines the location to which the log is exported and the value is the absolute path of the file that is detailed to the file name.

3. Add the following node to specify the log level.

```
<logger name="com.tibco.bw.palette.edi">
   <appender-ref ref="FILE"/>
   <level value="Error"/>
</logger>
```

   The `level` tag defines the log level and the value can be `Error` or `Debug`.

4. Save the file.

Setting Up Log Level

When you run a process in TIBCO Business Studio™, the runtime logs are displayed in the Console view. You can set up the log level before running a process.

Procedure

1. Navigate to the *TIBCO_HOME*\bw\6.3\config\design\logback directory and open the logback.xml file.

   When deploying the application in TIBCO® Enterprise Administrator, you need to navigate to the *TIBCO_HOME*\bw\domains\mydomain\appnodes\myspace\mynode directory to find the logback.xml file.

2. Add the following node to specify the log level.

```
<logger name="com.tibco.bw.palette.edi">
   <level value="Error"/>
</logger>
```

   The level tag defines the log level and the value can be Error or Debug.

3. Save the file.

Enabling EDI Plug-in Logging

The TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI allows users to enable TIBCO EDI Plug-in logging to troubleshoot errors that are thrown by TIBCO EDI Plug-in.

Before deploying the application in TIBCO® Enterprise Administrator, you need to add the Log Properties listed in the Properties Added in the INI File column to the config.ini file to enable TIBCO EDI Plug-in logging.

The config.ini file is located in the *TIBCO_HOME*\bw\domains\mydomain\appnodes\myappspace\myappnode directory.

Procedure

1. Start TIBCO Business Studio™.
2. Click **Run** > **Run Configurations**.
3. Click **BusinessWorks Application** > **BWApplication** in the left panel.
4. Click the **(x)=Arguments** tab in the right panel.
5. Type the following properties in the **VM arguments** panel. Click **Apply**.

```
-Dcom.tibco.plugin.as.filelog.enable=True
-Dcom.tibco.plugin.as.filelog.level=FINEST
-Dcom.tibco.plugin.as.filelog.directory=d:/logs/Loging_0107
-Dcom.tibco.plugin.as.filelog.filename=FINEST.log
```

   See Log Properties for more details about the value of each property.

Log Properties

The log properties allow users to set up log levels, specify the log file name, and the location of the log file.

The properties in the following columns are used when:

- Properties Added in VM Arguments

  you want to enable TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI logging in TIBCO Business Studio™ before running a project.

- Properties Added in the INI File

  you want to enable TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI logging in TIBCO Enterprise Administrator® before deploying an application.

See Enabling TIBCO EDI Plug-in Logging for more details.

Properties Added in VM Arguments	Properties Added in the INI File	Datatype	Description
`-Dcom.tibco.plugin.as.filelog.enable`	`com.tibco.plugin.as.filelog.enable`	boolean	Set the value to `true` to enable TIBCO EDI Plug-in logging.
`-Dcom.tibco.plugin.as.filelog.level`	`com.tibco.plugin.as.filelog.level`	string	The amount of logging produced by TIBCO EDI Plug-in core library can be adjusted and retrieved using this property. The following values can be used: • `None`: Do not return any information. • `FATAL`: Return only fatal errors. • `ERROR`: Return errors. • `WARN`: Return warnings. • `INFO`: Return debug information. • `FINE`: Return fine debug information. • `FINER`: Return more detailed debug information. • `FINEST`: Return the most detailed debug information. The default value is `ERROR`.
`-Dcom.tibco.plugin.as.filelog.directory`	`com.tibco.plugin.as.filelog.directory`	string	The output destination of the log file.
`-Dcom.tibco.plugin.as.filelog.filename`	`com.tibco.plugin.as.filelog.filename`	string	The file name.

Error Codes

The exceptions that are thrown by the TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI are listed with the corresponding descriptions and resolutions.

Error Code and Message	Role	Category	Description	Resolution
TIBCO-BW-PALETTE-EDI-501000 `The map is required.`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	A translation map is required.	Specify a map to be used for translation.
TIBCO-BW-PALETTE-EDI-501001 `The input data is required.`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	Input data for translation is required.	Specify input data to be used for translation.
TIBCO-BW-PALETTE-EDI-501002 `The operation type is required.`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	An operation type is required.	Specify an operation type for translation (e.g., EDItoXML, XMLtoFLAT, etc).
TIBCO-BW-PALETTE-EDI-501004 `TIBCO-BW-PALETTE-EDI-501004: return code [<return code>], error message : <error message>`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	This error points to a TIBCO Foresight® Translator return code. For example: `TIBCO-BW-PALETTE-EDI-501004:return code [158], error message : Error 44 : Access map failed`	Using the return code provided in the message, refer to the Return Codes section of the *TIBCO Foresight® Translator Using Translator* PDF document.
TIBCO-BW-PALETTE-EDI-501005 `BYMEM_MISSINGSTD={0}`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	The selected standard is missing.	Ensure the standard is in the correct location.
TIBCO-BW-PALETTE-EDI-501006 `METHORD_ERROR={0} {1}`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	An exception occurred during translation.	This is an internal product error. Contact TIBCO Support.

Error Code and Message	Role	Category	Description	Resolution
TIBCO-BW-PALETTE-EDI-501007 `Callback is checked, but no class selected.`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	The Java Callback checkbox is selected, but no class was specified.	Specify a Java Callback class name.
TIBCO-BW-PALETTE-EDI-501008 `The output directory {0} does not exist.`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	Output directory does not exist in the specified location.	Ensure output directory appears in the specified location.
TIBCO-BW-PALETTE-EDI-501009 `The output directory is required.`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	An output directory is required.	Specify a valid output directory.
TIBCO-BW-PALETTE-EDI-501009 `The input file {0} does not exist.`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	Input file does not exist in the specified location.	Ensure input file appears in the specified location.
TIBCO-BW-PALETTE-EDI-501010 `CallBack selected, but there is no class defined.`	errorRole	EDI-Plug-in ERROR_TRANSLATOR	The Java Callback checkbox is selected, but no class is defined.	Define a Java Callback class name.
TIBCO-BW-PALETTE-EDI-502000 `MESSAGE= return code [<return code>], error message : <error message>`	errorRole	EDI-Plug-in ERROR_RESPGEN	This error points to a ResponseGenerator return code. For example: `TIBCO-BW-PALETTE-EDI-501004:return code [180], error message : Failed to initialize Response Generator.`	Using the return code provided in the message, refer to the Return Codes section of the *Response Generator Technical Manual* PDF document.

Error Code and Message	Role	Category	Description	Resolution
TIBCO-BW-PALETTE-EDI-502001 `At least one acknowledgment must be selected.`	errorRole	EDI-Plug-in ERROR_RESPGEN	No acknowledgment types are selected.	Select at least one acknowledgment type.
TIBCO-BW-PALETTE-EDI-502002 `Output directory parameter must be supplied.`	errorRole	EDI-Plug-in ERROR_RESPGEN	An output directory is required.	Specify a valid output directory.
TIBCO-BW-PALETTE-EDI-502003 `Invalid output directory {0}.`	errorRole	EDI-Plug-in ERROR_RESPGEN	The specified output directory is invalid.	Specify a valid output directory.
TIBCO-BW-PALETTE-EDI-502004 `Could not configure 997 and 999 at the same time.`	errorRole	EDI-Plug-in ERROR_RESPGEN	A 997 acknowledgment and a 999 acknowledgment cannot be generated simultaneously.	Generate these response documents separately.
TIBCO-BW-PALETTE-EDI-502005 `Could not configure Control and X12 ACK at the same time.`	errorRole	EDI-Plug-in ERROR_RESPGEN	A CTRL response and an X12 acknowledgment cannot be generated simultaneously.	Generate these response documents separately.
TIBCO-BW-PALETTE-EDI-503000 `MESSAGE= return code [<return code>], error message : <error message>`	errorRole	EDI-Plug-in ERROR_INSTREAM	This error points to an Instream® return code. For example: `TIBCO-BW-PALETTE-EDI-501004:return code [131], error message : $dir.ini or fsdir.ini cannot be found.`	Using the return code provided in the message, refer to the Return Codes section of the *Instream Validation Technical Manual* PDF document.

Error Code and Message	Role	Category	Description	Resolution
TIBCO-BW-PALETTE-EDI-503001 `Instream callback missing guideline name.`	errorRole	EDI-Plug-in ERROR_INSTREAM	The Callback checkbox is selected, but no guideline was specified.	Specify a guideline name.
TIBCO-BW-PALETTE-EDI-503002 `Access input file failed, please check operation mode.`	errorRole	EDI-Plug-in ERROR_INSTREAM	There is a mis-match between the operation mode selected (file or memory) and the input file.	Select the correct Operation Type.
TIBCO-BW-PALETTE-EDI-503003 `Guideline is required.`	errorRole	EDI-Plug-in ERROR_INSTREAM	A validation guideline is required.	Specify a guideline to be used for validation.
TIBCO-BW-PALETTE-EDI-503004 `999997=Could not configure 997 and 999 at the same time.`	errorRole	EDI-Plug-in ERROR_INSTREAM	A 997 acknowledgment and a 999 acknowledgment cannot be configured simultaneously.	Configure these response documents separately.
TIBCO-BW-PALETTE-EDI-503005 `Could not configure Control and X12 ACK at the same time.`	errorRole	EDI-Plug-in ERROR_INSTREAM	A CTRL response and an X12 acknowledgment cannot be configured simultaneously.	Configure these response documents separately.
TIBCO-BW-PALETTE-EDI-503006 `For document level validation, doconly_parameters parameter must be supplied.`	errorRole	EDI-Plug-in ERROR_INSTREAM	The Document Level Validation checkbox is selected on the General tab but no Separators were specified.	Uncheck Document Level Validation checkbox OR Specify separators. Refer to document_only_parameters field in Instream.
TIBCO-BW-PALETTE-EDI-503007 `Invalid output directory [{0}].`	errorRole	EDI-Plug-in ERROR_INSTREAM	A valid output directory is required.	Specify a valid output directory.

Error Code and Message	Role	Category	Description	Resolution
TIBCO-BW-PALETTE-EDI-506000 `Unsupported encoding [{0}].`	errorRole	EDI-Plug-in ERROR_X12PRESCANNER	The specified encoding option is not supported.	Ensure a supported encoding type is selected: UTF-8, UTF-16, UTF-16BE, UTF-16LE, ISO-8859-1.
TIBCO-BW-PALETTE-EDI-506001 `Invalid EDI document.`	errorRole	EDI-Plug-in ERROR_X12PRESCANNER	The specified EDI document is invalid.	Supply a valid EDI document for prescanning.
TIBCO-BW-PALETTE-EDI-506100 `Unsupported encoding [{0}].`	errorRole	EDI-Plug-in ERROR_EDIFACTPRESCANNER	The specified encoding option is not supported.	Ensure a supported encoding type is selected: UTF-8, UTF-16, UTF-16BE, UTF-16LE, ISO-8859-1.
TIBCO-BW-PALETTE-EDI-506101 `Invalid EDIFACT document.`	errorRole	EDI-Plug-in ERROR_EDIFACTPRESCANNER	The specified EDIFACT document is invalid.	Supply a valid EDIFACT document for prescanning.
TIBCO-BW-PALETTE-EDI-507000 `MESSAGE={0}`	errorRole	EDI-Plug-in ERROR_SHUFFLER	This error points to a Shuffler return code. For example: `TIBCO-BW-PALETTE-EDI-507000:return code [2], error message : STC default file name failed.`	Using the return code provided in the message, refer to the Shuffler Return Codes section of the *TIBCO Foresight® Instream® MAC Adapter Introduction* PDF document.
TIBCO-BW-PALETTE-EDI-507001 `Input parameter must be supplied, {0}`	errorRole	EDI-Plug-in ERROR_SHUFFLER	An input file path and name is required.	Specify an input file path and name.
TIBCO-BW-PALETTE-EDI-507002 `Config parameter must be supplied, {0}`	errorRole	EDI-Plug-in ERROR_SHUFFLER	A configuration file path and name is required.	Specify a configuration file path and name.

Error Code and Message	Role	Category	Description	Resolution
TIBCO-BW-PALETTE-EDI- 507003 `Instream result parameter must be supplied,{0}`	errorRole	EDI-Plug-in ERROR_SHUFFLER	An Instream Detail file path and name is required.	Specify an Instream Detail file path and name.
TIBCO-BW-PALETTE-EDI-507004 `output file name could not empty.`	errorRole	EDI-Plug-in ERROR_SHUFFLER	An output file name is required.	Specify a name for the output file.
TIBCO-BW-PALETTE-EDI-507009 `Open STC files {0} failed.`	errorRole	EDI-Plug-in ERROR_SHUFFLER	The specified STC file could not be opened.	Ensure the permissions on the file allow open access.
TIBCO-BW-PALETTE-EDI-507010 `Open map file {0} failed.`	errorRole	EDI-Plug-in ERROR_SHUFFLER	The specified map file could not be opened.	Ensure the permissions on the file allow open access.
TIBCO-BW-PALETTE-EDI-507011 `Open instream result file {0} failed.`	errorRole	EDI-Plug-in ERROR_SHUFFLER	The specified Instream result file could not be opened.	Ensure the permissions on the file allow open access.
TIBCO-BW-PALETTE-EDI-507012 `Read instream result file {0} failed.`	errorRole	EDI-Plug-in ERROR_SHUFFLER	The specified Instream result file could not be read.	Ensure the permissions on the file allow open access.
TIBCO-BW-PALETTE-EDI-507013 `Parsing input data failed.`	errorRole	EDI-Plug-in ERROR_SHUFFLER	The input data could not be parsed.	Ensure input file formatting is valid.
TIBCO-BW-PALETTE-EDI-507014 `Parse table {0} failed {1}.`	errorRole	EDI-Plug-in ERROR_SHUFFLER	DEBUG_MESSAGE ONLY	N/A
TIBCO-BW-PALETTE-EDI-507015 `{0} {1}`	errorRole	EDI-Plug-in ERROR_SHUFFLER	DEBUG_MESSAGE ONLY	N/A

Error Code and Message	Role	Category	Description	Resolution
TIBCO-BW-PALETTE-EDI-507017 `Open output file {0} failed.`	errorRole	EDI-Plug-in ERROR_SHUFFLER	The specified output file could not be opened.	Ensure the permissions on the file allow open access.
TIBCO-BW-PALETTE-EDI-508000 `{0}Config parameter must be supplied.`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	A configuration file path and name is required.	Specify a file path and name for the configuration file.
TIBCO-BW-PALETTE-EDI-508001 `Input parameter must be supplied.`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	An input file path and name is required.	Specify a file path and name for the input file.
TIBCO-BW-PALETTE-EDI-508002 `INVALIDARGUMENT_MESSAGE={0}`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	One or more arguments are invalid.	Refer to the supporting text, which appears in `{0}` for more information.
TIBCO-BW-PALETTE-EDI-508003 `File {0} could not be opened.`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	The specified file could not be opened.	Ensure the permissions on the file allow open access.
TIBCO-BW-PALETTE-EDI-508004 `IOException {0}.`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	An IOException occurred.	This is an internal product error. Contact TIBCO Support.
TIBCO-BW-PALETTE-EDI-508005 `{0}`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	This error points to a ISErrorRefiner return code. For example: `TIBCO-BW-PALETTE-EDI-508005:return code [4], InStream results file must start with a VERsion record. (at line # n)"`	Using the return code provided in the message, refer to the ISErrorRefiner Return Codes section of the *ISErrorRefiner* PDF document.

Error Code and Message	Role	Category	Description	Resolution
TIBCO-BW-PALETTE-EDI-508006 `Illegal severity value {0} found in input file. Unable to continue.`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	The input file contains one or more invalid Error Severity codes.	Analyze the input file for invalid Error Severity codes (i.e., codes less than zero or greater than 6).
TIBCO-BW-PALETTE-EDI-508007 `Illegal type value {0} found in input file. Unable to continue.`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	The input file contains one or more invalid HIPAA Type codes.	Analyze the input file for invalid HIPAA Type codes (i.e., codes less than zero or greater than 8).
TIBCO-BW-PALETTE-EDI-508008 `STRUCT={0}`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	A structure-related error has occurred.	Refer to the supporting text, which appears in `{0}` for more information.
TIBCO-BW-PALETTE-EDI-508009 `RESULTFILE={0}`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	A result file-related error has occurred.	Refer to the supporting text, which appears in `{0}` for more information.
TIBCO-BW-PALETTE-EDI-508010 `Error on loading configure file {0}`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	The specified configuration file could not be loaded.	Ensure the configuration file is valid.
TIBCO-BW-PALETTE-EDI-508011 `Access configure file {0} failed.`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	The specified configuration file could not be accessed.	Ensure the configuration file appears in the specified location.
TIBCO-BW-PALETTE-EDI-508020 `The output directory is required.`	errorRole	EDI-Plug-in ERROR_ERRORREFINER	An output directory is required.	Specify a valid output directory.
TIBCO-BW-PALETTE-EDI-509000 `Input parameter must be supplied.`	errorRole	EDI-Plug-in ERROR_DOCSPLITTER	No Input Mode parameter was selected.	Select an Input Mode parameter: File or Memory.

Error Code and Message	Role	Category	Description	Resolution
TIBCO-BW-PALETTE-EDI-509001 `InstreamResult parameter must be supplied.`	errorRole	EDI-Plug-in ERROR_DOCSPLITTER	An Instream Detail file path and name is required.	Specify a file path and name for the Instream Detail file.
TIBCO-BW-PALETTE-EDI-509002 `{0}`	errorRole	EDI-Plug-in ERROR_DOCSPLITTER	This error points to a DocumentSplitter return code. For example: `TIBCO-BW-PALETTE-EDI-509002 :return code [205], Docsplitter could not open the INI config file.`	Using the return code provided in the message, refer to the DocumentSplitter Return Codes section of the *Document Splitter* PDF document.
TIBCO-BW-PALETTE-EDI-509003 `Output parameter must be supplied.`	errorRole	EDI-Plug-in ERROR_DOCSPLITTER	No Output Mode parameter was selected.	Select an Output Mode parameter: File or Memory.
TIBCO-BW-PALETTE-EDI-509004 `Invalid output directory {0}.`	errorRole	EDI-Plug-in ERROR_DOCSPLITTER	The specified output directory is invalid.	Specify a valid output directory.

EDI Plug-in Example Processes

The TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI examples are packaged with the installation of the plug-in.

This information describes how to run the EDI Plug-in examples.

By default, the following examples are located in the <*BW Home*> \palettes\EDI*x.x*\samples directory:

- Shuffler and IsErrorRefiner (ShufflerIsErrorRefiner)
- Validate and Translate (ValidateTranslate)

Setting up and Running the Examples

Use the following information to set up and run TIBCO ActiveMatrix™ BusinessWorks Plug-in for EDI examples.

You should be familiar with the plug-in before running the examples.

Copy Guidelines and Maps

The EDI Plug-in examples makes use of guidelines and maps. Before using any of the examples, you must copy the associated files to the appropriate directory.

For each example, **guidelines** and **map** files are found in the example's \Guidelines directory. For example: `\bw\palettes\edi\x.x\samples\ValidateTranslate\DataAndGuidelines\Guidelines`.

Copy the guideline and map files as follows:

1. Copy the .std, .xsd, and .map files to TIBCO Foresight® Translator's \Database directory.
2. Copy the .std files to TIBCO Foresight® Instream®'s \Database directory.

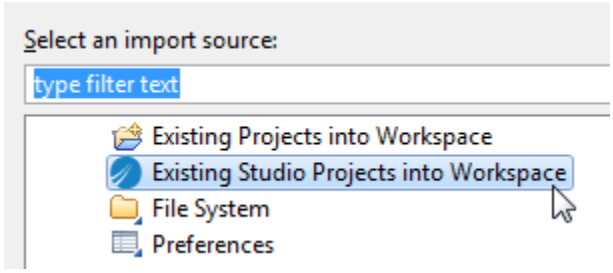
(Note that the .std files are copied to the \Database directory for BOTH Instream® and Foresight® Translator.)

Importing an Example

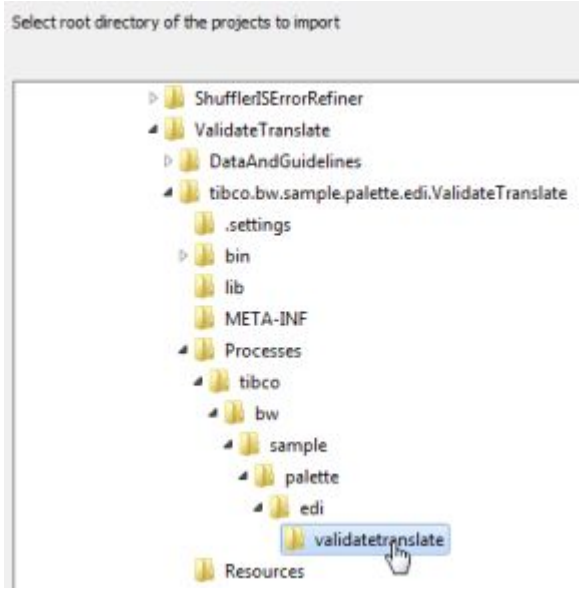
Use the following steps to import an example.

1. Select **File > Import**.

   The Import screen appears.
2. Select Existing Studio Projects into Workspace.



   The Browse For Folder screen appears.
3. Browse to the location of the desired EDI samples. By default this is the <*BW_Home*> \palettes\edi *x.x*\samples directory, as shown here:



The path will be different if you have saved the samples to another location.

Optional: Set Environment Variables for the Process

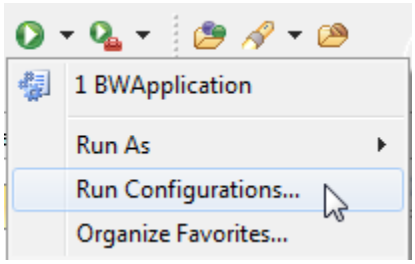
Instream and Translator environment variables should have been set as part of the Plug-in for EDI Post-installation tasks (refer to *TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI Installation*). Doing so tells ActiveMatrix BusinessWorks where to access the Instream and Translator executables when the products are used in a TIBCO ActiveMatrix™ BusinessWorks EDI Plug-in process.

It is also possible to specify environment variables on a per-process basis. The following procedure instructs the process to utilize the variables specified for the process instead of those specified for TIBCO ActiveMatrix™ BusinessWorks at the higher level.

Refer to the appropriate section for your operating system:

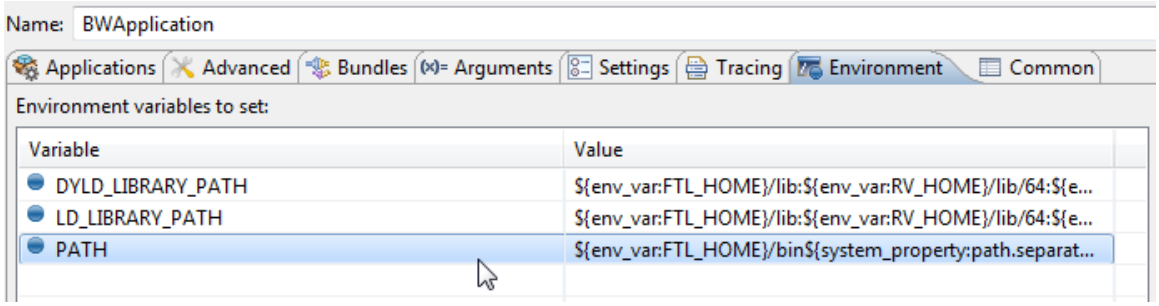
For Windows Platforms

1. Open the desired process and click within the process.
2. Select **Run > Run Configurations...**.



The Run Configuration screen appears.

3. Select the environment tab and double click on PATH.



The Edit Environment Variable screen appears.

4. Add the path to the desired location of the Instream\bin and Translator\bin directories at the beginning of the string, before the first $ character, and with a **;** at the end of each segment of added text.

   Do NOT delete the existing string, simply add information to the beginning.

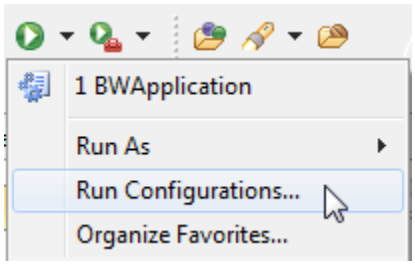
   Example:

   C:\tibco64\instream\8.5\bin**;**C:\tibco64\translator\3.5\bin**;**${env_var:FTL_HOME}.....

5. Select the Apply button.
6. Select the Close button.

For Unix Platforms

1. Open the desired process and click within the process.
2. Select **Run > Run Configurations...**.



The Run Configuration screen appears.

3. Select the environment tab and double click on LD_LIBRARY_PATH.

   The Edit Environment Variable screen appears.

4. Add the path to the desired location of the Instream/bin and Translator/bin directories at the beginning of the string, before the first $ character, and with a **:** at the end of each segment of added text.

   Do NOT delete the existing string, simply add information to the beginning.

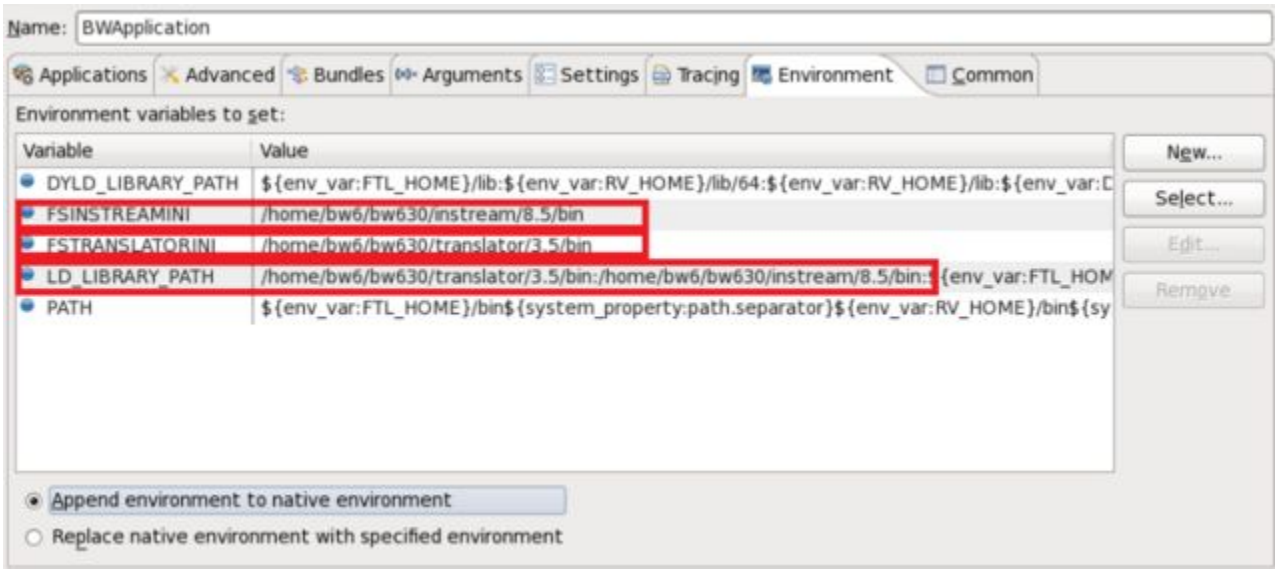
   Example

   /home/bw/bw630/instream/8.5/bin**:**/home/bw/bw630/translator/3.5/bin**:**${env_var:FTL_HOME}.....

5. Select the Apply button.
6. Select the New button.
7. Add a new variable FSINSTREAMINI and specify the desired location of the Instream\bin directory.

8. Select the Apply button.
9. Select the New button.
10. Add a new variable FSTRANSLATORINI, and specify the desired location of the Translator\bin directory.
11. Select the Apply button.
12. Select the Close button.

This example shows a sample Run Configuration screen after the variables have been set for Unix.

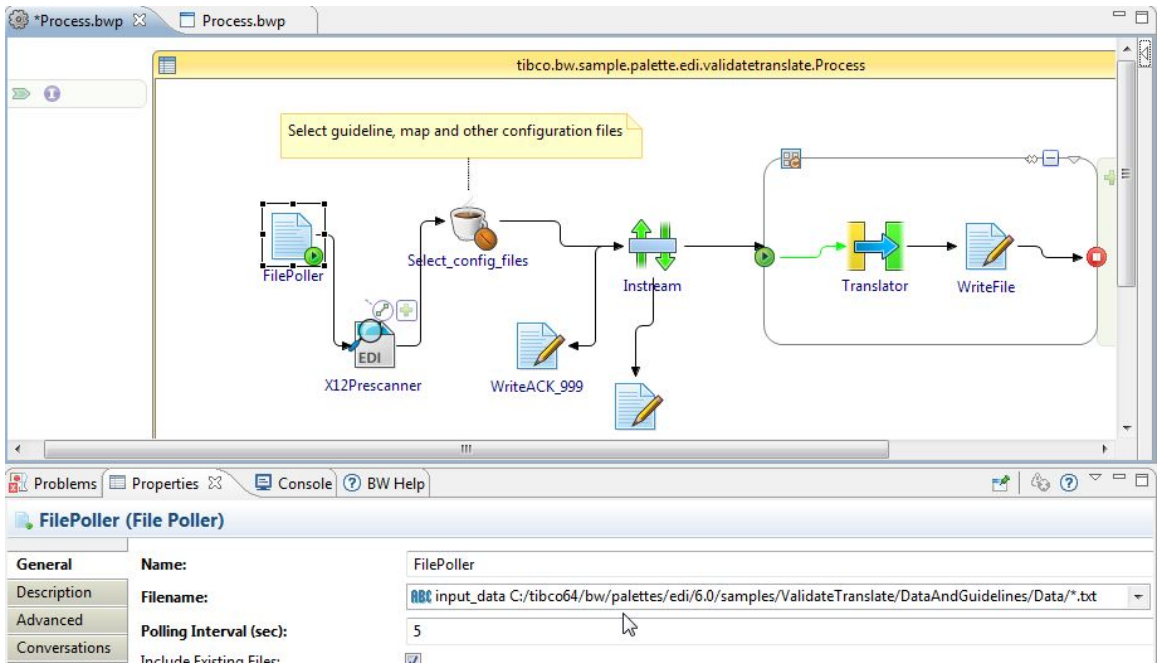


Verify File Paths in the Examples

Before running an example, verify the file path used in the example is correct for your environment.

1. Open the desired example.

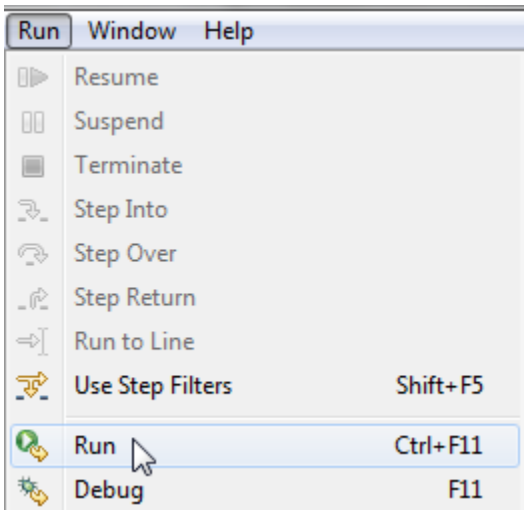
   In this image, the ValidateTranslate.bwp process is opened.



2. Check the file path. In this example the FilePoller activity is set to pick up the input file at the following path `<BW_HOME>\samples\palette\edi\x.x\samples\ValidateTranslate/DataAndGuidelines/Data/*.txt`. You may need to alter the file path if you have saved the samples to a different location.

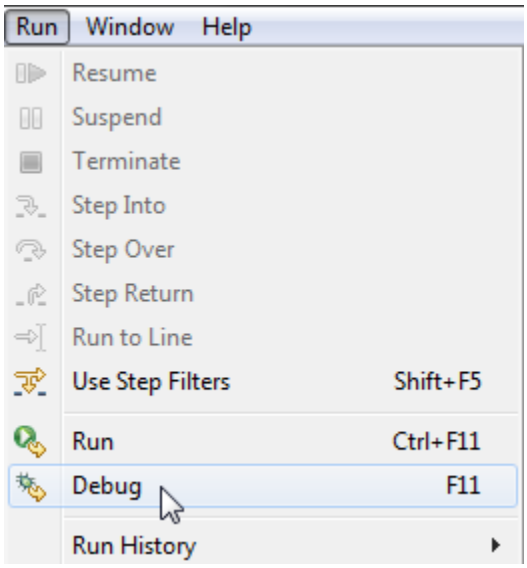
Running an Example

1. Open the desired example.
2. Click **Run > Run** to run the application.



After the process executes successfully, a Success message is written to the Console. Select **Window > Show Console** in TIBCO Business Studio™ to view the log messages.

3. Optional: Click **Run > Debug** to debug the application.



The perspective changes to Debug from Modeling. You can view the job details in the Console view from the Debug perspective.

Shuffler and ISErrorRefiner Example

The `\bw\palettes\edi\x.x\samples\ShufflerISErrorRefiner` folder contains a sample project that helps you understand how use the Shuffler and ISErrorRefiner activities to utilize data that has been

validated by Instream to convert EDI to and from flat file format for compliance with the CMS-supplied Claims Edit Module (CEM) requirements for electronic processing of claims.

The Shuffler and ISErrorRefiner activities are available **only** to customers who have purchased the TIBCO Foresight® Instream® MAC Adapter. Contact your TIBCO Foresight Account Representative for more information about Instream® MAC Adapter.

For more information about understanding and using this sample project, contact TIBCO Foresight Technical Support.

Validate and Translate Example

The `\bw\palettes\edi\x.x\samples\ValidateAndTranslate` folder contains a sample project that helps you understand how to validate input data, swap data, and translate data from one format to another.

For more information about validating and translating data, see Instream and Translator.

Example Data File

This sample project makes use of the following data file found in the `\bw\palettes\edi\x.x\samples\ValidateTranslate\DataAndGuidelines\Data` area:

- `837I_4010_H_5provider.txt` - The EDI text file that triggers the ValidateTranslate process.

Example Guideline and Map Files

This sample project makes use of the following guideline and map files found in the `\bw\palettes\edi\x.x\samples\ValidateTranslate\DataAndGuidelines\Guidelines` area:

- 837AQ320.std - The schema definition for an 837AQ320 message in ASCII text format. This is the source guideline that describes the message before translation.
- 837AQ320_837.xsd - The guideline for an 837AQ320_837 message in XML format. This is the target guideline that describes the message after translation.
- 837AQ320_837_EX.map - The predefined map file that is used to translate the 837AQ320 message from EDI format into XML format.
- 837AQ320_837_XE.map - The predefined map file that is used to translate the 837AQ320 message from XML format into EDI format.
- Content_Based_Split_Auto_Setup.ini
- PCBS.std
- sample.apf

Process Description

This example contains the following predefined process:

- ValidateTranslate Process

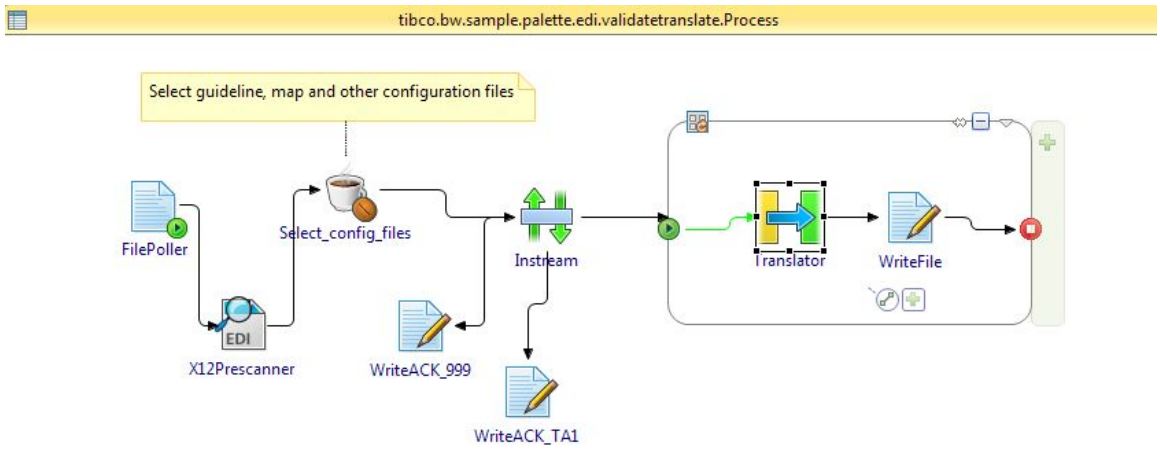
ValidateTranslate Process

The ValidateTranslate example shows how to validate input data and translate data from one format to another.

For more information about validating and translating data, see Instream and Translator.

Example Process

The ValidateTranslate process is illustrated here:



Process Definition

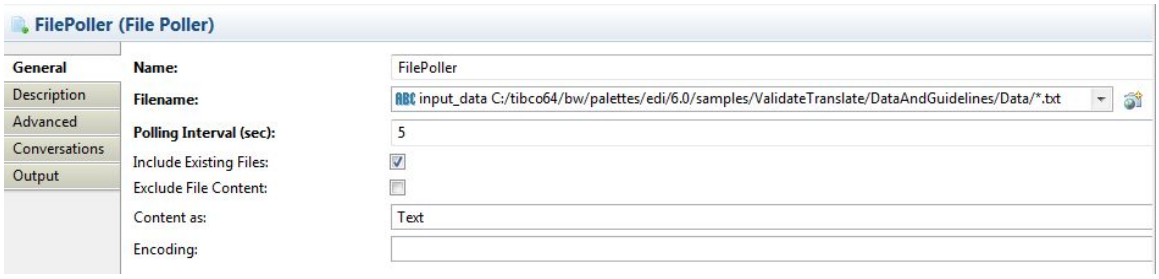
This process demonstrates how the Validation and Translation functions work together to process an inbound EDI. The process performs the following steps:

1. The FilePoller activity polls the input directory. If a file exists then the process starts.
2. The X12 Prescanner activity scans the X12 EDI.
3. A Java activity selects the appropriate configuration files based on the data contained in the EDI.
4. The Instream activity generates a 999 Acknowledgment response.
5. The Instream activity validates the input EDI and generates a TA1 Acknowledgment response.
6. The Translator activity translates the EDI to a XML file format and writes the translated data to a file.

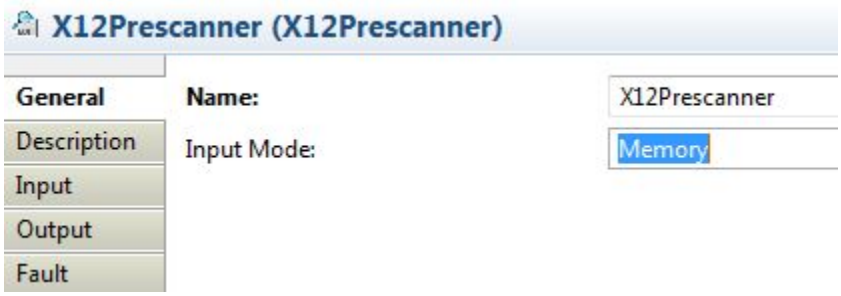
Prescanning

This activity performs the following operations:

1. The FilePoller activity polls for a file named *.txt in the directory specified in the File Name field of the Configuration tab. It finds the 837I_4010_H_5provider.txt file and the process starts.



   The 837I_4010_H_5provider.txt file is passed to the X12Prescanner activity.
2. The X12Prescanner activity parses the X12 document header (ISA/GS/ST) segments and saves each element into a separate output field.



3. The information is passed to the Java Activity named select_config_files.

For more information about prescanning data, see EDIFACTPrescanner and X12Prescanner.

Selection of Configuration Files

This activity performs the following operations:

1. The select_config_files Java activity examines the data provided by the X12 Prescanner activity and uses the information from the GS01, GS08, and ST01 elements to select the appropriate validation guideline, translation map, and configuration profiles.

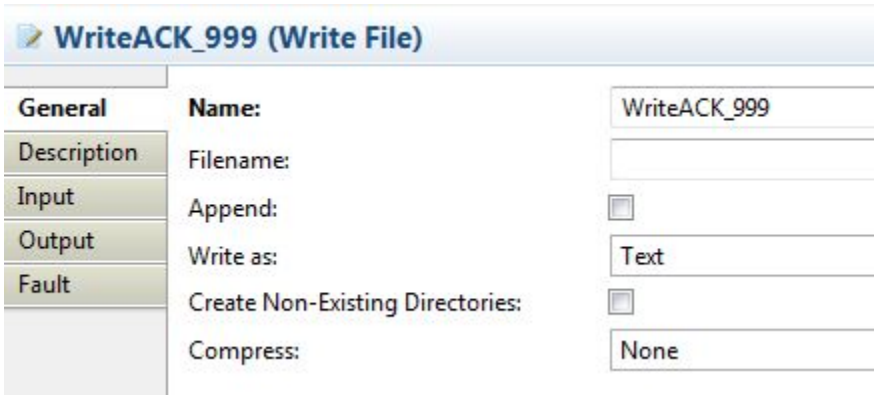
Field Type	Field Name
IN	GS01
IN	GS08
IN	ST01
OUT	Guideline
OUT	Map
OUT	Dos_profile
OUT	INS_profile

2. The select_config_files Java activity passes the information to both the Write File activity named WriteACK_999 and the Instream activity.

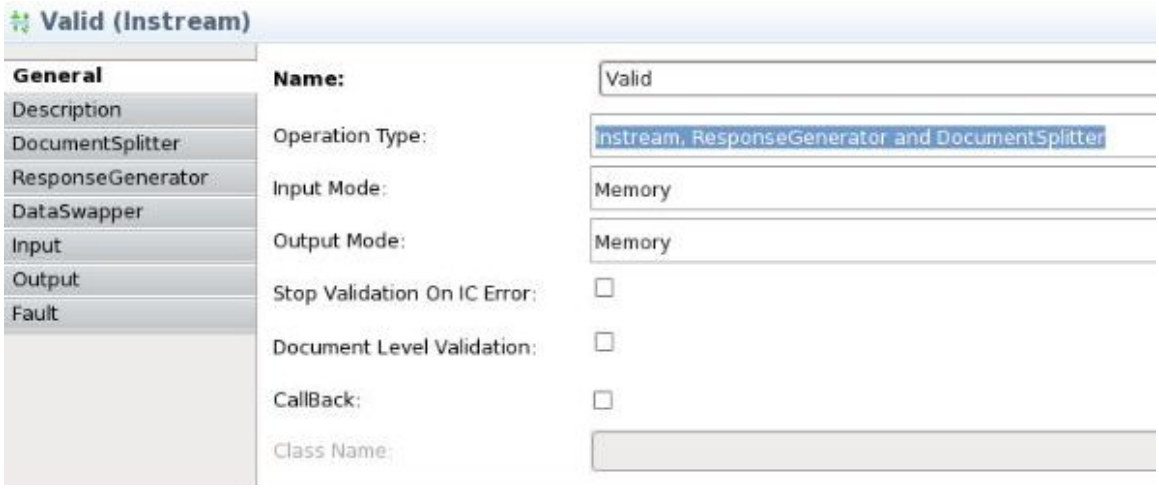
Validation and Response Generation

This activity performs the following operations:

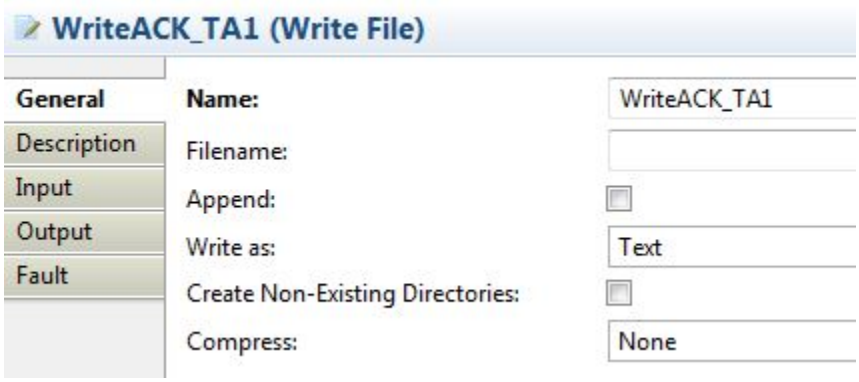
1. The Instream activity generates a 999 Acknowledgment response document, which reports the status of implementation guide syntax edits.



2. The Instream activity validates the input EDI (837I_4010_H_5provider.txt) using the validation guideline and configuration profiles selected by the select_config_files Java activity.



3. The Instream activity generates a TA1 Acknowledgment response document, which acknowledges the receipt of the contents of the transaction.



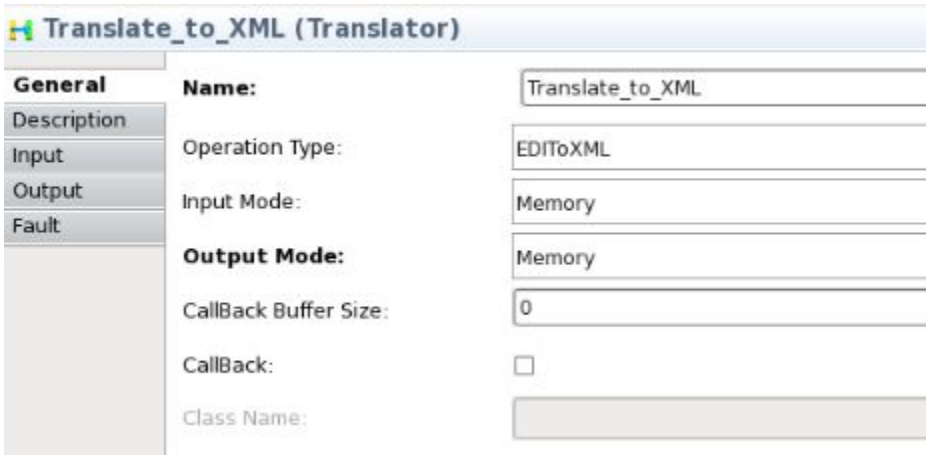
4. The TA1 Interchange Acknowledgment is saved as a text file.
5. The validated file is passed to the Translator activity.

For more information about validating data, see Instream.

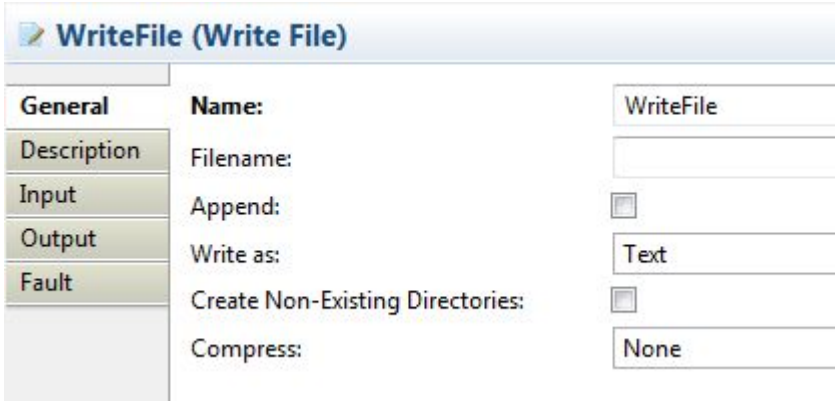
Translation

This process performs the following operations:

1. The Translator activity translates the validated file from EDI to Flat File format using translation maps specified by the select_config_files Java activity.



2. The Translator activity generates a file containing the translated output data.



For more information about translating data, see Translator.

Appendix A: Electronic Data Interchange

This appendix provides a basic introduction to Electronic Data Interchange (EDI).

Overview

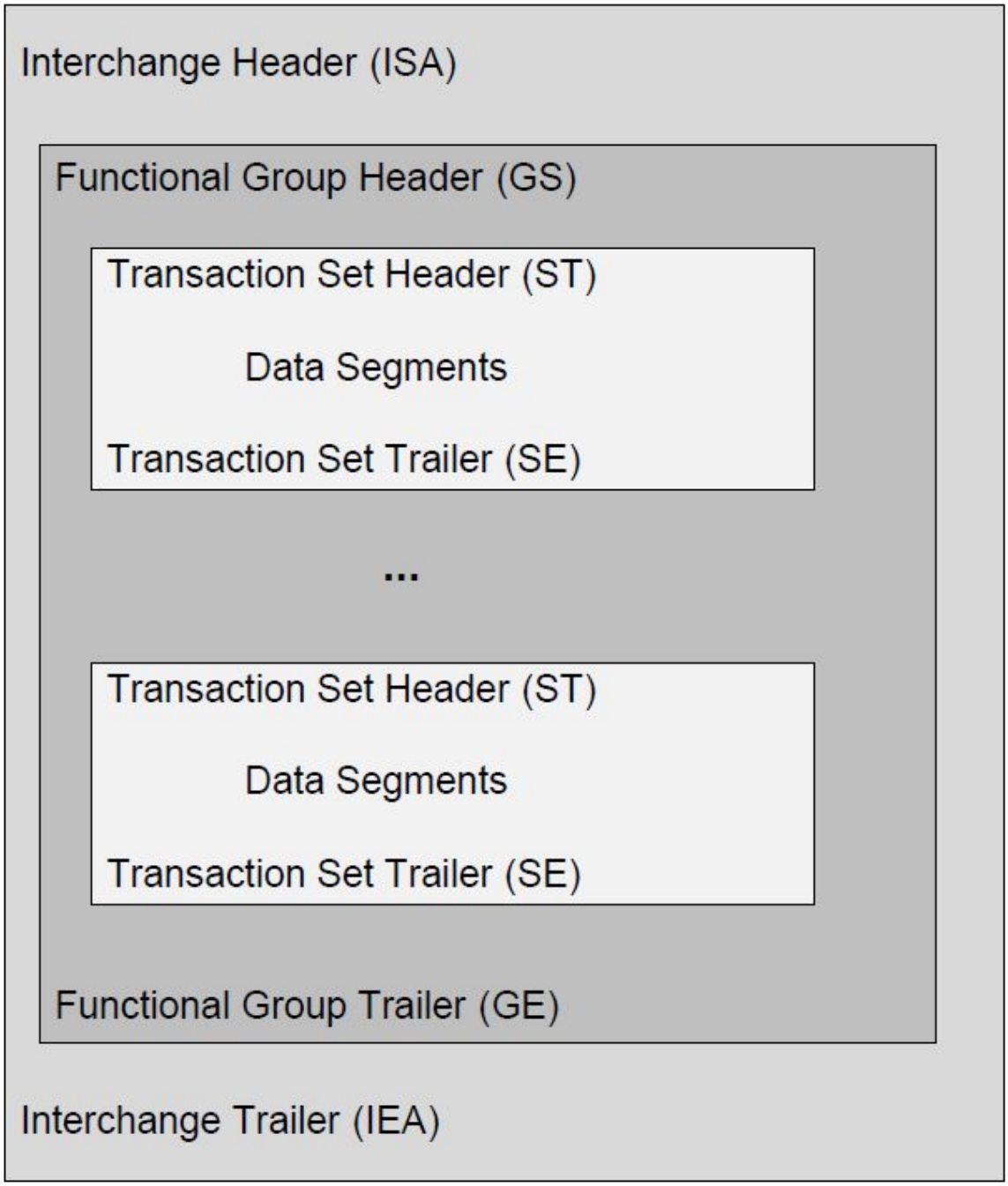
The EDI Palette contains EDI activities for TIBCO ActiveMatrix BusinessWorks™ Plug-in for EDI. Electronic Data Interchange (EDI) is a standardized messaging framework developed by industry groups for exchanging information between trading partners in a structured, predetermined format.

Structure

An EDI file contains a structure known as enveloping.

Enveloping is the way EDI ensures file integrity and lets the message determine its destination and type.

This figure shows an X12 EDI structure example:



Based on the figure, we see that:

- The interchange is the basic unit of an EDI file. Several interchanges can be bundled into a single file for data transfer.

  An interchange starts with an interchange header (ISA) and ends with an interchange trailer (IEA).
- A functional group is a group of similar transaction sets which have the same functions.

  A functional group starts with a function group header (GS) and ends with a functional group trailer (GE).
- A transaction set contains the information required by a receiver to perform a standard business transaction.

  A transaction set starts with a transaction set header (ST) and ends with a transaction set trailer (SE).

- Data segments are used to make up transaction sets. Data segments are roughly equivalent to a single line on a document. Data elements contain the basic units of information and are used in various combinations to make up data segments.

Standards

EDI standards (protocols) help facilitate EDI by providing a common format or rules of data structure and transmission protocols.

The objective is to use agreed upon structure for communicating the data in business documents. Each organization that maintains a set of standards provides a full set of documentation and definitions of each version of their standards. The following are brief introductions to some of the standards:

X12

X12 standards are EDI standards developed by the Accredited Standards Committee (ASC) X12. The ASC X12 are chartered by the American National Standards Institute (ANSI).

In the X12 standard, a transaction set contains the data for a well-defined business function (for example, a purchase order). There are more than three hundred X12 transaction sets used by more than thirty thousand organizations for nearly every facet of business-to-business operations.

HIPAA

The Health Insurance Portability and Accountability Act (HIPAA) is a specific set of messages based on the X12 standard, which is used for healthcare-related information exchange.

EDIFACT

EDIFACT stands for Electronic Data Interchange for Administration, Commerce, and Transport. EDIFACT refers to a set of international standards, directories, and guidelines for the electronic interchange of structured data between separate computer systems. The EDIFACT standards are supported by the United Nations Economic Commission for Europe (UN/ECE).

Flat File

Flat File standards are used to validate Flat File data.

XML

XML standard is used to validate XML data and to create HTML reports of validation results.

Appendix B: Response Documents (Acknowledgments) and Custom Reports

This appendix provides a brief introduction to the Response Documents (also called Acknowledgments) supported by TIBCO ActiveMatrix BusinessWorks Plug-in for EDI.

Overview of Acknowledgments

Acknowledgments are used to give feedback to the sender of a transaction on the status of the acceptance of the transaction by its recipient. For X12, acknowledgments such as 997 and 999 are used to provide feedback to trading partners on the validation success or failure of transactions they have sent. For EDIFACT, the CONTRL acknowledgment is used for this purpose. For HIPAA, there are acknowledgments used in the health care industry to report on the validation success or failure of business level application edits. These business-level acknowledgments are the 277 Unsolicited Health Care Claim Status Notification (277 U), 277 Claim Acknowledgment (277CA), and the 824 Application Advice.

Refer to Types of Response Documents for additional information.

Overview of Custom Reports

In addition to standardized acknowledgments like 824 and 999, Response Generator can be used to create customized reports from transactions validated by Instream.

This is done by creating a custom report template specifying the information you want to see and directing Response Generator to generate a text report using the specified template.

Refer to Response Generator Custom Reports for additional information.

Types of Response Documents

The following section lists a brief description of the acknowledgments used by X12, HIPAA, and EDIFACT.

227CA/U Acknowledgments

The 277 Claim Acknowledgment (277CA) is sent by a payer in response to an 837 to report on whether the pre-adjudication validation found them acceptable for adjudication.

The 277 Unsolicited Health Care Claim Status Notification (277U) reports the results of an application system's data content edits on the claims in a transaction set.

824 Application Advice

The 824 Application Advice (824) reports the results of an application system’s data content edits of a transaction set. Editing transaction sets results can be reported at the functional group and transaction set level, in either coded or free-form format.

997 Functional Acknowledgment

The Functional Acknowledgment (997) describes the syntax-level acknowledgment of the receipt of an X12 functional group. It tells a sender that a receiver has received EDI transactions successfully.

The transaction set can be used to define the control structures for a set of acknowledgments to indicate the results of the syntactical analysis of the electronically encoded documents. The encoded documents are the transaction sets (which are grouped in functional groups) used in defining transactions for business data interchange. This standard does not cover the semantic meaning of the information encoded in the transaction sets.

999 Implementation Acknowledgment

The Implementation Acknowledgment (999) was first available in the X12 004061 subrelease. It is used for reporting the status of implementation guide syntax edits.

The transaction set can be used to define the control structures for a set of acknowledgments to indicate the results of the syntactical and relational analysis of electronically encoded documents, based on a full or implemented subset of X12 transaction sets. The encoded documents are the transaction sets (which are grouped in functional groups) used in defining transactions for business data interchange. This standard does not cover the semantic meaning of the information encoded in the transaction sets.

TA1 Interchange Acknowledgment

An Interchange Acknowledgment segment (TA1) is a delivery acknowledgment. It reports the receipt of the contents of one interchange control header and trailer envelope in which the envelope surrounds one or more functional groups. The TA1 reports the results of the syntactical analysis of the interchange control header and trailer. Each interchange exchanged between trading partners may contain interchange-level control segments (TA1s) related to prior interchanges.

CONTRL Document (EDIFACT Responses)

When EDIFACT documents are exchanged between trading partners, an acknowledgment of the receipt and syntactical validation of the document can be returned to the sender of the document.

When exchanging EDIFACT documents, the CONTRL document is used to syntactically acknowledge or reject, with error indication, an interchange, group, or message received from a trading partner.

Note that CONTRL is the only message for acknowledgment in EDIFACT. In contrast, in ANSI X12 there are two messages used for acknowledgment: TA1 and 997/999 Acknowledgment.

Use of a CONTRL message for acknowledging receipt of an interchange, group, or message is not required in EDIFACT. The use of CONTRL is by trading partner agreement.

Response Generator Custom Reports

You can use Response Generator to create customized reports from transactions validated by Instream for the following transaction types:

- 270 - Eligibility, Coverage or Benefit Inquiry
- 276 - Health Care Claim Status Request
- 277U - Health Care Claim Status Response – Unsolicited (4010 only)
- 278 - Health Care Services Request for Review
- 820x218 - Payroll Deducted and Other Group Premium Payment for Insurance Products
- 820x306 - Health Insurance Exchange: Related Payment
- 834 - Benefit Enrollment and Maintenance
- 835 - Health Care Claim Payment/Advice
- 837 - Health Care Claims

The report can contain literal text, values from the data, and error information.

Refer to *Response Generator Technical Manual* PDF document for details.

Create a Custom Report

To create a custom report you must:

1. Create a custom report template.

   Refer to *Response Generator Technical Manual* PDF document for details.

2. Use the Instream or ResponseGenerator activity to produce a custom report and specify where to find the custom report template.

   Refer to Instream or ResponseGenerator for instructions on how to do this.

Custom Report Example

This example shows a custom report using plain text formatting.

```
Error Report on: 02/12/2006 for Interchange Number 000000386
-------------------------------------------------------------

LastName (Insured): SMITH
FirstName (Insured): MUFFY
LastName (Patient):
FirstName (Patient):
ClaimNumber: 1
TotalAmt: $100.00
StatementDate: 20030212-20030213
Error: :

LastName (Insured): SMITH
FirstName (Insured): MUFFY
LastName (Patient):
FirstName (Patient):
ClaimNumber: 10
TotalAmt: $1000.00
StatementDate: 20030212-20030213
Error: 10613: Svc: Element SV204 (D.E. 355) at col. 26 is missing,
though marked "Must Be Used"
       10613  Svc: Element SV205 (D.E. 380) at col. 26 is missing,
though marked "Must Be Used"
```

This example shows a custom report using column formatting.

```
Claim #      Claim Amount   Err #   Error Description
----------   ------------   -----   -------------------------

2                $200.00    10811   Clm: Missing Segment DTP (Stat
                                    ement Dates) at 2-135, though
                                    marked "Must Be Used"

4                $400.00    30354   Svc: The Previous Claim does n
                                    ot balance. Formula used CLM02
                                     (400.00) = sum of SV203 (665.
                                    75)

6                $600.00    10605   Clm: Code Value "NDDDGDN" not
                                    found in the dictionary code l
                                    ist for PWK01 (D.E. 755) at co
                                    l. 5
```

Refer to *Response Generator Technical Manual* PDF document for details.