Infor M3 CE Core Administration Guide

Release 16.0.x

Contents

About this guide

This guide describes M3 CE Core administration at the customer site, after provisioning of a specific tenant. This guide also includes installation instructions for components included in M3 CE Core that must be installed on premises at the customer site.

Intended audience

This guide is intended for system administrators who are responsible for administration tasks for M3 CE Core.

Prerequisite knowledge

To fully understand the information presented in this guide, you must meet these requirements:

- Understand the setup in M3 Business Engine
- Have experience setting up OpenText Exstream
- Know Infor Ming.le™ administration tasks
- Have experience in managing applications in Infor ION Grid

Related documentation

You can find these documents in docs.infor.com.

Document	Notes
Infor ION Grid Administration Guide	This document is found in the Infor OS online documentation.
Infor ION Grid Administration Guide	This document is found in the Infor OS online documentation.
Infor ION Desk User Guide - Cloud Edition	This document is found in the Infor OS online documentation.
Infor Document Management Administration Guide - Cloud Edition	This document is found in the Infor OS online documentation.

Document	Notes
Infor Ming.le® Online Help	This document is found in the Infor OS online documentation.
Infor M3 Cloud Configuration Guide	This document is found in M3 Cloud.
M3 Application Foundation User Guide	This document is found in the M3 Business Engine online documentation.

Contacting Infor

If you have questions about Infor products, go to Infor Concierge at https://concierge.infor.com/ and create a support incident.

The latest documentation is available from docs.infor.com or from the Infor Support Portal. To access documentation on the Infor Support Portal, select **Search > Browse Documentation**. We recommend that you check this portal periodically for updated documentation.

If you have comments about Infor documentation, contact documentation@infor.com.

Chapter 1: M3 CE Core overview

For companies in the business to Make, Move, and Maintain products, Infor M3 provides the tools they need to manage their operations. With the full system, they can monitor transactions throughout their organization, including manufacturing operations, supply chain activities, customer and supplier relationships, warehouse and distribution processes, and financial management. Infor M3 consolidates these into one coherent view of operations.

While the complete M3 solution is composed of several different software components, where some are unique for just a certain industry or micro vertical, there are components that are common for all M3 instances regardless of industry. These components are gathered under the product family named M3 Core. In the cloud context, this product family is called M3 CE Core.

Chapter 2: IFS security roles for M3 CE Core

If any of the required IFS security roles are missing, you must add these roles manually through the User Management menu in Infor Ming.le™. To add new security roles, you must be connected to the *UserAdmin* role.

See information about adding a security role in the *Infor Ming.le Cloud Edition Online Help*.

Required IFS security roles

This table shows the list of IFS security roles that are required to accomplish administration tasks for each M3 application type:

Application type	IFS Security role	Description
Grid	*GRID-Administrator*	Users with this role can download the CFT Client installation.
M3 Business Engine	*M3BE-FndAdmin*	Users with this role can list, kill, configure logging and log levels, and configure profiling for all M3 BE jobs on the M3 Business Engine Jobs page, as well as download logs for any M3 BE user on the Business Engine Logs page.
M3 Business Engine	*M3BE-FndUser*	Users with this role can list and kill their own M3 BE jobs on the Business Engine Jobs page, as well as download their own dump logs on the Business Engine Logs page.
M3 Business Engine	*M3BE-LogEnabler*	Users with this role can configure logging, log levels, and profiling for their own M3 BE jobs on the M3 Business Engine Jobs page.

Application type	IFS Security role	Description
M3 Business Engin	*M3BE-ConfigAdmin*	Users with this role can import M3 Business Engine configuration data as any M3 user, as well as manage configuration data files on the Business Engine Files page.
M3 Business Engine	*M3BE-ConfigUser*	Users with this role can import configuration data, as well as view and download configuration data files on the Business Engine Files administration page.
M3 Business Engine	*M3BE-DBViewer*	Users with this role can perform read operations on the M3 BE database using the Update Database administration tool in the Business Engine Data Management page.
M3 Business Engine	*M3BE-DBAdmin*	Users with this role can perform read and write operations on the M3 BE database in the Business Engine Data Management pages. Users can also manage BE data management files on the Business Engine Files page.
M3 Business Engine	*M3BE-DBAdmin-ReadToken*	Users with this role can create and revoke read tokens, which are required for direct data copy in the Business Engine Data Management pages.
M3 Business Engine	*M3BE-FATAdmin*	Users with this role can administer field audits on the Business Engine Field Audit Trail page.
M3 Business Engine	*M3BE-FATAdmin-Exporter*	Users with this role can export field audit data on the Business Engine Field Audit Trail page. Users can also manage FAT Export files on the Business Engine Files page.
M3 Business Engine	*M3BE-ServiceAdmin*	Users with this role can administer interactive program services on the Business Engine Interactive Program Services page.
M3 Business Engine	*M3BE-FileAdmin*	Users with this role can manage MvxFileTransfer files on the M3 Business Engine Files page.

Application type	IFS Security role	Description
M3 Business Engine	*M3BE-FileViewer*	Users with this role can view and download MvxFileTransfer files on the M3 Business Engine Files page.
M3 UI Adapter	*M3UI-Administrator*	Users with this role can perform all administration tasks for M3 H5.
Event Hub CE	*EVENTHUB-Administrator*	Users with this role can view and change configuration settings in Event Hub and Event Analytics.
Data Lake Publisher	*M3-DLSubscriptionAdmin*	Users with this role can control access to the admin interface where the user can manage which documents M3 will publish to Data Lake.
Mashup	*MASHUP-Administrator*	Users with this role can perform all administration tasks that are necessary in managing mashup applications.
MDP	*M3CE_About_Viewer*	Users with this role can view M3CE About. This web application is packaged together with MDP. M3CE About enables users to view content packages and the fixes included in those packages.
MDP	*M3MDP-User*	Users with this role can access M3 Metadata Publisher and use Search by System Configuration module.
EC	*M3EC-Administrator*	The administrator role is the primary role for working with BODs from an M3 integration perspective. Users with this role can activate and configure BODs sent from M3 to ION. It can identify, view and rectify failed BODs. It grants administrator privileges in Infor Enterprise Collaborator and the M3 BOD Processor Grid Management Pages.
EC	*M3EC-User*	The user role is limited and aimed at working with M3 data translations from an integration perspective.
EC	*M3EC-ClientDesigntime*	This user role grants permission to perform mapping development in a cloud solution.

Chapter 3: M3CE About overview

M3CE About is a non-tenant aware tool that shows changelog information for a given M3CE Environment.

As an administrator, you must use this function to view the current versions of M3 components in use. This function gives you the ability to determine version numbers for use in reporting issues and monitoring recent updates.

Accessing M3CE About

You can view recent M3 Cloud version updates by logging in to M3CE About:

1   Log in to Infor Ming.le as a tenant administrator.
2   Click **Infor M3**.
3   Expand **Administration Tools** and select **M3CE About**.

Viewing and searching using M3CE About

M3CE About shows you the M3 components available and the recent versions released. Each component will show multiple recent and current versions along with any important information on these versions. You may also expand or collapse M3 component names to reduce the amount of information on the page, by clicking on the triangular button adjacent the M3 component names.

To search for specific M3 component version information, follow these steps:

1   Specify the search information in one of the several search fields in the upper part of M3CE About page.

Component
Shows search results based on the M3 component name

Version
Shows search results based on the M3 component version

Activation Date
Shows search results based on the M3 component version's activation date

Module

Shows search results based on the M3 component module

Summary

Shows search results based on the M3 component's version summary

Issue Type

Shows search results based on the M3 component issue type

Issue Reference

Shows search results based on the M3 issue reference. Infor JIRA shows the issue key. Lawson JIRA shows the JIRA ticket number.

Externally Reported

Shows externally reported values to identify who created the ticket. Externally reported shows "Y" for a customer reported ticket and "N", if not.

NCR

Net Change Report shows the NCR number where the new feature or enhancement was implemented. Consult this NCR for more details on the update.

Feature Type

Shows the information regarding the feature type implemented. The feature type is commonly used for M3 BE Applications. There are five types of features, labeled 1 to 5.

- 1 - Zero Impact
- 2 - Functional Setting
- 3 - Feature Toggle
- 4 - Scheduled
- 5 - Immediately

Clicking the link shows the **Feature Information** window which contains these details:

Item	Definition
Description	Shows the feature's description.
Application Module	Shows the feature's application module.
Country	Shows what country the feature is applicable.
Knowledge Base	Shows the link to the feature's corresponding published KB article.
Released	Shows if the feature has been released. Released shows "Y" if Yes and "N" if No.
Configuration Component	Shows the feature's configuration component.
Completion Date	Shows the feature's completion date.
Update Schedule	Shows the feature's update schedule.

2 Press Enter to show the result.

Chapter 4: Optional post-provisioning steps for M3 CE Core

Using OpenText Exstream for M3 CE Core

OpenText Exstream is a third-party application used to transform M3 Business Engine data into predefined documents or report layouts. Exstream allows many different formats and distribution channels for data including email, mobile, electronic invoice, and printer. Infor has produced an XML StoryTeller Base Project template for Exstream. The Exstream application makes modification and adaptation of the template layouts easier.

OpenText also manages its own cloud. All development and configurations must be done in close cooperation with OpenText environment administrators.

Component	Supported version
OpenText Exstream and OpenText Cloud	16.x

These components are delivered:

- Main projects to create runtime environment
- All projects, source files for the layouts, test data, and example reports to be used for customer modifications
- Three separate dcpackage files in the base package for the M3 Business Engine

Note: The dcpackage format is an internal format.

Description	Related topics
OpenText Exstream uses both StoryTeller and PageOUT layout templates.	For information about the templates, see these topics: • Using PageOUT layout Templates for OpenText Exstream on page 22 • Using StoryTeller layout templates for OpenText Exstream on page 25 These topics are only applicable for OpenText Exstream, not for OpenText Cloud.

Description	Related topics
OpenText Cloud (OTC) or Exstream Cloud also uses StoryTeller Base Project templates for M3. One main difference of OpenText Cloud from the former OpenText is that cloud administrators closely monitor the configuration and handling of the layouts for this application. **Note:** PageOUT layout is not applicable for OTC.	For information about handling the Exstream Cloud, see Using OpenText Cloud on page 37. The configuration topic includes how to set up the StoryTeller layouts in OTC which is handled differently.

Using PageOUT layout Templates for OpenText Exstream

To install layouts for M3 Business Engine, you must accomplish these tasks:

- Download and unpack layouts and language files from the **Cloud File Transfer Agent** Administration page.
- Configure Exstream language handling.
- Create a runtime environment for Exstream.
- Set up M3 BE for Exstream layouts.
- Install patches for languages and layouts, as necessary.

Downloading packages for M3 BE standard, relevant markets, and industry

Depending on the login information supplied, the relevant packages to be downloaded are suggested.

1   Select **Administration Tools > Cloud File Transfer Agent Administration**.
2   Open **Download Contents**.
3   Select the check boxes for Exstream Base Layouts, Exstream Updates, and Languages.
4   Click **Download**.

   A zip file with the selected content is downloaded to your server.

Unpacking the downloaded packages

On your local Windows Exstream server, unpack the downloaded layouts and language files.

1   Open the place on your server where the downloaded files are stored.
2   The file `content.zip` contains zip files for all markets and industries to be installed. Extract the file content.zip first to a temporary location, for example the current folder.
3   Extract the zip file containing the MVX (Standard M3 Business Engine) layouts to `C:\OUT_M3BE_ [version]`, the folder that should contain the M3 BE layouts and languages.

Continue to extract all remaining zip files to the same location.

Infor delivers these projects for Exstream v16x:

- Main projects to create runtime environment for all standard programs and market modifications in M3 Business Engine.

- All projects, source files for the layouts, test data and example reports to be used for customer modifications.

Handling Exstream language

1 In the delivery for Standard there is a language file `MvxLangMgrdllU.dll`. Place this file in the folder:

<Exstream_installation_folder> `OpenText\Exstream\16.6\Server\bin`

2 Check the file `SSLang.reg` or `SSLang_x64.reg` to verify that the path to the language files correspond to your installation.

3 Run the file `SSLang.reg` or `SSLang_x64.reg` to create the registry entry.

Creating a runtime environment for Exstream

The layouts for M3 Business Engine Standard have been grouped into main projects. There is also one main project for each market and industry.

Editing a platform file

1 Open the Main Project to edit the file `Platform` in an editor.

2 On TCP/MOVEX, select **Settings > Physical layer Platform (Physical)**.

Set the port to be used for running your application. Set one port for the Standard application and one port for each market, industry, and other application that you want to run.

3 In the Platform file, on the MAIL connector settings, specify your mail server settings.

4 In the Platform file, on DirScan_MOVEX, select Physical layer Platform (Physical).

In the Folder section, specify the destination folder of stream files for printouts.

5 Save the `Platform` file.

Export the main projects

Export the main project according to the instructions in the Exstream documentation.

In the Design Center, perform an export.

In the M3 BE Exstream layouts delivery structure, there are empty export folders that you can use for this. We recommend that you use these folders, for example:

```
Exstream16.x\Export_15.1S\Export_15.1S
```

Create applications

You can either choose to create separate applications for the Main Projects, but this could be difficult for the M3 BE user who might need to know which layout is used by a certain application.

We recommend that you run and deploy all these Standard Main Projects in the same application. However, the projects must be deployed separately in the application and every deploy will delete the file `start.arg` from the previously deployed Main Project.

In the Control Center, create applications for Standard and all markets.

Deploying main projects

Note: Deploy the main project by following the instructions in the Exstream documentation. When selecting the physical layer to deploy, clear the Clear data folder on deploy check box.

Accessing M3 Business Engine

1   Log in to Infor Ming.le as a valid M3 Business Engine user.

2   Select **Infor M3**.

Setting up M3 Business Engine for Exstream layouts

In M3 Business Engine, run the programs in these sections to make the necessary setup for the layouts.

Create servers in MNS218

Define the available servers in MNS218 by specifying the IP address of the server to be used.

Create services in MNS216

Define the services for Exstream.

Note: If separate services are being used for different countries or for standard and customer modified prints, these services must be defined here.

Service definitions apply to all companies and divisions in the environment.

Create output server definition in MNS204

In MNS204, specify the service (from MNS216) that an output should be directed to.

Create output definition in MNS205

In MNS205, define the actual destination the output should be sent to once it reaches the server. This may be the printer name, email address, or fax number.

Check output jobs in MNS270

After MNS204 and MNS205 have been configured and the print has been run, a record should exist in the MNS270 M3 function. This function has limited functionality although it is useful to check:

- Whether the print job has been successful
- The service (IP address and port number) that the output has been sent to
- The date, time, job number, and status of the output
- Any error messages that may be related to the output job

Using StoryTeller layout templates for OpenText Exstream

Storyteller templates are used together with M3 Configurable Output XML. Input format is XML from M3.

Component	Supported version
OpenText Exstream	16.x

Main projects structure

This table shows the main projects that are delivered:

Projects	References
M3BE_Exstream_Layout_XML_Base-16.0.0.YYYYMMDDmmss - referred to as the Base project for the rest of this document	• See Adding a project to Exstream installation on page 26. • See Languages not supported by the Arial font on page 31.
M3BE_Exstream_Layout_XML_Forms-16.0.0.YYYYMMDDmmss - referred to as the Forms project for the rest of this document	
M3BE_Exstream_Layout_XML_Ad-Hoc-16.0.0.YYYYMMDDmmss	See Configuring Ad hoc reporting for OpenText Exstream on page 33.

The main difference between these projects is that in the Base project, language translations are sent in the XML while in the Forms project, the translation tables are installed in Exstream. Another difference between the two is that layouts are more defined in Forms project, and are specific for some country versions.

Downloading StoryTeller packages for M3 BE

1 In Infor Ming.le, open the **Cloud File Transfer Agent Administration** page.

2 Select **Download Contents**.

3 Select all the check boxes for StoryTeller Layouts Base Templates, StoryTeller Layouts Forms Templates, and StoryTeller Ad hoc Templates.

4 Click **Download Contents**. A zip file with the selected content is downloaded to your server.

Adding a project to Exstream installation

1 Open and unpack the zip file M3_BE_16.0.0_XML_Base_Project_Template_vxx.zip.

2 Start **OpenText Communications Builder**.

    a   Click **File > Unpack Project**.

    b   Select the unpacked file M3_BE_16.0.0_XML_StoryTeller_Base_Project_Template_vxx.dc package. Click **OK** in the pop up window.

3 If you did not open the project after extraction, open the InforM3MainProject.dcproject file from the Main Project folder.

4 In Communications Builder, select Platform. Modify the input and output connectors according to your needs.

See Platform - Input/Output connectors on page 27.

5 Save the project and select **create release**. Select the tenant that the project will be deployed to. Log on to it with your credentials.

6 Finalize the release and specify the needed information.

7 Open **Control Center** and create a new application.

For example, the new application can be named M3XML_DEV. The suffix refers to the physical layer deployed to the application. You must have at least one application for each physical layer used.

For detailed instructions on how to create a new application, visit http://www.exstream.com for the Exstream user documentation.

8 Java configuration must be enabled on the application because of the IDM output connector.

    a   Right-click the application in **Control Center**.

    b   Select **Java configuration**.

    c   Click **Value** under Java vendor. From the drop-down, select Oracle or IBM.

9 Right-click the application and select **Deploy Export File**.

10 Deploy the release from CAS according to the OpenText Exstream documentation.

11 Test the application by running a printout from M3 or manually processing an XML file in the Exstream application.

12 Optionally, repeat these steps for the Forms project.

Exstream project structure

The Exstream Base project is divided into different sub-projects, which are all connected to the main project.

- InvoiceProject
- ReportProject
- EQM - Equipment Quotation Management Project
- FIC - Financial Controlling
- FIM - Financial Accounting
- MAI - Maintenance Management
- MAN - Manufacturing Execution
- MNT - Maintenance Order Management
- MSF - Application Foundation
- PJM - Project Management
- PMD - Product Data Management
- PUR - Procurement
- RTM - Rental Management
- SCE - Supply Chain Execution
- SCM - Supply Chain Management
- SCP - Supply Chain Planning
- SLS - Sales Management
- SRV - Service Management

InvoiceProject and ReportProject aim to merge common documents from multiple business areas.

The Exstream Forms project has one sub-project, RU-Russian market.

For the Exstream global platform, there are physical layers: DEV, TEST, and PROD. These layers allow the user to export files to different environments separately, with the option of having different platform and runtime settings for each physical layer.

Platform - Input/Output connectors

In both the Base and Form projects, a number of input and output connectors are delivered. Some of these connectors must be configured in order to work, depending on the need. Configuration is done on the platform.

These input connectors are delivered:

- TCP/IP input

  This standard server address is the location where the application is running. Input ports can be modified as needed.

Physical layer	Base project	Forms project
DEV	20150	20160
TEST	20151	20161
PROD	20152	20162

- Directory scan

  In most cases, the input directory location should be modified.

Physical layer	Base project	Forms project
DEV	.\inputDEV	.\inputDEV
TEST	.\inputTEST	.\inputTEST
PROD	.\inputPROD	.\inputPROD

The same output connectors are used in both the Base project and the Forms project. These are the output connectors delivered:

- MAIL_PDF output

  For this connector, the TCPIP profile GenericMail is used for mail server settings. The profile must be modified with host and port for your mail server. Profile is located in the Global resource set, under the Profiles folder.

  The connector must also be modified with domain name.

- MAIL_HTML output

  This connector has the same configuration as MAIL output.

- ATTACHMENT_PDF

  No configuration required. This is used by MAIL output.

- FILE_PDF output

  This connector does not need to be configured unless used for special functionality.

- PCL output

  This connector does not need to be configured unless it will be used for special functionality. But the user has to update the printer information in M3. This can be done in the **Printer. Open** program in (CRS290) by adding the printer address in the server address field.

- MATRIX output

  This connector has the same setup as PCL, but the user has to update the printer information inM3. This can be done in the **Printer. Open** program in (CRS290) by adding MATRIX in the modification field.

- IDM output

  This connector requires a path to an ION API key. This key can only be generated by a system administrator for a generic user. The key must be placed on the same server as the Exstream application.

Project runtimes

The runtimes in the base project do not need to be configured unless special functionality is required.

Sample files

Sample XML files are included in each sub-project which makes it easy for the user to get started with Exstream configuration.

Global function file

No configuration for global functions is done unless special behavior is required.

StoryTeller fragments

Fragments are a tool to make it easier to maintain documents in StoryTeller. These fragments are general components that can exist in different documents at the same time. One advantage is that they only need to be updated once. In the Base project, some standard fragment documents are provided. A user can add more fragments to these documents or edit existing fragments, based on the need.

XMLIN event tool

The data from M3 is sent in the form of XML data. To collect the data, XMLIN is required. XMLIN provides a mapping for all standard fields. If a user adds a field to the XML in the M3 environment, the field also has to be added to the XMLIN mapping. Patterns are defined to connect the XMLIN event to specific printer files.

XSLT patterns are used to define fields and patterns in the XMLIN tool.

StoryTeller document

The actual output document is in the StoryTeller document.

The main document template design consists of two pages. The first page is required while the succeeding pages are optional. The header data for the document is designed by using various fragments. The part which can grow in size depending on input data is placed into a story. It is then put into a story frame and placed on the page layout.

Most of the components are designed with the help of tables, repeaters, and substitutions. Repeaters and substitutions are controlled by xpath which is used to control and adapt the layout based on the input data.

These examples are some useful xpath expressions:

- `/data/message/Invoice_No[. != ""])`

  This expression, if used in a repeater on a table row, will not display the row unless an invoice number is sent in the xml.

- `string(Line_Amount[. != 0])`

  This expression, if used in a substitution, will not display the field if the value is zero.

- `TextBlock[TextLine]`

  This expression, if used in a repeater on a table, will not create the textblock unless theTextBlock has a TextLine.

Formatting numbers and dates

All numbers and dates are formatted inside StoryTeller.

The standard delivered formatting is done with the help of variables which can be used in the modification field inside StoryTeller. These variables utilize the locale setting of XML to format numbers correctly, depending on the language and the country. The date format, thousand separator, and decimal format are not used in the standard solution. The value of the locale can be accessed using the *$locale* variable.

To format numbers and dates according to locale that uses the modification field, these variables must be set:

- $localeNumCurrFormat

  This formats the number as a currency for the country of the locale.

- $localeNumFormatXdec

  This formats the number with X number of decimals, with X is 0->9-.

- $localeNumFormat

  This variable should always be used in conjunction with the setLocaleNumFormatDecimals(string) function. The function takes the xpath of the current substitution field as an argument, then sets $localeNumFormat to use the correct amount of decimals. This variable is used on numbers without a fixed amount of decimals or values which are not currency amounts.

  Note: Always add the script and the modification when using this variable.

- $localeDateCurrFormat

  This formats the date in the standard short format for the locale.

If more alternatives are required, then add the variables to the setGlobalFormats in the global function file.

For example: $localeNumMediumDateFormat = "date.medium{} (" + $Locale + "){}"; could be added to use medium date format for given locale.

Another alternative is to use the formats provided in StoryTeller. Only use these formats if specific locale settings are required.

Type attributes

Certain blocks in the XMLIN event has type attributes mapped. To select a specific type of block, xpath can be used.

For example, to select payer name, use Block_Address[Address_Type="Payer"]/Address_Name.

Utility functions

Some utility functions, except for formatting numbers, are provided to produce a "zebra rows" effect. Zebra rows are rows in a table with alternating colors. To produce this effect, you can set a table row to the color that you want, and then use the provided functions to set alternating rows to be transparent.

- `initZebraRows`

  This utility function resets zebra row count. The next call to `setZebraRowsHead` makes the line transparent. This action must be called before the table where the zebra rows are to be displayed.

- `setZebraRowsHead`

  This utility function sets the row to a specific color and ensures that the next call to the same function yields a different color. This function must be used on a repeated row.

- `setZebraRows`

  This utility function sets the current row with the same color as the previous `setZebraRowsHead` `row`.

- `initZebraRowsHead`

  This utility function changes the color for the next call to `setZebraRowsHead` or `setZebraRows`. This function works like `initZebraRows`, but does not need be called on a table row.

Example:

For a simple zebra row table, the user can add `initZebraRows` as a script somewhere before the table, and then add the script `setZebraRowsHead` to a row under the repeated row.

Languages not supported by the Arial font

The Arial font is used in the layout templates for both the Forms and Base projects. For languages which use characters not supported by the font, a font substitution table is required.

Applying a font substitution table

1 Open the main project InforM3MainProject.dcproject from the location where you installed the XMLStoryTeller project template.

2 Double-click **GlobalResources** resource set.

3 Right-click the table folder. Click **New > table**. Name the file **fontmap.tbl**, or something similar.

4 Double-click the create table fontmap.tbl. This action opens the **Edit program** where you can define the font substitution table.

You can add these lines to change the Arial font to Tahoma:

Current string	New string
mapfont "Arial"	"Tahoma"
mapfont "Arial_Bold"	"Tahoma_Bold"
mapfont "Arial_Bold_Italic"	"Tahoma_Bold"
mapfont "Arial_Italic_Underline"	"Tahoma_Underline"

For further details, see the Exstream documentation.

5 Save the file and exit.

6 Double-click **pltGlobalPlatform**.

    a On **Platform**, right-click the output connector FILE_PDF and select **Settings**.

    b On **Output Connector Settings - FILE_PDF**, select the platform layer.

    c   Click the green square icon under **Device Driver Settings** to browse to the new font substitution table fontmap.tbl.

    d   On **Browse for Resources**, select the **Global Resources resource set**, then select font.map.bl.

    e   Click **OK**.

7   Repeat the previous step for other output connectors MAIL_PDF, ATTACHMENT_PDF, and IDM. You can also use the steps for PCL, MATRIX, and MAIL_HTML.

8   Click **Save** and close **Platform**.

9   To test the font substitution, export and deploy your application. Proceed with normal testing.

    See

Configuring IDM for OpenText Exstream

Infor Document Management is an application used to archive documents and files.

Connecting Exstream to IDM

To connect Exstream to Infor Document Management, an Application Program Interface or API key to ION is required. This key is generated by a system administrator for an Infor Document Management user in Infor Ming.le. The key should be generated for a generic user with the sole purpose of connecting Exstream to IDM.

After the ION API key is generated, the key must be moved to the Exstream server. Then, configure the IDM connector in OpenText Design Center for the M3 XML StoryTeller base project to point to this key.

Note: The ION API key provides login rights to generic users, so it must be stored in a secure location on the Exstream server.

Configuring M3 BE for IDM connection

The M3 Business Engine output is delivered with predefined connections to the delivered document types. Changes to the document types or connections must be done in the XML Structure.

1   Open the program in CMS006.

2   From the F-panel, connect a printer file to a document type.

3   From the G-panel, configure the document attributes.

Verifying the configuration

In these steps, the Manual Invoice ARS121PF is used as an example. ARS121PF can be replaced with other supported documents for this test. By default, this printout is connected to the document type M3_Customer_Invoice.

1   Start 'Output Server. Open' (MNS218). From the program, create a record and define the Exstream server by specifying the IP address of the server to be used.

2   Start 'Output Service Type. Open' (MNS217). From the program, create a record and define the interface as XML.

3   Start 'Output Service. Open' (MNS216). From the program, create a record. Specify the created server, the created service type, and the port of the server.

4   Start 'Output Service Selection. Open' (MNS204). From the program, create a record. Specify your user name, the printer file as ARS121PF, and the previously created service.

5   Start 'Output Media Selection. Open' (MNS205). From the program, create a record. Specify the printer file as ARS121PF and set the media to *ARCHIVE.

    In this example, printout ARS121PF is used and it can be reprinted from 'Customer Invoice. Display Manual' (ARS122) using Related option 6.

6   Open **Infor Document Management** to verify if the printout is added.

7   Start 'Output. Manage per Job' (MNS270). From the program, ensure that the output job is saved by verifying if the output job has the SAV status.

8   Check the OpenText Control Center log for your Exstream application to determine if any error occurred.

9   Start 'XML Structure. Open' (ARS205). Select an invoice from the list. After the invoice is selected, it is displayed in the Related Information pane.

Configuring Ad hoc reporting for OpenText Exstream

M3 Super Users and IT support personnel can create and maintain flexible user defined report layouts based on the M3 Business Engine database using the M3 Ad hoc reporting tool. This tool enables you to generate reports in XML format to be processed by applicable software.

M3 Ad hoc reporting tool has several benefits:

*   It enables the creation of operational reports and listings direct from the M3 BE application database.
*   It helps reduce costly and time-consuming modifications.
*   It increases business agility and speed by delivering ad hoc reports on demand.
*   It recognizes M3 and adheres to the user security privileges.

Designing custom lists in the M3 Ad hoc Reporting designer

In the M3 Ad hoc reporting designer, you can design custom lists by adding virtual fields and checking the preview of the result before you generate the report.

Ad hoc project for OpenText Exstream

Ad hoc project for OpenText Exstream is an optional tool that can produce Excel files in an XML file format, and PDF files.

Generic features

All 30 columns from Ad hoc XML are included in PDF and Excel files.

The parameter File type in AHS110/E and AHS150/E determines if the file to be created is a PDF file, an Excel file, or both file types.

Arial and Arial Unicode MS can be used. If Arial Unicode MS is not installed, Arial will be used.

Specific features for PDF file

In the Ad hoc XML, all values that are generated in pdf will have the same format.

Specific features for Excel file

- Numerical fields

  Values with decimal separator "," in the Ad hoc XML will be automatically converted to correct Excel format.

- Negative values

  Negative values with format "NNNN-" are automatically converted to format "-NNNN".

- No 1000 separator

  Values will not have the 1000 separator activated in the Excel file.

- Decimals

  Values will use the number of decimals defined for the column.

- Date fields

Enabling Ad hoc report layouts through Exstream

Run these programs to set up the M3 Business Engine to use the Ad hoc report layouts.

MNS218 - Create servers

Define the server in MNS218 by specifying the IP address of the server to be used.

MNS217 - Create output service type

Create an output service type with these values:

Field	Value
Service type	M3ADHOC
Description	M3 Ad Hoc
Interface	STREAM

Create M3ADHOC service ID in MNS216

Field	Value
Service ID	M3ADHOC
Description	M3 Ad Hoc Reports
Service type	M3ADHOC (same as for MNS217)
Server ID	Server name for Exstream server (as configured in MNS218)
Port	Port no as defined in the M3AdHoc application (default: 9000).

Create M3ADHOC location in AHS010

Field	Value
Description	M3 Ad Hoc Reports
Name	M3 Ad Hoc
File name meth	Select a value, for example 05-Report + Job number
File name	Optional, depending on the selected method
Service ID	M3ADHOC (the service ID configured in MNS216)

Specifying a document location in AHS020

A document location can be used by Exstream to save the PDF and Excel file in a folder on the Exstream server. The document location created in M3 Ad Hoc Documentation Location. Open (AHS020) can be used in Ad Hoc Report. Open (AHS110) and Ad Hoc Report. Print (AHS150).

To create a document location:

1   Open M3 Ad Hoc Documentation Location. Open (AHS020).
2   Specify a value for Doc location, and click the icon **Create**. For example:

Doc location

<userid>

3   In AHS020/E, specify this information:

Description

Provide a description, for example, Ad Hoc Location.

Name

Provide a name, for example, Ad Hoc Location.

Doc Loc method

Select 2 - Location path + Current user

Doc loc path

Specify a folder name, for example ADHOCLOCATION.

The documentation location path folder is created in the folder structure, for example: `\C:\Man agementGateway\5.6.0\root\applications\M3AdHoc\Physical\ADHOCLOCATION`

Create Ad Hoc Report version in AHS110

Use the Ad Hoc Report Location created in AHS010 when you create a Ad Hoc Report version in AHS110.

Field	Value
Report Location	M3ADHOC

Create mail records in CRS111

Report sender

Specify the email address of the sender of the reports.

Field	Value
Type	19
E-mail key	*name of the sender*
E-mail addr	*email address for the sender*

Report receiver

Specify the email address of the receiver of the reports.

Field	Value
Type	04
E-mail key	*user ID for the receiver*
E-mail addr	*email address for the receiver*

Using OpenText Cloud

M3 customers can license OpenText Cloud for M3, a third-party Output Management Solutions, and use all capabilities with the configurable XML. To connect to OpenText Cloud (OTC), XML files are delivered out of M3 CE through an HTTPS communication.
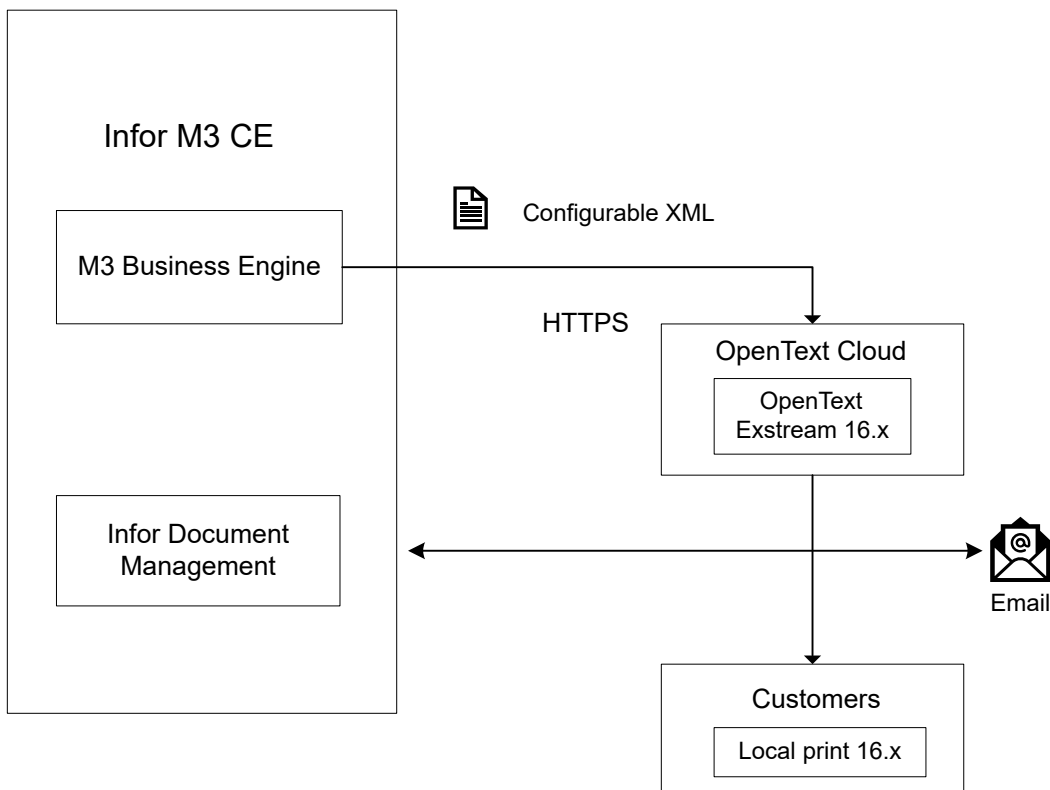
Configuring OpenText Cloud for M3

1 Generate metadata for all printer functions that support configurable XMLs in program 'List and Printer programs. Configure' (CMS005). Select 3–'Printer files' and use F14='Gen standard'. This will create the standard configurations. From (CMS005), you can access many of the configuration functions through related options.

2 Define a service type for configurable XML by creating an output service type in program 'Output Service Type. Open' (MNS217) with interface equal to XML. If a Streamfile application is used, a service type for that application must also be created in (MNS217).

3 Define a service for configurable XML by creating a service identity in program 'Output Service. Open' (MNS216). On the E-panel, select an output service type with interface equal to 'XML' and select communication type equal to 5-'OpenText Cloud'. The user, password, and path should be provided by OpenText.

4   Connect the service identity for XML or Stream file to the Printer file in program 'Output Service Selection. Open' (MNS204).

5   Review the output medium defined in program 'Output Media Selection. Open' (MNS205). Existing output media *PRT = Printer, *FILE = File, *MAIL = Email are supported. To send the output to Infor Document Archive, Output medium must be *ARCHIVE = Archive be defined.

6   Optionally, if program 'Std Document. Connect Media Ctrl Object' (CRS949) is used for a specific printer file, the interface defined in program 'Document Media. Open' (CRS116) is connected to the standard document in programs, 'Document. Connect Media' (CRS929) and 'Doc Media Control Object. Connect Media' (CRS949).

7   Define the printer address in program 'Printer. Open' (CRS290). Only needed if output medium *PRT = Printer is used in program 'Output Media Selection. Open' (MNS205).

8   Define the ISO language code for all languages in program 'Language. Open' (CRS010). The ISO language code is used when setting the **Local** field in the XML file.

9   Run the printer function as normal, the XML file (or Stream file) can be viewed through related option 11-'View' in program 'Output. Manage per Job' (MNS270).

Using Project templates for OpenText Cloud

OpenText manages its own cloud. All development and configurations must be done in close cooperation with OpenText environment administrators.

Component	Supported version
OpenText Cloud	16.x

Main projects structure

The Projects can be downloaded from Open Texts User Acceptance Test (UAT) tenant. Ensure that the latest version is downloaded. Updates will also be available in this tenant.

This table shows the main projects that are delivered:

Projects	Description
InforM3MainProject_version_XX	Main project for Storyteller layouts with XML input
InforM3MainProjectForms_version_XX	Main project for Storyteller layouts for Russian Forms, uses XML input
M3AdHoc_version_XX	M3 Ad hoc reports project

The main difference between these projects is that in the Base project, language translations are sent in the XML while in the Forms project, the translation tables are installed in Exstream. Another difference between the two is that layouts are more defined in Forms project, and are specific for the Russian market.

Downloading Project packages for M3 BE

1 In Communications Builder, log in to Open Text Clouds Management Gateway and specify the tenant name, user name, and password.

2 Select **Open Project**, select the connection above, and then select the correct project and the latest commit.

3 When the project has been opened in the Communications Builder, log out of the current Management Gateway connection.

4 Log on to your own Tenant connection and select **Create release**.

5 Deploy the release to the correct application.

See configuration steps in <u>Adding projects to Exstream installation</u> on page 39.

Knowledge prerequisites

Enabling the XML StoryTeller Base Project in Exstream requires these knowledge and experiences:

- Passed the training for OpenText Exstream version 16, XMLIN, and StoryTeller. The training is applicable if the mentioned Exstream project is used. Knowledge in StreamIn and PageOut is necessary if development is done in the PageOut project.
- System administrator experience in the Windows platform is an advantage.

Creation and access to environment

The administrators of OpenText Cloud creates tenants, application domains and applications for the environment. Access details are given to the developers and administrators on the customers' side.

Developers can only access the web-based Management Gateway where deployment of releases are done. Log files are also accessible from that web page.

The Communications Builder must be installed on the PC or server that creates the release for deployment on the server.

Adding projects to Exstream installation

1 Extract the downloaded projects individually. Open the projects and create a release for each project. .

2 In the Communications Builder, edit the platforms according to the specifications from Open Text administrators. The HTTP connector and password files must be configured for all layers.

See <u>Platform - Input/output connectors</u> on page 41.

Mail server settings must be specified for the email connectors. Configuration of the IDM connector is optional, see <u>Connecting Exstream to IDM</u> on page 42.

3 The XML_Storyteller_Base_Project can be opened and exported as one release. This release should be deployed into the XML/Storyteller application as one single release.

4   To deploy a release in the Management Gateway, use the web link supplied by OpenText. Log in to the Tenant, select the Application Domain and click the **Deploy** icon for the Application. After that, you can select the Project, Release and physical layer.

Note: The PageOut project is divided into sub-projects. For all sub-projects, they must be deployed using the script Labelsubdeploy:

```
Labelsubdeploy ^<projectname^> ^<appname^> ^<physplatform^> ^<tenant
name^> ^<tenantadmin^> ^<tenantpwd^> ^<ReleaseName^>
```

The script must be used because of the option -keep-data, in the script. It is equivalent to clearing the check box **Clear data on deploy**. This option does not exist in the current version when deploying in the web-based Management Gateway.

5   Test a printout from M3 and ensure that everything works as it should. Verify through the log if something is not working.

Exstream Cloud projects structure

The Exstream Base project is divided into different sub-projects, which are all connected to the main project.

- InvoiceProject
- ReportProject
- EQM - Equipment Quotation Management Project
- FIC - Financial Controlling
- FIM - Financial Accounting
- MAI - Maintenance Management
- MAN - Manufacturing Execution
- MNT - Maintenance Order Management
- MSF - Application Foundation
- PJM - Project Management
- PMD - Product Data Management
- PUR - Procurement
- RTM - Rental Management
- SCE - Supply Chain Execution
- SCM - Supply Chain Management
- SCP - Supply Chain Planning
- SLS - Sales Management
- SRV - Service Management

InvoiceProject and ReportProject aim to merge common documents from multiple business areas.

For the Exstream global platform, there are physical layers: DEV, TEST, and PROD. These layers allow the user to export files to different environments separately, with different communication settings depending on each physical layer.

Platform - Input/output connectors

In all projects, a number of input and output connectors are delivered. Some of these connectors must be configured in order to work, depending on the need. Configuration is done on the platform.

The files are delivered to Exstream using HTTP-communication, therefore all projects have HTTP input connectors. The other input connectors are currently not used in the cloud environment.

These settings are required for the HTTP connectors for all layers. The details should be supplied by Open Text Cloud Administrators:

*   Connector settings = *Port*
*   HTTP access = *URI*
*   HTTP Realms = check the *Realm*; *URI* and *Password File* (see below) for the layer.

Resources - Password Files: Ensure that the Password Files have correct settings for all layers.

These standard output connectors are delivered and may differ depending on the project:

*   MAIL output

    For this connector, the TCPIP profile MailServer profile must be modified with your mail server settings. The profile is located in the Global resource set.

*   FILE output

    No configuration required unless used for a special functionality.

*   PCL output

    No configuration required unless used for a special functionality. Although, you have to update the printer information in M3. In M3, you have to add the printer address in the **Prt srv address** field in (CRS290).

*   MATRIX output

    This connector has the same setup as PCL but the user also must add MATRIX in the **Modification** field in (CRS290).

*   IDM output

    This connector requires a path to an ION API key. This key can only be generated by a system administrator for a generic user. The key must be placed on the same server as the Exstream application.

Note: The output connectors may need to be configured for delivering output to Local Delivery. See

Configuring IDM for OpenText Exstream Cloud

Infor Document Management is an application used to archive documents and files.

This part of the documentation is only valid for the XML/Storyteller projects that use the IDM connector.

Connecting Exstream to IDM

To connect Exstream to Infor Document Management, an Application Program Interface or API key to ION is required. This key is generated by a system administrator for an Infor Document Management user in Infor Ming.le. The key should be generated for a generic user with the sole purpose of connecting Exstream to IDM.

After the ION API key is generated, the key must be moved to the Exstream server. A suggestion is to add the files to the global resource set, one file per layer or environment. The ION API file will then be exported to the working directory of the application. The location of the file must be specified as a setting in the IDM connector, for example:

ION API key location = ..\data\IDMExstreamAdapter.ionapi

Note: The ION API key provides login rights to generic users, so it must be handled securely.

Configuring M3 BE for IDM connection

The M3 Business Engine output is delivered with predefined connections to the delivered document types. Changes to the document types or connections must be done in the XML Structure.

1 Open 'XML Structure. Open' (CMS006).
2 From the F-panel, connect a printer file to a document type.
3 From the G-panel, configure the document attributes.

Local Delivery

To be able to deliver printouts, files and other resources to servers, printers, and resources on the client's network, an application called Local Delivery must be installed.

The application must be installed on a local server on the client's side for the data to be pulled from the Exstream application.

Connector configuration

For each layer on the connector, three options are available:

• Communications server only

  Output will be handled in the cloud.

• Local Delivery only

  Output will only be delivered to the Local delivery application on the clients' side.

• Communications server and Local Delivery

To be able to get printouts directly to local printers, all connectors used for printing must be set up to use Local Delivery.

Configuration files

These files must be configured. The details are supplied by Open Text Administrators:

- {working directory}\packages\localdelivery.xml
- {working directory}\securityprofiles\rest-documents.xml
- {working directory}\securityprofiles\rest-otds.xml

Configuring M3 user management in Infor Ming.le

The data flow between M3 and Infor Ming.le for synchronizing users, roles, and permissions is automatically activated during the provisioning of the tenant.

If the data flow is missing or must be replaced, you can download and import the data flow manually.

Verifying the connection points for M3 and Infor Ming.le

If the connection point for M3 is not named `m3` or if the connection point for Infor Ming.le is not named `instance01`, you must update the connection points and the data flow manually.

In ION Desk, select **Connect > Connection Points**.

a   On the **Connection Points** page, select the Infor Ming.le connection point and verify that the **Documents** tab includes all the inbound documents that are required for this integration:

Document	Receive in Application	Send from Application
Sync.AccountingEntity	selected	
Sync.Person	selected	
Sync.SecurityRoleMaster	selected	
Sync.SecurityUserMaster		selected

Your connection point can also include documents for other integrations.

b   On the **Connection Points** page, select the M3 connection point and verify that the **Documents** tab includes all the outbound documents that are required for this integration:

Document	Receive in Application	Send from Application
Sync.AccountingEntity		selected
Sync.Person		selected
Sync.SecurityRoleMaster		selected

Document	Receive in Application	Send from Application
Sync.SecurityUserMaster	selected	

Your connection point also includes documents for other integrations.

Verifying the data flow between M3 and Infor Ming.le

1 In ION Desk, select **Connect > Data Flows**.
2 On the **Data Flows** page, click the M3-Mingle_security_BODs_integration data flow.
   If the data flow is missing, follow the steps in Downloading the data flow for M3 and Infor Ming.le on page 44 and Importing the data flow for M3 and Infor Ming.le on page 44 before completing the next step.
3 Click the document icons in the flow and verify that they contain the expected documents.
   Verifying the connection points for M3 and Infor Ming.le on page 43.

Downloading the data flow for M3 and Infor Ming.le

Normally, this data flow is activated during provisioning. This task is only required if the data flow is missing or must be replaced.
Note: To perform this task, you must have the `M3BE-FndAdmin IFS` security role.

To download the data flow:

1 In Infor M3, select **Administration Tools > Cloud File Transfer Agent Administration**.
2 Click **Download Contents**.
3 Expand **M3 ION Content**, select the **M3-Mingle_security_BODs_integration** check box and click **Download**.
4 Extract `M3-Mingle_security_BODs_integration.xml` from the .zip archive.

Importing the data flow for M3 and Infor Ming.le

Normally, this data flow is activated during provisioning. This task is only required if the data flow is missing or must be replaced.
1 In ION Desk, select **Connect > Data Flows**.
2 Click **Import**.
3 Browse and select `M3-Mingle_security_BODs_integration.xml`.
4 Click **OK**. The connections, data flows, and mappings are imported from the file.

5 This import is expected to contain duplicate items for the connection points, because the m3 and Infor Ming.le connection points are created during the provisioning. Select **Merge additional documents** and click **OK**.

The Infor Ming.le connection point is referenced as *instance01* in the data flow. If the data flow is imported without the option to merge with an existing connection point, check the list of existing connection points. If applicable, replace the reference to the imported connection point with a reference to an existing connection point for the application.

Configuring BOD-specific settings

The settings are only applicable for a specific BOD or BODs that a customer selects to use.

Configuring settings for customer master data BODs

Complete these steps to receive and process BODs corresponding to these agreements:
- M3_CustomerPartyMaster_Process_In
- M3_ShipToPartyMaster_Process_In
- M3_BillToPartyMaster_Process_In
- M3_PayFromPartyMaster_Process_In

1 In `'Customer. Open' (CRS610),` create the customer **IONCUST** and specify this information:

Customer Type
Specify **0**.

2 Specify input in all required fields and ensure that the status is set to **20**.

Note: Do not add an Invoice recipient (INRC) or Payer (PYNO) on the `(CRS610/J)` panel.

Configuring settings for ProcessContactMaster

Complete these steps to receive and process BODs corresponding to M3_ContactMaster_Process_In.

1 Select **Administration Tools > Enterprise Collaborator Administration**. On the **Server** tab, select **Agreement Control Properties** and locate M3_ContactMaster_Process_In.

2 Verify these control properties:
- **m3beContactPrefix**: This optional control property can be used to set the prefix for naming a new contact that has been created with ProcessContactMaster.
- **m3beContactSequenceDigits**: This optional control property can be used to set the number of digits for the sequence following the prefix when naming a new contact that has been created using ProcessContactMaster. The default value is 3 if this value is not provided.

If m3beContactPrefix is set to **CONTACT** and m3beContactSequenceDigits is set to **3**, the input to the MI program for field contact will be **CONTACT???** where ??? auto-generates a sequence with 3 digits.

If the input BOD xml contains an ID for the contact, it is used as input in the MI program.

3 Click **Save**.

Configuring settings for ProcessQuote

Complete these steps to receive and process M3_Quote_Process_In.

1 In `'CO Type. Open' (OIS010),` create an order type with order category **4**.

2 Select **Administration Tools > Enterprise Collaborator Administration**. On the **Server** tab, select **Agreement Control PropertiesM3_Quote_Process_In**.

3 Verify the **m3beDefaultQuoteOrderType** control property.

This optional control property can be used to set the default order type when creating new sales quotations using ProcessQuote.

4 Click **Save**.

Configuring settings for inbound supplier invoice BODs

Complete these steps to receive and process BODs corresponding to *M3_SupplierInvoice_Load_In* and *M3_SupplierInvoice_Process_In*.

1 Select **Administration Tools > Enterprise Collaborator Administration**. On the **Server** tab, select **Agreement Control Properties** and locate **M3_SupplierInvoice_[Load][Process]_In**.

2 Ensure that the **m3beInvoiceValidation** control property has one of these values. It defines whether the invoice validation routine is started automatically or manually in Infor M3 CE Core.

Value	Description
*YES	The validation routine is started automatically immediately after the invoice has been added in Infor M3 CE Core.
*NO	The validation routine must be started manually in Infor M3 CE Core.

3 Click **Save**.

Configuring settings for inbound warehouse management BODs

These settings are only valid for the corresponding BODs:

- M3_InventoryAdjustment_Process_In
- M3_InventoryAdjustment_Sync_In
- M3_InventoryHold_Sync_In
- M3_ReceiveDelivery_Process_In
- M3_ReceiveDelivery_Sync_In
- M3_ServiceConsumption_Sync_In
- M3_Shipment_Process_In
- M3_Shipment_Sync_In

Complete these steps to receive and process these partner agreements:

1 In `'M3 Stock Msg Partner. Open' (MMS865),` create a message partner specifying this information:

Whs
Leave blank.

Msg
Specify `I`.

Partner
Specify the application ID of the system that sends the BOD, for example, `WM`.

Msg type
Specify `BOD`.

2 Select **Options > Create**.

3 On (MMS865/E), specify a valid M3 user in the parameter **300 Partner manager**.

4 Select **Administration Tools > Enterprise Collaborator Administration**. On the **Server** tab, select **Agreement Control Properties** and locate the specific M3_*Noun_Verb*_In BODs.

5 Verify that the **M3beWarehouseInterfaceProcessFlag** control property has one of these values:

Value	Description
*AUT	Message is processed in an asynchronous mode through a batch job.
	This setting is recommended for high volume environments. Users can find, correct and re-run failed messages in `MHS850` and can also use the mailbox functionality in M3 Business Engine to be notified when a message fails.
	Note: Error messages are not returned by the API.
*EXE	Message is processed interactively in M3 Business Engine.

Note: Control properties enable messages to be processed online or through batch jobs. Leaving the control properties value in M3 Business Engine empty means that the user must manually run the messages through the `21` option in (MHS850)/(MMS850).

6 Click **Save**.

Configuring settings for LoadAdvanceShipNotice

Follow these steps to receive and process M3_AdvanceShipNotice_Load_In.

1   In `'M3 Stock Msg Partner.Open' (MMS865)`, create a message partner and specify this information:

Whs
Leave blank.

Msg
Specify `I`.

Partner
Specify the name of the system that sends the LoadAdvanceShipNotice. Currently, only `SW` is valid.

Msg type
Specify `BOD`.

2   Select **Options > Create**.

3   In (MMS865/E), specify a valid M3 user in the parameter **300 Partner manager** and verify these values:

Msg direction
Specify `I-Input`.

Partner
Specify `SW`.

Msg type
Specify `BOD`.

4   Select **Administration Tools > Enterprise Collaborator Administration**. On the **Server** tab, select **Agreement Control Properties** and locate *M3_AdvanceShipNotice_Load_In*.

5   Ensure that the `M3beWarehouseInterfaceProcessFlag` control property has one of these values:

Value	Description
`*AUT`	AdvanceShipNotice is created in an asynchronous mode using a batch job.
	This setting is recommended for high volume environments. Users can find, correct, and re-run failed messages in (MHS850). They can also use the mailbox functionality in Infor Infor M3 Business Engine to be notified when a message fails.
	Note: Error messages are not returned by the API.
`*EXE`	MHS850MI is run and the AdvanceShipNotice is created in M3.

Note: Control properties enable messages to be processed online or through batch jobs in Infor Infor M3 Business Engine. Leaving the control properties value blank means that the user must manually run the messages through the `21` option in (MHS850).

6   Click **Save**.

Proceed as follow to fulfill the prerequisite for the Address ID.

7   In Infor M3 Business Engine, open `'Supplier. Connect Address' (CRS622)`.

8   Browse through the Supplier field and select the data that is used in the transaction.

9   In `'Supplier. Connect Address' (CRS622)`, specify this information:

Address type
03

Address number
PICKUP

Valid from date
Select a date.

10   Click **Create**.

11   Input all mandatory fields on (CRS622/E).

12   Press **Enter**.

Configuring settings for ProcessItemMaster

Complete these steps to receive and process M3_ItemMaster_Process_In.

1   In Infor M3 Business Engine, create an item type in `'Item Type. Open' (CRS040)`.

2   Create a Template Item (MMS001/MMS002/MMS003) and an Item Numbering rule (MWS050 and MWS051) with valid data. The item type should be set on (CRS040/E) with the status `10`.

Note: The **Item interface controlled** field on (CRS040/E) must be enabled.

3   In `'M3 Stock Msg Partner. Open' (MMS865)`, specify this information:

Whs
Leave blank.

Msg
Specify `I`.

Partner
The value should be the Infor application ID, specified in capital letters, which is retrieved from the logical ID.

Example: If the logical ID is `lid://infor.plmprocess.server123`, the application ID is plmprocess in capital letters (PLMPROCESS).

Msg type
Specify `BOD`.

4 Select **Options > Create**.

5 On (MMS865/E), specify this information:

Partner manager
Specify M3.

Default Item type
Specify the item type that controls which data is the default for each item and how the item numbering is performed. user who is managing the partner settings for this record.

6 Select **Administration Tools > Enterprise Collaborator Administration**. On the **Server** tab, select **Agreement Control Properties** and locate **M3_ItemMaster_Process_In**.

7 Ensure that the **M3beItemInterfaceProcessFlag** user who is managing the partner settings control property has one of these values:

Value	Description
blank	Data is only added. Validation and import to M3 is performed manually.
*VAL	Data is added and validated. Import to M3 is performed manually.
*IMP	Data is validated and, if there are no validation errors, imported to M3.
*AUT	Same as *IMP, but processing is performed by auto job MHS250.

This property enables messages to be processed online or through batch jobs in M3 Business Engine.

8 The optional **M3beLotControlMethod** control property can be used to set the Infor M3 Business Engine field Lot control method (MITMAS.MMINDI) when creating new items using ProcessItemMaster. The control property is only considered if the incoming ProcessMaster BOD has either the element <SerialControlIndicator> or the element <LotControlIndicator> set to `true`. Check the status of the **M3beLotControlMethod** control property.

  a If the control property is missing, the Infor M3 Business Engine field Lot control method is set according to these rules:

  • If <SerialControlIndicator> element is `true`, then Infor M3 Business Engine Lot control is set to **2**.

  • If <SerialControlIndicator> element is `false` or missing, and <LotControlIndicator> element is `true`, then Infor M3 Business Engine Lot control is set to **3**.

  b If the control property is set, the Infor M3 Business Engine **Lot control method** field is set according to these rules:

  • If **M3beLotControlMethod** is **1** or **3** and the <LotControlIndicator> element is `true` AND <SerialControlIndicator> is set to `false` or missing, then Infor M3 Business Engine lot control is set to **1** or **3** respectively.

  • If **M3beLotControlMethod** is **2** or **5** and <SerialControlIndicator> is set to `true`, then Infor M3 Business Engine lot control is set to **2** or **5** respectively.

9 Click **Save**.

Configuring settings for M3 EDI BODs

See KB 2023035 for instructions on how to configure settings for M3 EDI BODs.

Enabling M3 Financial Business Messages (M3 FBM) per country

This section is only applicable if your environment includes M3 Country Versions that use M3 FBMs.

You must import the custom noun data and data flows and verify the content in ION Desk to enable sending and receiving M3-specific financial business messages per country. The content is installed per country.

For certain M3 divisions, FBMs for several countries can be used. For example, the FBMs for Austria can be used for a German division. In this case, the secondary M3 country version must also be licensed.

Note: An M3 Country Version can contain both M3 Financial Business Messages and Infor Localization Services GEMS messages.

See the *M3 Country Version for [country]* for the full set of supported financial business messages for the country.

See KB 2111332 for details of the release month - the first and last internal validation performed - and the external validation, that is, the validation at the bank or authority by the customer.

Downloading the data flows for M3 FBMs

To download the data flow:

Note: To perform this task, you must have the M3BE-FndAdmin IFS security role.

1 In Infor M3, select **Administration Tools > Cloud File Transfer Agent Administration**.
2 Click **Download Contents**.
3 Select the applicable **M3FBM_ION_Integration_to_Country_<Country>** check boxes where *<Country>* is represented by a two-letter abbreviation such as AT, DE, or FR, and click **Download**.
4 Extract `M3FBM_<Country>_CustomNounData.zip` and `M3FBM_<Country>_DocumentFlows. zip` from `M3FBM_ION_Integration_to_Country_<Country>.zip`.
5 Extract the `M3FBM_<Country>_<Direction>.xml` data flow where the direction is either Inbound or Outbound from `M3FBM_<Country>_DocumentFlows.zip`. Examples: `M3FBM_AT_Outbound` or `M3FBM_FR_Inbound`.

Importing the custom noun data for M3 Financial Business Messages

1 In ION Desk, select **Data Catalog > Object Schemas**.

2 Click **Import**.

3 Browse and select `M3FBM_<Country>_CustomNounData.zip`.

4 Click **OK**. The custom noun data is imported from the file.

Importing the data flows for M3 Financial Business Messages

1 In ION Desk, select **Connect > Data Flows**.

2 Click **Import**.

3 Browse and select `M3FBM_<Country>_<Direction>.xml`.

4 Click **OK**. The connections, data flows, and file templates are imported from the file.

 This import is expected to contain duplicate items for the connection points, because the M3 connection point was created during provisioning.

5 Select **Merge additional documents**.

6 This import may also contain duplicate items for the data flow, because the file templates are already imported previously. Select **Skip already existing**.

7 Click **OK**.

8 Repeat steps 2 to 7 for additional data flows as applicable.

The M3 FBM file connection point is referenced as **M3FBM_FILE_CONN_POINT** in the data flow. Check the list of existing connection points if a file connection point already exists for M3 FBM with a different name. If applicable, replace the reference to an imported file connection point with a reference to an existing file connection point for M3 FBM.

Verifying the connection points and data flow for M3 and M3 FBM

If the connection point for M3 is not named m3 or if the connection point for M3 FBM is not named M3FBM_FILE_CONN_POINT, you must update the existing connection points with the applicable documents and reference the existing connection points from the data flow.

1 In ION Desk, select **Connect > Connection Points**.

 a On the **Connection Points** page, select the M3 FBM connection point (M3FBM_FILE_CONN_POINT) and verify that the **Documents** tab includes all the inbound and outbound documents that are required for this integration:

 b Select the desired document and set up the appropriate scenario (**Write a file to a folder** or **Read a file from a folder**) as instructed in the table below.

Document	Write a file to a folder	Read a file from a folder
Sync.M3FBM_AR_Out_VATReport_DigitalVATBook	selected	
Sync.M3FBM_AT_Out_CustomerOrderInvoice_ebInterface	selected	
Sync.M3FBM_AT_Out_VATReport_UVA_U30	selected	
Sync.M3FBM_AT_Out_VATReport_UVA_U30_2021	selected	
Sync.M3FBM_BE_Out_AnnualVAT-Report_LK7	selected	
Sync.M3FBM_BE_Out_AnnualVATReport_LK9	selected	
Sync.M3FBM_BG_Out_Electronic_VAT_Purchase-Books_POKUPKI	selected	
Sync.M3FBM_BG_Out_Electronic_VAT_Sales-Books_PRODAGBI	selected	
Load.M3FBM_BR_In_BankConfirmation_BDB		selected
Load.M3FBM_BR_In_BankConfirmation_BRADESCO		selected
Load.M3FBM_BR_In_BankConfirmation_HSBC		selected
Load.M3FBM_BR_In_BankConfirmation_ITAU		selected
Load.M3FBM_BR_In_BankConfirmation_Santander		selected
Load.M3FBM_BR_In_COInvoice_NFe2_CancelReturn		selected
Load.M3FBM_BR_In_COInvoice_NFe2_IssueReturn		selected
Load.M3FBM_BR_In_COInvoice_NFe4_Approved_IssueReturn		selected
Load.M3FBM_BR_In_COInvoice_NFe4_Unapproved_IssueReturn		selected
Load.M3FBM_BR_In_COInvoice_Synchro31_CancelReturn		selected
Load.M3FBM_BR_In_COInvoice_Synchro31_IssueReturn		selected
Load.M3FBM_BR_In_ItemComplementFCI		selected
Sync.M3FBM_BR_Out_BankRemittance_BDB	selected	
Sync.M3FBM_BR_Out_BankRemittance_BRADESCO	selected	
Sync.M3FBM_BR_Out_BankRemittance_HSBC	selected	
Sync.M3FBM_BR_Out_BankRemittance_ITAU	selected	

Document	Write a file to a folder	Read a file from a folder
Sync.M3FBM_BR_Out_BankRemittance_Santander	selected	
Sync.M3FBM_BR_Out_COInvoice_NFe2_CancelRequest	selected	
Sync.M3FBM_BR_Out_COInvoice_NFeXML4_CancelRequest	selected	
Sync.M3FBM_BR_Out_COInvoice_NFe4_Mastersaf_IssueRequest	selected	
Sync.M3FBM_BR_Out_COInvoice_NFe4_2019_Mastersaf_IssueRequest	selected	
Sync.M3FBM_BR_Out_COInvoice_NFe4_2021_Mastersaf_IssueRequest	selected	
Sync.M3FBM_BR_Out_COInvoice_NFe4_Standard_IssueRequest	selected	
Sync.M3FBM_BR_Out_COInvoice_NFe4_2019_Standard_IssueRequest	selected	
Sync.M3FBM_BR_Out_COInvoice_NFe4_2021_Standard_IssueRequest	selected	
Sync.M3FBM_BR_Out_COInvoice_Synchro31_CancelRequest	selected	
Sync.M3FBM_BR_Out_COInvoice_Synchro4_IssueRequest	selected	
Sync.M3FBM_BR_Out_ItemFCIDeclaration	selected	
Sync.M3FBM_BR_Out_SupplierPayment_BDB	selected	
Sync.M3FBM_BR_Out_SupplierPayment_HSBC	selected	
Sync.M3FBM_CH_Out_SupplierPayment_SEPA_113_CH02_CreditSuisse	selected	
Sync.M3FBM_CL_Out_VATReport_RegistrodeCompras	selected	
Load.M3FBM_CN_In_GoldenTaxInvoice_IssuedInvoiceList		selected
Sync.M3FBM_CN_Out_GoldenTaxInvoice_RejectedInvoiceList	selected	
Sync.M3FBM_CN_Out_GoldenTaxInvoice_SalesInvoiceList	selected	
Sync.M3FBM_CZ_Out_VAT_Report_Control_Statement_XML_2019	selected	

Document	Write a file to a folder	Read a file from a folder
Sync.M3FBM_CZ_Out_VAT_Return_Statement_DPHDP3	selected	
Sync.M3FBM_DE_Out_VATReport_Elster_UStVA	selected	
Sync.M3FBM_DE_Out_VATReport_Elster_UStVA_2021	selected	
Sync.M3FBM_DE_Out_VATReport_UVA18	selected	
Load.M3FBM_ES_In_OnlineVATReport_SII_RespuestaEmitidas_1_1Bis		selected
Load.M3FBM_ES_In_OnlineVATReport_SII_RespuestaEmitidas_2018		selected
Load.M3FBM_ES_In_OnlineVATReport_SII_RespuestaRecibidas_1_1Bis		selected
Load.M3FBM_ES_In_OnlineVATReport_SII_RespuestaRecibidas_2018		selected
Sync.M3FBM_ES_Out_CustomerOrderInvoice_Facturae_321	selected	
Sync.M3FBM_ES_Out_OnlineVATReport_SII_FacturasEmitidas_1_1Bis	selected	
Sync.M3FBM_ES_Out_OnlineVATReport_SII_FacturasEmitidas_2018	selected	
Sync.M3FBM_ES_Out_OnlineVATReport_SII_FacturasRecibidas_1_1Bis	selected	
Sync.M3FBM_ES_Out_OnlineVATReport_SII_FacturasRecibidas_2018	selected	
Sync.M3FBM_ES_Out_VATReport_Format340	selected	
Sync.M3FBM_ES_Out_VATReport_Format347_2015	selected	
Load.M3FBM_FR_In_ExternalAccountStatement_ETEBAC		selected
Sync.M3FBM_HR_Out_Electronic_VAT_PurchaseBooks_URA	selected	
Sync.M3FBM_HU_Out_AnnualVATReport_1965	selected	
Sync.M3FBM_HU_Out_AnnualVATReport_2065	selected	
Sync.M3FBM_HU_Out_AnnualVATReport_2165	selected	
Sync.M3FBM_IT_Out_VATReport_Esterometro_2019	selected	

Document	Write a file to a folder	Read a file from a folder
Sync.M3FBM_IT_Out_VATReport_Spesometro_2017	selected	
Sync.M3FBM_LT_Out_ConsignmentNote_iVAZ_133	selected	
Sync.M3FBM_LT_Out_VATReport_iSAF_11	selected	
Sync.M3FBM_NO_Out_VATReport_SAFT_2022	selected	
Sync.M3FBM_PH_Out_VATReport_Sales	selected	
Sync.M3FBM_PH_Out_VATReport_Purchases	selected	
Sync.M3FBM_SI_Out_VATReport_DDV_O	selected	
Sync.M3FBM_SK_Out_InvoiceList_VATReport_KVDPH_2017	selected	
Sync.M3FBM_SK_Out_VATReport_DPH2018	selected	
Sync.M3FBM_SK_Out_VATReport_DPH2021	selected	

c On the **Connection Points** page, select the M3 connection point and verify that the **Documents** tab includes all the inbound and outbound documents that are required for this integration:

Document	Receive in Application	Send from Application
Sync.M3FBM_AR_Out_VATReport_DigitalVATBook		selected
Sync.M3FBM_AT_Out_CustomerOrderInvoice_ebInterface		selected
Sync.M3FBM_AT_Out_VATReport_UVA_U30		selected
Sync.M3FBM_AT_Out_VATReport_UVA_U30_2021		selected
Sync.M3FBM_BE_Out_AnnualVAT-Report_LK7		selected
Sync.M3FBM_BE_Out_AnnualVATReport_LK9		selected
Sync.M3FBM_BG_Out_Electronic_VAT_Purchase-Books_POKUPKI		selected
Sync.M3FBM_BG_Out_Electronic_VAT_Sales-Books_PRODAGBI		selected
Load.M3FBM_BR_In_BankConfirmation_BDB	selected	
Load.M3FBM_BR_In_BankConfirmation_BRADESCO	selected	
Load.M3FBM_BR_In_BankConfirmation_HSBC	selected	
Load.M3FBM_BR_In_BankConfirmation_ITAU	selected	
Load.M3FBM_BR_In_BankConfirmation_Santander	selected	

Document	Receive in Application	Send from Application
Load.M3FBM_BR_In_COInvoice_NFe2_CancelReturn	selected	
Load.M3FBM_BR_In_COInvoice_NFe2_IssueReturn	selected	
Load.M3FBM_BR_In_COInvoice_NFe4_Approved_IssueReturn	selected	
Load.M3FBM_BR_In_COInvoice_NFe4_Unapproved_IssueReturn	selected	
Load.M3FBM_BR_In_COInvoice_Synchro31_CancelReturn	selected	
Load.M3FBM_BR_In_COInvoice_Synchro31_IssueReturn	selected	
Load.M3FBM_BR_In_ItemComplementFCI	selected	
Sync.M3FBM_BR_Out_BankRemittance_BDB		selected
Sync.M3FBM_BR_Out_BankRemittance_BRADESCO		selected
Sync.M3FBM_BR_Out_BankRemittance_HSBC		selected
Sync.M3FBM_BR_Out_BankRemittance_ITAU		selected
Sync.M3FBM_BR_Out_BankRemittance_Santander		selected
Sync.M3FBM_BR_Out_COInvoice_NFe2_CancelRequest		selected
Sync.M3FBM_BR_Out_COInvoice_NFeXML4_CancelRequest		selected
Sync.M3FBM_BR_Out_COInvoice_NFe4_Mastersaf_IssueRequest		selected
Sync.M3FBM_BR_Out_COInvoice_NFe4_2019_Mastersaf_IssueRequest		selected
Sync.M3FBM_BR_Out_COInvoice_NFe4_2021_Mastersaf_IssueRequest		selected
Sync.M3FBM_BR_Out_COInvoice_NFe4_Standard_IssueRequest		selected
Sync.M3FBM_BR_Out_COInvoice_NFe4_2019_Standard_IssueRequest		selected
Sync.M3FBM_BR_Out_COInvoice_NFe4_2021_Standard_IssueRequest		selected

Document	Receive in Application	Send from Application
Sync.M3FBM_BR_Out_COInvoice_Synchro31_CancelRequest		selected
Sync.M3FBM_BR_Out_COInvoice_Synchro4_IssueRequest		selected
Sync.M3FBM_BR_Out_ItemFCIDeclaration		selected
Sync.M3FBM_BR_Out_SupplierPayment_BDB		selected
Sync.M3FBM_BR_Out_SupplierPayment_HSBC		selected
Sync.M3FBM_CH_Out_SupplierPayment_SEPA_113_CH02_CreditSuisse		selected
Load.M3FBM_CN_In_GoldenTaxInvoice_IssuedInvoiceList	selected	
Sync.M3FBM_CL_Out_VATReport_RegistrodeCompras		selected
Sync.M3FBM_CN_Out_GoldenTaxInvoice_RejectedInvoiceList		selected
Sync.M3FBM_CN_Out_GoldenTaxInvoice_SalesInvoiceList		selected
Sync.M3FBM_CZ_Out_VAT_Report_Control_Statement_XML_2019		selected
Sync.M3FBM_CZ_Out_VAT_Return_Statement_DPHDP3		selected
Sync.M3FBM_DE_Out_VATReport_Elster_UStVA		selected
Sync.M3FBM_DE_Out_VATReport_Elster_UStVA_2021		selected
Sync.M3FBM_DE_Out_VATReport_UVA18		selected
Load.M3FBM_ES_In_OnlineVATReport_SII_RespuestaEmitidas_1_1Bis	selected	
Load.M3FBM_ES_In_OnlineVATReport_SII_RespuestaEmitidas_2018	selected	
Load.M3FBM_ES_In_OnlineVATReport_SII_RespuestaRecibidas_1_1Bis	selected	
Load.M3FBM_ES_In_OnlineVATReport_SII_RespuestaRecibidas_2018	selected	
Sync.M3FBM_ES_Out_CustomerOrderInvoice_Facturae_321		selected

Document	Receive in Application	Send from Application
Sync.M3FBM_ES_Out_OnlineVATReport_SII_FacturasEmitidas_1_1Bis		selected
Sync.M3FBM_ES_Out_OnlineVATReport_SII_FacturasEmitidas_2018		selected
Sync.M3FBM_ES_Out_OnlineVATReport_SII_FacturasRecibidas_1_1Bis		selected
Sync.M3FBM_ES_Out_OnlineVATReport_SII_FacturasRecibidas_2018		selected
Sync.M3FBM_ES_Out_VATReport_Format340		selected
Sync.M3FBM_ES_Out_VATReport_Format347_2015		selected
Load.M3FBM_FR_In_ExternalAccountStatement_ETEBAC	selected	
Sync.M3FBM_HR_Out_Electronic_VAT_PurchaseBooks_URA		selected
Sync.M3FBM_HU_Out_AnnualVATReport_1965		selected
Sync.M3FBM_HU_Out_AnnualVATReport_2065		selected
Sync.M3FBM_HU_Out_AnnualVATReport_2165		selected
Sync.M3FBM_IT_Out_VATReport_Esterometro_2019		selected
Sync.M3FBM_IT_Out_VATReport_Spesometro_2017		selected
Sync.M3FBM_LT_Out_ConsignmentNote_iVAZ_133		selected
Sync.M3FBM_LT_Out_VATReport_iSAF_11		selected
Sync.M3FBM_NO_Out_VATReport_SAFT_2022		selected
Sync.M3FBM_PH_Out_VATReport_Sales		selected
Sync.M3FBM_PH_Out_VATReport_Purchases		selected
Sync.M3FBM_SI_Out_VATReport_DDV_O		selected
Sync.M3FBM_SK_Out_InvoiceList_VATReport_KVDPH_2017		selected
Sync.M3FBM_SK_Out_VATReport_DPH2018		selected
Sync.M3FBM_SK_Out_VATReport_DPH2021		selected

Your connection point also includes documents for other integrations.

2 In ION Desk, select **Connect > Data Flows**.

3 On the **Data Flows** page, select the data flow and use the configured connection points. If the file connection point for M3 FBM named `M3FBM_FILE_CONN_POINT` is to be used, you must update the file connection point to use the correct enterprise location and connection details.

    a   In ION Desk, select **Connect > Connection Points** and select `M3FBM_FILE_CONN_POINT`.

    b   On the **Documents** tab, update the Write Location, Read Location, Archive Location and Error Location. Update the tenant with the actual tenant value.

          For inbound FBM messages, update the accounting entity with the actual accounting entity value. The placement of the accounting entity in relation to the folder structure in the read location should be reflected in the FilePathAELevel control property of the agreement. This can be updated in **Enterprise Collaborator Administration > Server > Agreement Control Properties**.

          For outbound FBM messages, [accounting_entity] will depend on the value of the accounting entity sent in the manifest info for the FBM mapping. Therefore, it is not necessary to update [accounting_entity] for outbound FBM messages.

    c   Click **Save**.

    d   Click **Activate/Reactivate**.

If a file connection point is to be used but does not exist, create a file connection point. Follow the instructions in the *Infor ION Technology Connectors Administration Guide–Cloud Edition*.

Furthermore, if an enterprise location is not available for use in the file connection point, create an enterprise location. Follow the instructions in the "Enterprise Connector application connection points" section of the *Infor ION Desk User Guide - Cloud Edition*.

Chapter 5: M3 Cloud File Transfer Client administration tasks

After successfully installing M3 Cloud File Transfer Client (CFT Client) on premises, you can perform administration tasks on both the CFT Client and the Infor ION Grid from which the client is installed.

See information about CFT Client installation in the *Infor M3 Cloud Configuration Guide*.

Setting tenant configuration to enable export of files to Amazon S3

Optionally, files from print programs that are enabled for configurable XML can be an output to a private Amazon S3 bucket of the client. To do this, configure the tenant configuration first in Infor M3 Cloud File Transfer Agent. The configuration is stored encrypted.

Disabling automatic upgrade of CFT Client

The M3 Cloud File Transfer client detects any new version of the application and can upgrade itself automatically. Optionally, this functionality can be disabled:

1   In the Grid Admin UI, click the link Application CloudFileTransferClient.
2   In the application menu, select **Properties**.
3   Click **CFT Client upgrade flag** and clear the check box.
4   Click **Update Property** or **Create Property** if doing this for the first time.

Administrating Infor ION Grid for CFT Client

The M3 Cloud File Transfer Client (CFT Client) is installed as an application in Infor ION Grid.

For information about how to start the Grid Admin UI, how to start and stop the grid, and other administrative tasks, see *Infor ION Grid Administration Guide.*

Upgrading Infor ION Grid

The M3 Cloud File Transfer detects if there is a new grid version available and recommends an upgrade. The download of the new Grid Installer is done automatically, but the upgrade still requires manual intervention. The instructions of the manual upgrade will be visible in the status of the CFT Client and duplicated in the logs as well.

1 Select the grid installation location (for reference the location that will be used for this document is `C:\Infor\InforIONGrid`)

2 Go to `C:\Infor\InforIONGrid\grid\applications\CloudFileTransfer\appdata\installers`

3 As Administrator, run `gridinstaller-2.0.<XXXXXXXX-XXXXXX>.cmd`.

Disabling automatic upgrade of Infor ION Grid

The M3 Cloud File Transfer detects if there is a new grid version available and recommends an upgrade. Optionally, this functionality can be disabled:

1 In the Grid Admin UI, click the link Application CloudFileTransferClient.

2 In the application menu, select **Properties**.

3 Click **Grid Client upgrade flag** and click **Edit**.

4 Clear the check box and click **Save**.

5 Click **Update Property** or **Create Property** if doing this for the first time.

Configuring the CFT Client server for proxy server use

If the CFT Client server is using a proxy server, the REST calls to CFT Agent cannot be executed. You must set the settings for proxy host and port in CFT Client Grid.

1 In the Grid Admin UI, click the link Application CloudFileTransferClient.

2 In the application menu, select **Properties**.

3 Filter on grid.jvm.generic.

4 Click **Generic JVM Commands**.

5 Click **M3CFTClientNode**.

6 Click **Edit**.

7 Add these lines:

```
-Dhttps.proxyHost=<proxy server address>
-Dhttps.proxyPort=<proxy server port>
```

8   Click **Save**.

9   Restart the Cloud File Transfer Client application.

Chapter 6: Using M3 Business Engine administration tools

These administration tools for M3 Business Engine (M3 BE) are available from the Infor M3 menu in Infor Ming.le:

Tool	Description	Required IFS Security role
Business Engine Configuration Data	Use this tool to import configuration data to the M3 BE database. See Importing configuration data to the M3 BE database on page 65.	*M3BE-ConfigUser* to import configuration data *M3BE-ConfigAdmin* to import configuration data as a different M3 user
Business Engine Data Management	Use these tools to export or import data on the company or division level, delete company data, and update data by modifying the content of a specific record. See Managing M3 BE data on page 67.	*M3BE-DBAdmin* to maintain data *M3BE-DBViewer* to view data through the Update Database tool
Business Engine Field Audit Trail	Use this tool to set up audits of data changes. See Using the Field Audit Trail on page 78.	*M3BE-FATAdmin*
Business Engine Interactive Program Services	Use this tool to automate the execution of interactive M3 BE programs through web services. See Using M3 BE Interactive Program Services on page 81.	*M3BE-ServicesAdmin*
Business Engine Jobs	Use this tool to view and delete jobs for troubleshooting, as well as configure job logging. See Viewing and killing jobs on page 376.	*M3BE-FndUser* to manage jobs for your user *M3BE-LogEnabler* to configure job logging for your user *M3BE-FndAdmin* to manage jobs and job logging for all users

Tool	Description	Required IFS Security role
Business Engine Logs	Use this tool to view and download logs for troubleshooting. See Managing Business Engine logs on page 378.	*M3BE-FndUser* *M3BE-FndAdmin* to manage log files for to manage log files for all users
Business Engine Files	Use this tool to manage M3 BE files. See Managing M3 BE files on page 86.	To manage M3 BE files, different IFS Security roles are required depending on the file location: • Data Management (Import/Export) in BE-DataMgmt folder To view, download, upload, and delete files, you must be connected to the role *M3BE-DBAdmin*. • Configuration data in config_data folder To view, download, upload, and delete files, you must be connected to the role *M3BE-ConfigAdmin*. To view and download files, you must be connected to the role *M3BE-ConfigUser*. • Files in transfer to or from M3 Business Engine in MvxFileTransfer folder To view, download, upload, and delete files, as well as create folders, you must be connected to the role *M3BE-FileAdmin*. To view and download files, you must be connected to the role *M3BE-FileViewer*.

Importing configuration data to the M3 BE database

You can import configuration data from a zip file to the M3 Business Engine (BE) database. Importing configuration data is useful when Infor delivers template data or when exported data must be moved

between different M3 BE companies or environments. The M3 BE programs that you must use to export the configuration data depends on the type of configuration data. For example, Ad Hoc Report Group. Open (AHS100) is used to export configuration data for ad hoc reports.

During import, the zip file is read, and specified M3 BE API programs and transactions are used to add the data to the M3 BE database.

To import configuration data, select **Administration Tools > Business Engine Configuration Data** from the Infor M3 menu in Infor Ming.le. You must be connected to the IFS Security role *M3BE-ConfigAdmin* or *M3BE-ConfigUser* to access the link.

Prerequisites to importing configuration data

Note: These prerequisites apply only when importing already exported configuration data. These prerequisites do not apply when importing template configuration data delivered by Infor.

To import exported configuration data zip files to the M3 BE database, an M3 Cloud File Transfer (CFT) Client installation is required. Zip files with the exported data will be stored in the `M3FILES` path that is configured during the CFT Client installation. This same path is used during import. The full path that the tools use is `[M3FILES]\MvxFileTransfer\config_data`.

If configuration data must be moved between different environments or tenants, move the exported file from the `[M3FILES]\MvxFileTransfer\config_data` location of the current environment to the `[M3FILES]\MvxFileTransfer\config_data` location of the target environment.

Note: The config_data folder can be managed by using the M3 Business Engine Files administration tool.
See

Importing configuration data

On **Business Engine Configuration Data** page, there are two tabs for importing configuration data:

• Use **Exported** for importing previously exported configuration data zip files.

• Use **Content** for importing template configuration data delivered by Infor.

1 On **Business Engine Configuration Data** page, click the applicable configuration data tab.

2 Select the data zip file to import. Optionally, filter on file name, size, and status to find the zip file.

3 Click the **Import** icon.

4 In the **Import Configuration Data** window, specify this information:

   M3 user
   Use the default or specify a different user.
   Note: You can only specify a different user if you are connected to the role *M3BE-ConfigAmin*.

   Company
   Use the default or specify a different company.

Division
Use the default or specify a different division.
Note: Ensure that the M3 user that you specified has access to the specified company and division.

Replace existing records
Select this if existing data should be replaced, then a Delete transaction is run before the Add transaction.

Keep existing records
Select this if existing data should be retained, then only the Add transaction is run.

5 Click **Import**.
6 When the task finished, a message is displayed. Click **Ok**.

Managing M3 BE data

There are several ways to maintain data in the M3 Business Engine (M3 BE) database:

- Export data on company or division level.
- Import data on company or division level.
- Delete a specific company.
- Update data by modifying the content of a specific record.

To maintain data in the M3 BE database, select **Administration Tools > Business Engine Data Management** from the Infor M3 menu in Infor Ming.le. You must be connected to the IFS security role *M3BE-DBAdmin, M3BE-DBAdmin-ReadToken* or *M3BE-DBViewer*.

- To maintain data by using the tools, the role *M3BE-DBAdmin* is required.
- To manage read access for importing data from an environment, the role *M3BE-DBAdmin-ReadToken* is required.
- To view data through the Update Database tool, the role *M3BE-DBViewer* is required.

Prerequisites to managing M3 BE data

Note: These prerequisites only apply to exporting and importing data, not to deleting company and updating database.
- To export or import M3 BE data, an M3 Cloud File Transfer (CFT) Client installation is required. Zip files with the exported data will be stored in the `M3FILES` path that is configured during the CFT Client installation. This same path is used during import. The full path that these tools use is `[M3FILES]\MvxFileTransfer\BEDataMgmt`.

  Note: The BEDataMgmt folder can be managed by using the M3 Business Engine Files administration tool.

  See Managing M3 BE files on page 86.

- Templates for export are stored in `[M3FILES]\MvxFileTransfer\BEDataMgmt\templates\ export`.
- Templates for Import from Database functionality are stored in `[M3FILES]\MvxFileTransfer\ BEDataMgmt\templates\copy`.
- Log files are stored in `[M3FILES]\MvxFileTransfer\BEDataMgmt\log`.

Company/division

Infor only supports export/import on company or division level. No tables without a company field will be exported/imported. Before using the tool, consider these limitations:

- **Company export/import:** Only tables with a company field will be included. Note that we do not verify if company field is in the unique key.
- **Division export/import:** Only tables that has both a company field AND a division field will be included. Note that we do not verify that these two fields are in the unique key.
- The user must know which companies and divisions that exists in the environment and what tables to select during export.
- All tables available in the exported data zip file will be used with the company/division filter that is given during import.

Tables excluded from import/export

Currently, three tables are blocked from Export and Import and these are CMNUSR, CMNLNG, and BULKMI. These tables are selectable in the list of tables but will never be included in the export or import of data.

Import strategy

We recommend that the default import strategy **Keep existing records** is used. We also recommend to skip the **Allow data mismatch** option, which allows the format of the existing table and the imported data to differ without reporting back any errors.

Exporting M3 Business Engine data

Use this procedure to export data from the M3 Business Engine database to a zip file.

General information

- A zip file is always created regardless if it contains any data or not.
- All selected tables, even those that are empty, will be included in the zip file.
- Log files can be downloaded, either from the **Job Overview** tab or from the **Business Engine Files** page.

    See Viewing and managing M3 Business Engine data management jobs on page 76 or Managing M3 BE files on page 86.

Selecting tables to export

In the **Export** tab, a list of all tables is displayed (tables excluded from the list are join-tables). At the bottom left are the number of tables selected out of all available tables in the list. At the bottom, you can step forward or back in the list of tables. You can sort the list by clicking the column names. You can filter the list by specifying the table you are looking for in the filter field.

You can select tables in several ways:

- Manually select tables by ticking the check boxes in front of the table name.
- Select/deselect All tables by selecting the top check box.
  It is possible to deselect/select specific tables after you have selected the top check box.

- Select to export using a template by clicking the **Export with template** button.
  During export, all tables available in this template will be used.

Exporting a company

You can export a company with or without a template.
Note: Company copy can be done between environments or within the same environment.
1   On the **Business Engine Data Management** page, click the **Export** tab.
2   Select the component to export data.
3   Select the tables to export.
    Note: Select tables only when exporting data without a template. Skip this step if you intend to use a template for the export.

4   You can select from these options for company export:
    a   To export data without using a template, click **Export** and specify this information:

        Company

        Specify the company to export. Only the selected tables where company=*<the specified value>* are exported.

        All companies

        Alternatively, select to export all companies. Only the selected tables with a company field are exported.

        File name

        Specify a zip file name. The file name specified here will get a time-stamp prefix when created. Zip file names must end with .zip and valid characters are: a-z A-Z 0-9 . _ -

        Save as template

        Specify a template name if you want to save this run as a template to be used in future runs.

        Valid characters are: a-z A-Z 0-9 . _ -

    b   To export data using a template, click **Export with template** and specify this information:
        Template
        Select the template to use during export.

Company

Optionally, specify the company to export. This field is automatically populated based on the template you selected.

All companies

Optionally, select to export all companies. This field is automatically populated based on the template you selected.

File name

Specify a zip file name. The file name specified here will get a time-stamp prefix when created. Zip file names must end with .zip and valid characters are: a-z A-Z 0-9 . _ -

5   Click **Export**. A BE Data Management Export job is submitted.

6   Click **Job Overview** to go to the **Job Overview** tab. See Viewing and managing M3 Business Engine data management jobs on page 76. When the job is finished, the data zip file will be ready for import.

A zip file is always created regardless of if it contains any data or not. All selected tables, even those that are empty are included in the .zip file.

Exporting a division

You can export a division with or without a template.

Note: Division copy should only be done within the same environment.

1   On the **Business Engine Data Management** page, click the **Export** tab.

2   Select the component to export data from.

3   Select the tables to export.

Note: Select tables only when exporting data without a template. Skip this step if you intend to use a template for the export.

4   You can select from these options for division export:

a   To export data without using a template, click **Export** and specify this information:

Company

Specify a company to export a division for.

Filter on Division

Select the check box.

Division

Specify the division to export. Only the selected tables where company=*<the specified value>* with division=*<the specified value>* and division=BLANK(empty) will be exported.

File name

Specify a zip file name. The file name specified here will get a time-stamp prefix when created. Zip file names must end with .zip and valid characters are: a-z A-Z 0-9 . _ -

Save as template

Specify a template name if you want to save this run as a template to be used in future runs. Valid characters are: a-z A-Z 0-9 . _ -

b   To export data using a template, click **Export with template** and specify this information:

Template
Select the template to use during export.

Company

Optionally, specify a company to export a division for. This field is automatically populated based on the template you selected.

Filter on Division

Select the check box.

Division

Specify the division to export. Only the selected tables where company=*<the specified value>* with division=*<the specified value>* and division=BLANK(empty) will be exported.

File name

Specify a zip file name. The file name specified here will get a time-stamp prefix when created. Zip file names must end with .zip and valid characters are: a-z A-Z 0-9 . _ -

5   Click **Export**. A BE Data Management Export job is submitted.

6   Click **Job Overview** to go to the **Job Overview** tab. See Viewing and managing M3 Business Engine data management jobs on page 76. When the job is finished, the data zip file will be ready for import.

A zip file is always created regardless of if it contains any data or not. All selected tables, even those that are empty are included in the .zip file.

Importing M3 Business Engine data

Use this procedure to import data from data zip files to the M3 Business Engine database.

In the **Import** tab, a list of exported data zip files will be displayed. Log files can be downloaded, either from the dialog that pops up after the run or from the Business Engine Files page.

See Managing M3 BE files on page 86.

Importing a company

Note: Company copy can be done between environments or within the same environment.
1   On the **Business Engine Data Management** page, click the **Import** tab.
2   Select a component.

3 Select the data zip file to import.

4 Click **Import**. The **Import data** window is displayed.

5 Based on what is available in the data zip file, consider these options for company import:

a. To import a specific company, specify this information:

Source company

Specify the company available in the data zip file. Only the tables where company=*<the specified value>* are imported.

Target company

Specify the target company for the import.

b. To import all companies, specify this information:

All companies

Select the check box to import all companies. Only the tables available in the data zip file with a company field will be imported.

6 Select an import strategy:

a **Keep existing records:** The record that exists in both source and target are left unchanged in the target.

b **Replace existing records:** The records in target that have a duplicate in source are replaced by the ones imported. For example, tables like MITBAL that has CONO and WHLO as key but also a DIVI field not in key, will be replaced during import.

c **Clear target table:** The tool clears the target tables of all entries that match the target company before import.

7 Skip the **Allow data mismatch** option (It allows the format of the existing table and the imported data to differ without causing any errors.

8 Click **Import**. A BE Data Management Import job is submitted.

9 Click **Job Overview** to go to the **Job Overview** tab. See <u>Viewing and managing M3 Business Engine data management jobs</u> on page 76. When the job is finished, the data has been imported into the database.

Importing a division

Note: Division copy should only be done within the same environment.

1 On the **Business Engine Data Management** page, click the **Import** tab.

2 Select a component.

3 Select the data zip file to import.

4 Click **Import**. The **Import data** window is displayed.

5 Specify this information:

Source company

Specify the company available in the data zip file. Only the tables where company=*<the specified value>* are imported.

Target company

Specify a target company. During division import you should have a target company that already exists in the environment.

Filter on division

Select Filter on division.

Source division

Specify the division available in the data zip file. Only the tables where division=<*the specified value*> are imported.

Target division

Specify a target division.

Normally, you do not require the central division (blank division) to be included. This should already exist in the environment.

6   Select an import strategy:

a   **Keep existing records:** The record that exists in both source and target are left unchanged in the target.

b   **Replace existing records:** The records in target that have a duplicate in source are replaced by the ones imported. For example, tables like MITBAL that has CONO and WHLO as key but also a DIVI field not in key, will be replaced during import.

c   **Clear target table:** The tool clears the target tables of all entries that match the target company and division before import. If the check box **Include central division** is selected, the tool clears entries that contain the specified company and the central division.

7   Skip the **Allow data mismatch** option (It allows the format of the existing table and the imported data to differ without causing any errors.)

8   Click **Import**. A BE Data Management Import job is submitted.

9   Click **Job Overview** to go to the **Job Overview** tab. See <u>Viewing and managing M3 Business Engine data management jobs</u> on page 76. When the job is finished, the data has been imported into the database.

Exporting and importing M3 Business Engine data between environments (tenants)

To export and import M3 Business Engine data between environments, there are two options:

*   To use the Import from Database procedure to copy the data in one go. See <u>Importing M3 Business Engine data from a database</u> on page 74.
*   To first use the export procedure in the source environment and then the import procedure in the target environment. See below.

Use this procedure to export M3 Business Engine data from one environment (tenant), and import it to another environment (tenant).

1 On the **Business Engine Data Management** page, use the export procedure for the applicable data.

See Exporting M3 Business Engine data on page 68.

2 Move the exported file from the `[M3FILES]\MvxFileTransfer\BEDataMgmt` location of the current environment (tenant) to the `[M3FILES]\MvxFileTransfer\BEDataMgmt` location of the target environment (tenant).

Note: The BEDataMgmt folder can be managed by using the M3 Business Engine Files administration tool.

See Managing M3 BE files on page 86.

3 Log in as the target tenant, and import the .zip package for the target tenant environment.

See Importing M3 Business Engine data on page 71.

Importing M3 Business Engine data from a database

Use this procedure to copy company data between different environments, or to copy data for a company or division within the same environment.

Note: The Import from Database operation effectively performs an export operation followed by an import operation for each database table. If it is needed to optimize the export time, use the export procedure described in Exporting M3 Business Engine data on page 68.

To import data from a source environment, read access in the source environment must be established first. This is described in Managing read tokens in the source environment on page 74.

Managing read tokens in the source environment

Managing read tokens require the role *M3BE-DBAdmin-ReadToken*.
Note: A read token is required even if the source environment for the import operation is the same as the target environment.

1 In the source environment, click the **Import from Database** tab on the **Business Engine Data Management** page.

2 On the menu, select **Manage read access**. The tab displays a list of existing read tokens.

3 To refresh the list, click **Refresh**.

4 To revoke one or more tokens, select the tokens from the list and click the **Revoke** button. Click **OK** to confirm the action.

5 To create a new read token, click **Create**. In the dialog, select the target environment and expiration time, max length is 90 days. Click **Create**.

6 Download the read token and the token password.

Note: This is the only time that the read token and the password are available.

Importing a company or division

1 On the **Business Engine Data Management** page, click the **Import from Database** tab.

2 Select a source environment by specifying a read token issued by the source environment administrator. See Managing read tokens in the source environment on page 74.

    a Click **Configure**.

    b Specify the read token and the read token password.

    c Click **Submit**.

    d Click **OK**.

3 You now have two options:

    • To import data without using a template, then continue with step 4.

    • To import data using a template, then continue with step 7.

4 Select a component.

5 Select tables to import.

6 Click **Import** and specify this information:

Source company
Specify the company in the source environment to copy. Only the selected tables where company=*<the specified value>* are copied.

Target company
Specify the company in the target environment to import to.

All companies
Alternatively, select to import all companies. Only the selected tables with a company field are imported.

Filter on division
Select this option to import a division.
Note: Division copy should only be done within the same environment.

Source division
Specify a division to copy. Only the tables where division=*<the specified value>* are copied.

Target division
Specify a division to copy to.

Include central division
Normally, you do not require the central division (blank division) to be included. This should already exist in the environment.

For the **Select strategy** field, specify this information:

Replace existing records
Select this option to identify records in target with duplicates in source that should be replaced by imported records.
For example, tables like MITBAL that has CONO and WHLO as key but also a DIVI field not in key, will be replaced during import.

Keep existing records
Select this option to keep records in the target table.

Clear target table
Select this option to clear the target tables of all entries that match either the target company or division before import. If the check box **Include central division is selected**, the tool clears entries that contain the specified company and the central division.

Save as template
Specify a template name if you want to save this run as a template to be used in future runs. Valid characters are: a-z A-Z 0-9 . _ -

7   Skip this step if you import without using a template. To import data using a template, click **Import with template** and specify this information:

Template
Select the template to use during import. This will populate the remaining fields based on the template. Optionally, the fields can be edited as described in step 6.

8   Click **Import**. A BE Data Management Import from Database job is submitted.

9   Click **Job Overview** to go to the **Job Overview** tab. See Viewing and managing M3 Business Engine data management jobs on page 76. When the job is finished, the data has been imported into the database.

Deleting M3 Business Engine company data

Use this procedure to delete all M3 Business Engine data for a company.

1   On the **Business Engine Data Management** page, click the **Delete Company** tab.

2   Select a component to delete a company from.

3   Specify the company.

4   Click **OK**.

5   In the confirmation dialog box, click **Yes** to to submit the delete job.

6   Click **Job Overview** to go to the **Job Overview** tab. See Viewing and managing M3 Business Engine data management jobs on page 76. When the job is finished, the company data is deleted from the database.

Viewing and managing M3 Business Engine data management jobs

Use this tab to view, abort, or download log files of submitted Export, Import, Import from Database and Delete Company jobs.
Note: Only the latest job information can be viewed. Log files for older jobs are located in M3BE Files.

1 On the **Business Engine Data Management** page, click the **Job Overview** tab. The tab displays a list of submitted Export, Import, Import from Database and Delete Company jobs. The status column indicates the current status of each job.

This table shows all available statuses:

Status	Description
PENDING	Indicates that the job is placed in queue and will be started as soon as there are system resources available.
RUNNING	Indicates that the job is currently executing.
OK	Indicates that the job has successfully completed.
NOT_OK	Indicates that the job has finished with an error. Details about the error can be found in the log file.
CFT_UNAVAILABLE	Indicates that the job could not be started or was terminated because the file transfer service was unavailable.
UNEXPECTED_TERMINATION	Indicates that the job was terminated due to an internal error.
ABORTED	Indicates that the job was aborted by a user.

2 To refresh the list, click **Refresh**.

3 To hide or show columns, click the menu button and **Personalize Columns**, and use the check boxes to toggle the visibility of each column.

4 To abort a job, select the job from the list and click **Abort**. It is only possible to abort a job with status PENDING.

5 To download a log file, click the icon in the Download log column for the job.

Updating a specific record in the database

Use the **Update Database** tab to view or modify the content of a specific record in the M3 Business Engine database.

1 On the **Business Engine Data Management** page, click the **Update Database** tab.

2 Specify the name of the applicable table and click **Open**. If the table belongs to a different accounting entity than the default company and division, change these settings before specifying the table:

  a Click the **Settings** icon.

  b In **Settings**, select the applicable combination of company and division, or select the **Use default company and division** check box.

Once the table data is displayed, you can change sorting order from the menu.

3 Optionally, you can filter records in the displayed table:

  a Click the **Filter records** icon.

  b Select a field and an SQL operator, and specify a value to match the field against.

Note: Click the question mark for details on how the SQL operators are used.

c   Click **Add** to add the filter. Multiple filters can be added.

d   Click **OK**. The records that matches the filters are listed.

Select the check box next to **Filter records** to enable the filtering. Clear the check box to disable the filter.

4   Use the positioning fields above the column names to search for the record to edit, or browse through the table using **Next** and **Previous**.

5   To update a record, select the record and click the **Edit** icon in the menu. A detailed view of the record is displayed.

6   Make your changes and click **Submit**.

7   Click **Yes** in the confirmation dialog box.

Using the Field Audit Trail

Use the Field Audit Trail to set up audits of data changes made to specific M3 Business Engine database fields. Field Audit Trail enables you to keep track of when the changes were made, what the previous value was prior to the change, and which user made the change.

To manage field audit trails, select **Administration Tools > Business Engine Field Audit Trail** from the Infor M3 menu in Infor Ming.le. You must be connected to the IFS security role *M3BE-FATAdmin* to access the link. To export data, the role *M3BE-FATAdmin-Exporter* is also required.

Adding a new audit

1   On the **Business Engine Field Audit Trail** page, click **New**.

2   Select a table to audit.

3   Select the **Active** check box if the audit trail should be active.

4   Select any of these audit operations:

- Insert
- Update
- Delete

5   Select the fields to be audited from the **Non-audited fields** list box and move them to the **Audited fields** box.

6   Click **Save**.

Editing an audit

1   On the **Business Engine Field Audit Trail** page, select an audit in the Current Audits table and click **Edit**.

2   Change audit operations, add or remove fields in the audit, or deactivate the entire audit by clearing the **Active** check box.

3   Click **Update**.

Deleting an audit

Deletions can only be made for audit trails that do not contain any data yet.

1   On the **Business Engine Field Audit Trail** page, select an audit in the Current Audits table.

2   Click **Delete**.

3   Verify that you have selected the correct audit, and click **Yes**.

Exporting an audit

To export field audit data, the IFS security role *M3BE-FATAdmin-Exporter* is required. The Export button is only enabled if you are connected to this role.

Note: There is a default export limit of 100,000 rows and only one export job is allowed at a time.

1   On the **Business Engine Field Audit Trail** page, select an audit in the Current Audits table.

2   Click **Export**.

3   Specify this information:

Start date
Select the start date for the date interval from which you want to export data.

End date
Select the end date for the date interval from which you want to export data.

Partition number
Depending on the date interval given, a list of available partitions is displayed. Select a partition number for your export.

Separator
Select a separator.

4   Optionally, clear the **Include column description** check box to remove a column description. This check box is selected by default.

5   Specify the file name.

6   Click **Export**.

7   An information message is displayed when the export is started. Click **OK**.

8   It will not be possible to start a new export job until the previous export job is finished. When the export is finished, you can download the exported file from the **Business Engine Files** page by selecting FAT Export under file location.

See information about [Managing M3 BE files](#) on page 86.

Viewing history of audit settings

1   On the **Business Engine Field Audit Trail** page, select an audit in the Current Audits table.

2   Under the History header, click **Show More**.

The history of audit settings is displayed in two tables. The user column shows the user responsible for the change of settings, and the Date and Time columns shows the time stamp when the change were made. The two tables are:

- Table

  Shows the history of changes made to audit operations on the audited table.

- Fields

  Shows the history of changes made to the audited fields.

3   To hide the tables, click **Show Less**.

Viewing data changes of audited fields

1   On the **Business Engine Field Audit Trail** page, select an audit in the Current Audits table.

2   Under the Data header, click **Show More**.
The history of data changes is displayed in two tables:
- Keys

  Shows the key records of the modified row.

- Data

  Shows the data changes of the audited fields for the selected row in the Keys table. It shows the date and time for the change, the user who made the change, the operation, and the history values. To divide the audited information into smaller units, partitions are available. Partitions are automatically created when an audit is created or changed, for example, if activating an audit or if a new field is added to the audit. Partitions are displayed in the table as a header in bold font.

3   To hide the tables, click **Show Less**.

Using M3 BE Interactive Program Services

Interactive Program Services is a tool built in M3 Business Engine (M3 BE) that enables you to automate the execution of interactive M3 BE programs. To execute a program or a series of programs automatically, you can create a service that contains all the required metadata through the Interactive Program Services Designer. A SOAP request is then sent to M3 BE with the variables, and the result from the execution is returned in a SOAP response.

Interactive Program Services also supports legacy web services, that is, in the form of MPDs from the M3 Web Service tool, through REST API.

To access the Interactive Programming Services Designer, select **Administration Tools > Business Engine Interactive Program Services** from the Infor M3 menu in Infor Ming.le. You must be connected to the IFS Security role *M3BE-ServicesAdmin*.

About the Interactive Program Services Designer

Use the Interactive Program Services Designer to create, update, and test web services and all sub-components that belong to a web service.

Before creating a web service, you should first determine the M3 BE functionality that should be exposed as a web service. We recommend that you accomplish these preparatory steps:

- Decide the M3 BE program flow that should be exposed as a web service.
  Note: If an MI program or transaction already exists for that M3 BE program, use that MI program instead of creating an IPS web service.

  An IPS web service is created to script an existing M3 BE program. If the M3 BE program is modified, then the IPS web service might need to updated to work.

- Start the M3 BE program in Infor Ming.le and run through the program flow.
- Note the panels that the program goes through to accomplish the task.
- Note the fields in each panel that are required for input. These key fields are typically found on the A and B panels.
- Note the fields in each panel that should be added as output.

Note: To view field ID information in Infor Ming.le/H5, hover over the field and press **F1**. The field ID can then be used in the Interactive Program Services Designer to ensure that the correct field is added to the web service.

This is an example of a basic web service that fetches description for a currency. The M3 BE program in Infor Ming.le/H5 for the program flow is *CRS055 - Currency. Open*:

Definition	Value
Service	CRS055Demo
Method	Display
Wrapped program(s)	CRS055

Definition	Value
Panel Sequence	A (ENTER)
	E (F3)
Input	WQCUCD, A-Panel, Mandatory
	WWOPT2, A-Panel, CONSTANT=5
	WWPSEQ, A-Panel, CONSTANT=E
Output	WTTX40, E-Panel

Follow these steps to start working with web services:

- Create or load a web service.

  The web service serves as the container for methods, which in turn specifies the programs to be executed.

- Create a method.

  A method contains programs to be executed, as well as the relationship of these programs to one another. In every method, there is always a main program that is the starting point for an execution. A main program can then have related subprograms.

- Add a program.

  A program contains the panel sequence to be followed. A program also contains all input and output parameters to be used in the execution of a method on the service. Input parameters are data handled by the program that control the behavior and execution of the program. Output parameters are expected data to be extracted and returned in the SOAP response after executing the program.

- Test the web service.

  A test case is used to verify the functionality of a web service.

Creating or loading a web service

You can create a new web service or load an existing web service from the Interactive Program Services home page.

1 On the **Interactive Program Services** home page, select **New > New Service**.

2 Specify this information:

Name
Specify a name for the web service.

System Configuration
Select a system configuration.

3 Click **Create**.

The new service is created and automatically selected on the home page.

Alternatively, you can load and modify an existing web service. Select **File > Load** from the home page and select the web service to display.

Creating a method

1   On the **Interactive Program Services** home page, select **New > New Method**.

2   Specify a name for the method.

3   Click **Create**.

4   To wrap programs to this method, click **Add**.

5   Select a program.

6   Click **Add**.

 Note: If the program that you selected contains a corresponding MI program, a confirmation message stating that you use the MI program instead of creating a web service is displayed. Click **Add** to proceed.

Adding a panel sequence

Specify the program panel sequence for a method. The sequence that you specify determines available input and output parameters.

1   Select a program and expand **Panel Sequence**.

2   Click **Add**.

3   Specify this information:

 Panel
 Select a panel from the list.

 Function Key
 Select a function key from the list.

4   Click **Add**.

5   Repeat steps 2 to 4 to add more panels.

6   Optionally, use the **Up** and **Down** buttons to reorder panels.

Specifying input and output parameters

1   To specify program input fields, expand **Input to program name**.

2   In the Available Inputs table, select parameters to add, and then click **>** to move selected parameters to the Selected Inputs table.

3   In the Selected Inputs table, specify this information:

 Constraint
 Select one of these values:

- Constant sets the value as the default value and cannot be changed in the executing SOAP call.
- Mandatory sets that the value must be specified in the executing SOAP call.
- Optional sets that the value may be specified in the executing SOAP call. If it is not, the default value will be used.

Default Value
Specify the default value to be used based on the constraint you selected.

4   To specify program output fields, expand **Output from program name**

5   In the Available Outputs table, select parameters to add, and then click **>** to move selected parameters to the Selected Outputs table.

6   To save all your changes, select **File > Save**.

7   Provide a description of the changes you made in the **Description** field.

8   Click **Save**.

Testing a web service

Interactive Program Services Designer provides the capability to create test cases and execute them to validate a web service. Testing a web service involves two tasks: creating test cases and running the test cases.

Note: Before executing a test case for web service, any updates to the web service must be saved. The test case will not be executed based on the current state of the web service in the Interactive Program Services Designer.

Creating a test case

1   Select a web service and a method to test.

2   On the **Interactive Programming Services** home page, select **New**.

3   Specify this information:

Name
Specify a name for the test case.

Description
Specify a description for the test case.

4   Click **Create**.

5   To save all your changes, select **File > Save**.

6   Provide a description of the changes you made in the **Description** field.

7   Click **Save**.

Executing a test case

1   Select a test case.

2   Specify this information:

Company / Division
Select a company / division from the list.

Input values
Specify the input value to test.

3   Select the **>** icon.

4   Click **Execute Service**

Test results are displayed on the page.

Exporting a web service

1   On the **Interactive Programming Services** home page, select **File > Export**.

2   Select a service to export.

3   Click **Export**. A .json file is saved on your local machine.

Importing a web service

You can import files containing web service definitions in the standard .json format or in the legacy
.mpd M3 web services format. Upon import of legacy format files, the tool converts them into standard
.json format before saving.

1   On the **Interactive Programming Services** home page, select **File > Import**.

2   In the **Import Service** dialog window, click **Choose File**.

3   Browse for the file to import.

4   Click **Open**.

Renaming a web service

You can rename an existing web service by loading it and selecting **File > Rename**.

Deleting a web service

You can delete a web service by loading a service and selecting **File > Delete**. A warning message that indicates deleting a service cannot be undone will be displayed.

Consuming a web service

The WSDL for a web service is found by using this path: `https://<server-access-path>/ips/service/webservicename?wsdl`. The server access path is specific for each environment and must be provided by the system administrator.

Managing M3 BE files

You can download, upload, and delete files in the MvxFileTransfer folder, the BE Data Management folder and the BE Configuration Data folder, including subfolders, by using the Business Engine Files tool.

To access this tool, select **Administration Tools > Business Engine Files** from the Infor M3 menu in Infor Ming.le.

To manage M3 BE files, different IFS Security roles are required depending on the file location:

- Data Management (Import/Export) in BEDataMgmt folder

  To view, download, upload, and delete files, as well as create folders, you must be connected to the role *M3BE-DBAdmin*.

- Configuration data in config_data folder

  To view, download, upload, and delete files, as well as create folders, you must be connected to the role *M3BE-ConfigAdmin*.

  To view and download files, you must be connected to the role *M3BE-ConfigUser*.

- Files in transfer to or from M3 Business Engine in MvxFileTransfer folder

  To view, download, upload, and delete files, as well as create folders, you must be connected to the role *M3BE-FileAdmin*.

  To view and download files, you must be connected to the role *M3BE-FileViewer*.

- Exports of Business Engine Field Audit Trail data in FAT Export folder

  To manage the files, you must be connected to the role *M3BE-FATAdmin-Exporter*.

Using the Business Engine Files tool

1 On the **Business Engine Files** page, select a file location from the list:

- Data Management (Import/Export)
- Configuration data
- MvxFileTransfer
- FAT Export

Note: Only file locations that the user has access to are displayed.

2 Optionally, click a file name to specify a file.

3 Click a folder to navigate into that folder. Clicking a folder named **..** navigates to the parent folder in the file hierarchy.

4 Do any of these actions:
- Click **Refresh** to refresh the list of files on the specified location.
- Click **Download** to save a selected file.
- Click **Upload** to add a file to the specified location.
- Click **Delete** to remove a selected file.
- Click **New folder** to create a folder on the specified location.

Chapter 7: Output Management for M3 CE Core

This solution is intended for administrators who need to set up, manage, administer, and maintain output in M3. The layouts that can be used by M3 users for document and report distribution are available through these products:

Product	Description
Infor Document Management	Mandatory.
	Included in Infor OS and is connected to M3 CE Core through a configuration of M3 Business Engine programs.
	For information how to use and connect IDM to M3 CE Core, see Configuring Infor Document Management with M3 CE Core on page 88.
OpenText Exstream or OpenText Cloud	Optional.
	Is a third-party product and can be used independently in parallel with Infor Document Management.
	For information how to use this product, see Using OpenText Exstream for M3 CE Core on page 21.

Configuring Infor Document Management with M3 CE Core

Infor Document Management and M3 CE Core are integrated by M3 CE Core using Infor Document Management APIs. The integration supports these output media types:

- Printing and sending the document in PDF format by e-mail.
- Saving the PDF document in Infor Document Management.

Infor Document Management and M3 CE Core are integrated through these methods:

- ION API (REST enabled)
- Context Business Messages

- BOD nouns

  The CaptureDocument BOD is an outbound BOD that is sent by Document Management when capturing different types of document, for example, scanned documents.

  The ContentDocument BOD is both an outbound BOD and inbound BOD. M3 CE Core creates documents in Document Management by sending a ProcessContentDocument BOD. M3 CE Core receives updates to documents within Document Management by receiving SyncContentDocument BODs.

M3 CE Core delivers its own document models and set of links to the UI applications (if applicable).

Document models and links can be configured for a tenant.

M3-specific content for Infor Document Management

A content pack is provided by M3 and automatically installed in Infor Document Management. It contains this information:

Category	Content	Description
Document Type	`M3 Layout Template`	This document type is used for layout templates that are delivered by M3. See [KB 1974263](#) for the complete list of printer files.
Document Type	`M3 Layout`	Default document type that is used for layouts that are designed by customers.
Document Type	A set of files named `M3_ xxxxxx`, for example `Cus tomer Invoice` or `Suppli er Invoice`	Default document type per type of generated business documents.
Microsoft Word files	A set of files named *ABC123* `PF.docx`, for example `OIS199PF Invoice.docx` or `MMS480PF Delivery Note. docx`	Microsoft Word files with layouts that are used for document design. Each printer file has a corresponding word file.
Access control list	M3_LayoutTemplate	Access control list that is used for Word documents.
Business Context Model	A set of entities, for example *InforCustomerInvoice* or *InforSupplierInvoice*	Business context models that are used by related information in H5. There are several files.

Configuring M3 Business Engine programs for output management

To connect M3 Core to Infor Document Management, you must configure a set of Infor M3 Business Engine programs in Infor M3.

1 Create a standard configuration for output management.

    a   Open `'List and Printer programs. Configure' (CMS005)`.

    b   Select **3 - Printer files**.

         All printer files that are enabled for XML are displayed.

    c   Select **ACTION > Gen standard F14** to generate a standard configuration for printer files and basic data programs.

    d   Select these check boxes and click **Next**.

         Printer files
         Generates standard output for all printer files.

         Basic data for Output Management
         Generates standard basic data for output management. This data is shared between all companies. These programs are updated:

         `'Output Service Type. Open' (MNS217)`
         Service type M3_XML is added. The interface is set to **`XML`**.

         `'Output Service Type. Connect Media' (MNS214)`
         Records with the service type M3_XML are added for MAIL, PRT and ARCHIVE.

         `'Document Type. Open' (MNS060`)
         These document types are added: M3_TEMPLATE_DEF, M3DEF

         `'Output Service. Open' (MNS216)`
         Service ID M3_IDM is added

         Output selection based on user
         Generates standard output based on user.

         `'Output Service Selection. Open' (MNS204)`
         In the program where the output service is connected to a printer file and user, a record is added for Blank printer file and Blank User. Services ID is set to M3_IDM. New records are added for each division in the current company. Existing records, if any, are preserved as is.

2 Open `'Output Service Selection. Open' (MNS204)` and ensure that all combinations of printer file and user applicable for Infor Document Management output management are connected to Service ID **`M3_IDM`**.

3 Open `'Output Media Selection. Open' (MNS205)` and ensure that all combinations of printer file and user applicable for Infor Document Management output management are connected to MAIL, *PRT or ARCHIVE.

Modifying layouts

You can modify the layout for a printer file by creating a new Word file, using the files delivered as M3 Layout Templates as the starting point.

Note: Always save the layout as `M3 Layout` Document Type before you start editing (or any other document type). Any file modifications that are made for the `M3 Layout Template` Document Type are replaced when an updated template is imported.

1   In Document Management, click **Control Center** and select **Development > Downloads**.
2   Click **Output Management Word Add-in**.
3   Download and install `IDMOutMgmtWordAddinInstaller.msi`.
4   In Document Management, click **Add document**.
5   In **Document Type**, select `M3 Layout Template`.
6   In **Templates**, select the applicable file. Set **New Document Type** to `M3_Layout`, and click **Create**.
7   Specify the Attributes and click **Save**.
8   Click **Check Out**.
9   Click the Word file and use the Microsoft Word add-in to modify the layout.
10 When the edits are completed, save the document in Word, and click **Check In** in **Document Management**.

Handling updates of IDM content for M3 output management

New or updated Infor Document Management (IDM) content for M3 output management is downloaded from the Cloud File Transfer Agent Administration page. Word layouts will be added to the document type `M3 Layout Templates`.

See KB 1974263, attachment `M3 Reports and Documents with Configurable XML_9999-99-99.xlsx` (where 9999-99-99 is the last update date) for details about which printer files are included.

1   In Infor M3, select **Administration Tools > Cloud File Transfer Agent Administration**.
2   Click **Download Contents**.
3   Select the **M3BE IDM Content**.
4   Click **Download**. A .zip file with the content is downloaded to your PC or server.
5   Extract the `content.zip` file.
6   Extract each of the `M3BE_xxx_M3BE_IDM_CONTENT.zip` .zip files, where xxx stands for the delivery unit. MVX is the default.
7   Import each of the `LayoutTemplates_y.xml` files into Infor Document Management. The xml files are split into several files due to size restrictions in Document Management. See the Excel file in KB 1974263 for details about which printer files each xml file contains.

   Note: To open the import tool in Infor Document Management, click **Control Center** and select **Administration > Import/Export**. Make sure that your user has an IDM role that allows `Create` (in the ACL role list) for the document type `M3 layout templates`.

IDM Enterprise Printing service

IDM Enterprise Printing service uses the ION Enterprise Connector to enable integration with M3 and on-premises printers.

See administration and configuration information in the *Infor Document Management Administration Guide Cloud Edition.*

When adding a printer to the print service in IDM, you must set the printer ID similar to the value in the 'Printer ID' field in `'Printer. Open'` (CRS290).

Chapter 8: M3 Function Search administration tasks

Use the M3 Function Search Administration management page to review the crawl status of the indexes. Usually, the indexes are created during tenant deployment, and indexed data is kept up to date automatically. You can re-crawl selected indexes if required. A re-crawl is required when a test environment is refreshed with new data through the data copy tool. A crawl is required if new M3 tables should be indexed and searchable through the M3 function search.

You can crawl up to 10 tables simultaneously. Avoid performing crawls during work hours, and while M3 is active with MRPs or other scheduled activities.

Accessing M3 Function Search

1 Go to Infor M3 menu in Infor Ming.le.

2 Select **Administration Tools > M3 Function Search Administration**.

   Note: The IFS Security role *M3ENTERPRISESEARCH-Administrator* is required to access the **M3 Function Search Administration** page.

Admin UI update overview

To modernize the underlying technology for M3 Function Search, an updated version is gradually rolled out during the second half of year 2021. Included in the update is the Admin UI change that follows the latest Infor UI standards.

The update introduces these features:

- New search or filter fields are now above the list, while the current Admin UI only has a simple table without any search or filter fields above the list.
- A new feature is also available that deactivates an indexed table.

   Note: There are no action needed from customers.

Information about the M3 Function Search update is also found in the *M3 CE Early Announcement* KB document. For a short time period that the legacy and updated Admin UIs exists in parallel, a Tenant

with an updated M3 Function Search automatically gets the updated Admin UI. While the functionality is the same, minor differences are present.

For more details, see

Legacy Admin UI

This section refers to the configurations applicable to the legacy admin UI of M3 Function Search.

Updating table indexes

1   Select **Administration Tools > M3 Function Search Administration**.

2   Select the **Index** check boxes for the relevant tables.

3   Click **Crawl Selected** to update the indexes.

   Note: Updating several indexes can be a time-consuming process, depending on the size of the tables. You can crawl up to 10 tables simultaneously.

4   Click **Refresh** to show the latest status. You can do this while a crawl is ongoing.

Viewing and changing the log level for M3 Function Search

Use the M3 Function Search Administration management page to view and change the log level for search functionality.

1   Select **Administration Tools > M3 Function Search Administration**.

2   Click the Search Logs link.

   Optional steps in M3 Function Search Administration:

   • Filter log text

      Specify a description or a term that you want to filter in the filter box, for example, your M3 user ID.

      When you press the Enter key, it shows only the entries that match the specified description or term.

      To clear the filter, clear the filter box and click **Enter**.

   • Download the log file

      Click **Download**.

      The browser downloads the log file with the default name m3fs*<current-date>-<current-time>*.

   • Refresh

      Click **Refresh** to reload the log from the server.

- Enable and stop Debug or Trace logging

   Select a log level to enable or stop. You may select between **Debug** and **Trace**.

   The selected log level is enabled.

   To stop selected logging, clear the check box.

   Note: Clear the Debug or Trace logging after inspecting is finished. Low-level logging can be resource-consuming on the server side.

Updated Admin UI

This section refers to the configurations applicable to the updated admin UI of M3 Function Search.

Updating table indexes

1 Select **Administration Tools > M3 Function Search Administration**.

2 Select relevant tables by selecting the table check boxes.

3 To make the tables searchable, click **Activate table(s)**. Alternatively, to make the tables unsearchable, click **Deactivate table(s)**.

   Note: Updating the searchable and unsearchable functions of several tables can be a time-consuming process depending on the size and number of selected tables.

4 Click **Refresh** to show the latest status. You can do this while an activation is ongoing.

Configuring logging for M3 Function Search

For more details about configuring the logging related to the new search functionality, see Viewing and configuring logging on page 376.

Chapter 9: M3 H5 administration tasks

M3 H5 provides an HTML 5-based user interface for accessing work-related information and applications from the M3 business system. It enables the user to run M3 programs under predefined workflows.

The IFS Security role *M3UI-Administrator* is required to access the **H5 Administration** page.

H5 Enterprise contains these features:

- Personalizations
- Shortcuts
- Enabled saving of Context Publisher settings
- Web mashups
- Multi-web browser support

Logging in through a web browser

1   Open a web browser.

2   Log on to Infor Ming.le as tenant administrator.

3   Click **Infor M3**.

4   Expand **Administration Tools** and select **H5 Administration**.

Link Manager

You must create links to associate information to a program. An administrator can link web pages or documents from Infor Document Management.

Creating a link to documents in Infor Document Management

Before you start, ensure that IDM is installed and properly configured in Infor Ming.le.

1   Open the program from which the link must be created.

2   Select **Tools** > **Link Manager**.

3   In the **New Link, step 1 of 2** window, specify this information:

Title

Specify a title for the link that you are creating.

Target

Select **Document Archive** from the list.

4   Click **Next**.

5   On the **New Link, step 2 of 2** window, specify this information:

Document Types

Select the type of document to which the program must be linked. The values in the list are obtained from the Document Management Entity URL source.

Attributes

Select the attribute of the document type. The values in the list are obtained from the Document Management Entity URL source.

Operators

Select "AND" or "OR" as the operator.

- Attributes within the same document type have an "AND" relation. The link finds the documents of the specified type that match all the selected attribute criteria.

- Different document types within the same link have an "OR" relation. The link matches documents of either type that meet all the criteria of the attribute.

Value

To map the attribute to a variable, select this option and specify the field name.

To map the attribute to a specific value, select this option and specify the value in the input field. Select the **static** option.

M3 Transactions

To map the attribute to a variable, select this option and select a field from the list.

6   Click **Add**. The new link is added under Summary Link Relations.

7   To remove all created links, click **Clear**.

To remove a specific link, click the delete icon for that link.

8   Click **Finish**.

In the **Link Manager** window, verify that the created link is added to the list.

Creating web links

1 Open the program from which the link must be created.

2 Select **Tools > Link Manager**.

3 In the **New Link, step 1 of 2** window, specify this information:

Title

Specify a title for the link that you are creating.

Target

Select **Web** from the list.

4 Click **Next**.

5 In the **New Link, step 2 of 2** window, specify this information:

Url

Specify the web address to which the program must be linked.

M3 Transactions

Select a field to be used as the search string and click **Add**. The selected field is added as a key value in the URL.

New Window

Select the check box if you want to open the link in a new window.

Some websites does not open in a separate frame, for example, Google. In such cases, you must select **New Window** to force web sites to be displayed in a separate window.

6 Click **Finish**.

In the **Link Manager** window, verify that the new link is added to the list.

Editing a link

1 Open a program.

2 Select **Tools > Link Manager**. Close the **Link Properties** window.

3 In **Link Manager**, select the link to edit. Click **Edit**.

4 Edit the necessary changes to the properties.

5 Click **Finish**.

Deleting a link

1 Open a program.

2 Select **Tools > Link Manager**. Close the **Link Properties** window.

3   In the **Link Manager** window, select the link to remove. Click **Delete**.

4   Click **Yes** to confirm deletion.

Copying a link

1   Open a program.

2   Select **Tools > Link Manager**. Close the **Link Properties** window.

3   In the **Link Manager** window, select the link to copy. Click **Copy**.

4   Edit the necessary changes.

5   Click **Finish**.

Using the H5 Settings tool

You can make special settings assignments based on user roles by using the H5 Settings tool. These special assignments are called rules.

For example, if you want only the M3 managers to make personalizations, you can disable the setting by default and then enable it only for those specific users. Or, if your site has many users within a role who will make personalizations, you can enable the setting for a group of users through their role.

In H5, user settings can be changed by users in the **Tools > User settings**.

Application settings can be changed only by an administrator.

Changing a default setting

Changing the default settings only affects new users. This table shows the effect of changing the default setting in different scenarios:

User	Effect
New users	New users have the new default or the default from the first rule match.
Existing users who have not saved any settings	These users get the new default. (This would be possible to implement on a per settings basis.)
Existing users who have not changed their settings and are members of a rule	The rule gives the default setting.
Existing users who have changed their setting (or any other setting)	No effect.

User	Effect
Existing users who have changed their settings and are members of a rule	No effect.

1 From the application menu, select **Administration Tools > H5 Administration > H5 Settings tab**.

2 In the User settings section, enable or disable these settings:

To lock a setting so that users cannot update it, toggle the Lock setting for user button that corresponds to that setting. When a check mark is displayed, a setting is locked.

Place cursor at end of fields
If enabled, this setting places a cursor at the end of the value in an input field. If disabled, the entire value in the input field is highlighted.

Default value: disabled

Expand list
If a program supports expandable rows of lists and this setting is disabled, the expandable content is displayed in list columns.

Default value: enabled

Default basic list option
This setting sets the default action for an item in the list of an M3 program.

Default value: 5

Valid values: 1, 2, 3, 4, and 5

1 Select, 2 Change, 3 Copy, 4 Delete, 5 Display (read only)

Program options
This setting sets a default action after opening a record in a specific M3 program.

Specify a list of key-value pairs when setting default options for several programs. Use this list syntax:

program=option;*program=option*

For example: OIS100=1;MMS100=5.

Show messages in dialogs
If enabled, this setting shows messages from the M3 server in a dialog window instead of on the status bar. Messages are displayed on the status bar by default.

Default: disabled

Use plus key as tab
If enabled, this setting allows the use of the plus key (+) to navigate across the M3 panel as an alternative to the tab key.

Default: enabled

Right align labels
If this setting is enabled, field labels are aligned to the right.

Default value: disabled

Show field help on mouse over
If enabled, this setting shows a help tooltip on mouse over.

Default value: disabled

Show label tooltips on mouse over
If enabled, this setting shows the full description of column header labels.

Default value: disabled

Show company and division in status bar
If enabled, this setting shows the company and division information on the status bar.

Default value: enabled

Use small panel navigator
If enabled, this setting shows the panel navigator in mini-mode.

Default value: enabled

Show program identifier on tabs
If enabled, this setting shows the program ID beside the program name.

Default: disabled

Use Infor Ming.le Language
If enabled, this setting allows H5 to display content based on the language used by Infor Ming.le.

If selected Infor Mingle Language is not supported by M3, the H5 client will use English as the default language.

See information about changing the site default language in *Infor Ming.le User Guide*.

Default: enabled

Theme
This setting changes the color of H5. The theme is only applied on the top part of H5.
Note: This setting is also available in the new H5 UI but the colors for the theme may vary. When you select this setting for the new H5 UI, the theme is both applied on the top part of H5 and the application menu.

Auto-switch to Homepages
When enabled, H5 will automatically switch to Homepages when all programs in H5 are closed.

Default: disabled

Activate context help
If enabled, this setting causes the field context details to be published when you select and move your pointer to a particular field or press F1.

Default: disabled

Calendar view

Use this setting to select the calendar format displayed when using the date picker tool.

Note: The default format may vary. Gregorian is the default value for most supported languages, whereas for some languages the default format might be different.

Form Size
This setting changes the size of the fields and buttons.
Note: This is only available in the new H5 UI.

Default: Compact

Row Size
This setting changes the row height of the data grid.
Note: This is only available in the new H5 UI.

Default: Compact

Enable Classic UI
This setting switches the new UI to the classic H5.

For existing tenants, default value is enabled while for new tenants, default value is disabled. If disabled, the new generation UI will replace the classic H5.

Enable browser autofill
This setting automatically fills the text fields using saved input. This setting overrides the Enable Favorites personalization feature.
Note: Ensure that the 'Save and fill addresses' setting in Chrome is also enabled to make the autofill work.

This is only available in the new H5 UI.

Default: disabled

3   In the Application settings section, enable or disable these settings:

Expand filter options
If enabled, this setting expands the Filter Options for applicable programs regardless of its last state (expanded or collapsed). When disabled, the last state of the Filter Options is saved.

Default value: disabled

Enable Excel export
If this setting is enabled, users can use the Export to Excel function.

Default value: enabled

Enable Link Manager
If this setting is enabled, users can use the Link Manager function. For administrators, the Link Manager is always available.

Default value: disabled

Enable automatic publishing of Infor Business Context in Ming.le
If enabled, Infor Business Context messages are published on all panels running in Infor Ming.le. This is done through the Infor Business Context by filling up the entities to be published.

Default value: enabled

Enable Google Sheets Export
If this setting is enabled, users can use the Export to Google Sheets function.

Default value: disabled

Google Sheets API key
This field must contain the API key that enables users to access and view data in Google sheets.

See information about generating the API key in [Enabling Google Sheets API](#) on page 110.

Enable Pendo
If enabled, this setting allows H5 to display in-context support through the Pendo in-app guides.

Default value: disabled

Enable the personalization tools
If this setting is enabled, users can access the personalization tools.
Note: The cache on the server is cleared automatically when personalization tools are enabled. Running clients must be signed in after loading the updated personalization.

Default value: enabled

Enable conditional styles
If enabled, users can create conditional style personalizations.

Default value: enabled

Enable favorites
If enabled, users can create favorite personalizations.
Note: This setting applies to the input field-based favorites, which are stored values in specific fields or also known as textbox history.

Default value: enabled

Enable hidden field
If enabled, users can create hidden field personalizations.

Default value: enabled

Enable hyperlinks
If enabled, users can create hyperlink personalizations.

Default value: enabled

Enable labels
If enabled, users can create label personalizations.

Default value: enabled

Enable mandatory fields
If enabled, users can create mandatory field personalizations.

Default value: enabled

Enable scripts
If enabled, users can create scripts personalizations.

Default value: enabled

Enable saved search queries
If enabled, users can save search queries personalizations.

Default value: enabled

Enable shortcuts

If enabled, users can create shortcut personalizations.

Default value: enabled

Enable tab order

If enabled, users can create tab order personalizations.

Default value: enabled

Enable view

If enabled, users can create view personalizations.

Default value: enabled

Enable use of favorite name for MForms

If enabled, the defined name will be used by the program launched from the Favorites list.

Default value: disabled

4   Click **Save**.

Adding a rule

A rule is a setting customization that you make that determines the access of a user or a group of users to features when you want that access to be different from the default. For example, if you want only certain users to create personalizations, your default setting for the option **Enable the personalization tools** would be disabled. By default, users cannot make personalization if the setting is disabled. Then, create a rule that can turn on **Enable the personalization tools**, that is, personalizations would be enabled for those users.

1   From the application menu, select **Administration Tools > H5 Administration > H5 Settings tab**.
2   Select the setting to be changed and click the **Browse** button.
3   Click **Add rule**.
4   Specify a name for the new rule. Use a name that describes the purpose of the rule. For example, if you are creating a rule to enable M3 managers to change personalizations, you can use `Enable user personalizations for M3 managers`.
5   Toggle the **Value** button to enable or disable the setting.
6   To add users individually, specify a User ID and click **Add**.

To add users by role, specify a Group ID, or click the group selector on the right corner of the Group ID field and select a group. Click **Add**.

Users or groups that are added are displayed in the box. To remove users or groups from the box, select the users and click **Remove**.

7   Click **OK**.
8   Click **Save**.

Changing a setting for selected users and roles

1 From the application menu, select **Administration Tools > H5 Administration > H5 Settings tab**.

2 Select the setting to be changed and click the **Browse** button.

3 From the Rules list, select the rules to be changed.

4 Click the **Settings** button. Optionally, click the **Delete** button to remove the rule from the list.

5 Change the settings in the **Edit rule** dialog box.

6 Click **OK**.

7 Click **Save**.

Using the Personalizations tool

The Personalizations tool can be used by H5 administrators to manage personalizations (customizations) to M3 forms that were created by H5 users. From the application menu, select **Administration Tools > H5 Administration > Personalizations tab**.

The tool has different personalization views:

- Global

- Role

- User

- Program

You drill down starting at any of these views. When you reach a specific panel, it is possible to see what types of personalizations exist on a panel and how many of that type. Examples of personalization types are hyperlinks, conditional styles, context publisher, and replacement texts. There is no preview, but if you drill down on a specific type, a dialog with the definition XML is displayed.

Copy to vs. Merge to

It is important to know how personalizations are copied from a source to a target in the tool. A personalization entity is the panel for the personalization and the personalization type. This is the smallest entity that you can manage in the tool. It is not possible to manage, for example, different conditional styles within a panel.

In the tool, you can select **Copy to** or **Merge to** to apply a personalization from a source to a target. **Copy to** is only enabled for program, user, and role. Merge to is only allowed for programs and personalization types on a specific panel.

Use **Copy to** if you are copying all personalizations from a source to a target, which removes any existing personalization that the target has. The Copy to function applies the same personalization that the source has to the target.

Merge to compares and only overwrites those personalization types that the source has to the target. If the target already has hyperlinks and you merge from a source that has a context publisher (and no other personalization exists), the target gets the context publisher but still keep the hyperlinks. There is no merge within a type. If both source and target have hyperlinks, the hyperlinks in the target is completely replaced by those in the source. To handle that level of complexity, you must edit the XML through export/import manually.

If you copy personalization from, for example, one user (source) to a role (target), any existing personalization of the same type in the target will be overwritten, not merged.

Personalization lookup and priorities

The M3 UI Adapter server uses different modes for loading personalization depending on the settings for the current user. The modes are User, Central, and None:

- None: No search for personalizations is done. Used when personalizations are disabled globally or for the user.

- Central: Role and global personalizations are searched. If role personalizations are found, the Central mode will use these personalizations first. Otherwise, it will search for global personalizations to use.

- User: Does the same search as the Central mode, but also searches the user personalization. There can be two hits if the same program has both a user and central personalization.

The result from the personalization search can be zero results, or one or two personalization files, and the content of these are embedded in the response to the client. If the client receives one central and one user personalization, it will merge the two on a panel level. User personalization has priority over central personalization.

Exporting a personalization file

1   Select a user, role, or program from the list.

2   Click **Export** to export the personalization XML.

The file is automatically downloaded in the browser.

Export is available for a program, user, and role. If only one program is selected, an XML file with the personalization will be exported. If a user is selected, the exported file is a .zip with all the personalizations.

Importing a personalization file

1 Click **Import**.

2 In the Import personalization file dialog box, specify the path to the XML or the zip file that contains one or multiple personalization files.

   If the target does not exist, you can add a user or a role by clicking **Add** and specifying a name.

3 Select the target and click **OK**.

Removing a file

1 Select a personalization, role, or user from the list.

2 Click **Delete**.

Using the Data files tool

You can use the Data files tool to manage files in H5.

1 From the application menu, select **Administration Tools > H5 Administration > Data files tab**.

2 Specify the file type by selecting from the Change file type list. You can use the tool to manage these types of data files:

   • Automation Template

   • Configuration

   • Document Link

   • Favorite

   • H5 Script

   • MForms Script

   You can filter or sort the files by using the Name, Description (if one exists), Change Date, Changed By, or Created By information.

3 Select files from the list.

4 Use these buttons to perform specific actions:

Button	Description
Export	Exports selected files.
	If you select a single data file, it will be exported as files with extensions such as .json, .js, .dll, or .xml.
	If you select multiple files, the files will be combined and exported as a .zip package.

Button	Description
Import	Imports a single data file. **a** Browse the file to import. **b** Optionally, you can add a description for the file being imported. **c** Click **OK** when done. **Note:** A file with a .dll extension is not supported by this function.
Import Zip	Imports a zip package containing multiple files. Browse the .zip package to import and click **Upload**. **Note:** Files with .dll extensions must be placed in a zip first before you import.
Delete	Removes files.
Description	Adds or updates the description of a file. The description is not included when exporting a file.

Creating custom context applications

1 Click the **Data files** tab in administration tools and select the Configuration file type.

2 Select the ContextApplicationsConfiguration.json file and click **Export**.

The ContextApplicationsConfiguration.json file is used to manage the Context Messages that you can publish in H5 Context Publisher. The file contains an array of JSON objects. Each JSON Object should contain these elements:

• uid: A unique ID (any unique value) for the Context Message.

• title: The menu item that will be displayed in the Context Publisher Sub-Menu (any value).

• type: The message type for the Context Message. This type should match the expected message type of your Mingle Context Application.

• data: The data template for the Context Message. This template should match the expected data structure of your Infor Mingle Context Application.

See this sample of a ContextApplicationsConfiguration.json file:

```
[
        {
                "uid" : "M3_BC",
                "title" : "Currency Converter",
                "type" : "currencyExchange",
                "data" : {
                                "baseCurrency" : "USD",
                                 "baseValue" : "1",
                                 "targetCurrency" : "EUR"
                }
        },
```

```
          . . . . . .

    ]
```

Note: Do not change or remove the entry with the uid=M3_BC because it has a special function in the Context Publisher.

", ..." indicates where you can add other valid JSON array of objects. If there is only one object in the JSON array, you should remove this.

3   Open and edit the file by using a text editor.

     We recommend that you use a text editor that supports json and json formatting.

4   Validate the son format in the editor, if possible. You can use Notepad ++ with JSON Viewer Plugin which is an open source free tool for Windows.

5   Return to Data files and select Configuration category.

6   Import the file that must be named ContextApplicationsConfiguration.json.

     The existing file is replaced.

     Note: The file name must be ContextApplicationsConfiguration.json. All other files are ignored in this Configuration category.

Using the export/import tool

The purpose of the export/import tool is to quickly move all data from test to production or from one server to another. The export tool creates a .zip file that contains data from the database, and then the import tool imports that data. The data category that is imported overwrites any existing data.

Note: We recommend to use the export/import tool only during a service window.

Data category

You can export/import different data types:

Data category	Description
Applications	This setting includes installed web applications that can be found in the H5 client. These applications are built by using the Mashup SDK that are installed on Application tab in the Administration tool.
Automation templates	These templates are XML files for automation that you can use with scripts to simulate passing values and key strokes to an M3 program.
Document links	This setting is used for document links for H5. Note that these are not the same as Smart Office.

Data category	Description
Favorites	This setting contains user favorites for H5.
H5 Personalizations	This setting covers personalization files for roles and users.
H5 Scripts	This setting includes JavaScripts for H5 client.
MForms Scripts	This setting includes JScript for MForms in Smart Office.
User data	This user data setting is currently for shortcuts only.
User settings	This setting includes user settings for H5.

Exporting data

1 From the application menu, select **Administration Tools > H5 Administration > Export tab**.

2 Select the data category to export.

3 Click the **Export** button.

4 When the operation is successful, click **Download**.

A copy of the .zip file will be downloaded to your local machine.

Importing data

1 From the application menu, select **Administration Tools > H5 Administration > Import tab**.

2 Locate the .zip file to import and click **Upload**.

A copy of the .zip file will be uploaded to the Import directory.

3 Select the data category to import. If you select categories that are not available in the .zip file, the output log will have error messages.

4 Click the **Import** button.

5 When the import is successful, close the **H5 Administration** tab.

6 Verify the imported data.

Note: Restart the browser if you opened an M3 program affected by the import.

Enabling Google Sheets API

You can visit https://developers.google.com/sheets/api/quickstart/js to validate if there are any changes in the steps indicated in this procedure.

1 Use this wizard https://console.developers.google.com/start/api?id=sheets.googleapis.com to create or select a project in the Google Developers Console and automatically turn on the API. Click **Continue**, and then click **Go to credentials**.

2 On the Add credentials of your project page, click **Cancel**.

3 At the top of the page, select the **OAuth consent screen** tab. Specify an email address and a product name. Specify the URL in the **Privacy Policy** field. Click **Save**.

  Due to a new policy by Google, a Privacy Policy is required to have the API access verified. This Privacy Policy should be hosted within the domain of the H5 environment you are using.

4 On the **Credentials** tab, click **Create Credentials** and select OAuth client ID.

5 Select `Web application` for the application type.

6 In the **Authorized JavaScript origins** field, specify the URL used for deployment.

  For example, *https://m3.server.com*.

  Optionally, leave the Authorized redirect URL field blank.

7 Click **Create**.

  Note: The client ID in the resulting dialog box will be used in the Google Sheets API Key field in application settings. Take note of the project ID too. This is for the verification of the OAuth Client to be used in H5 Export to Google Sheets feature. You can download a JSON file containing this information by clicking the Project Name of your new project, then clicking the **DOWNLOAD JSON** link.

8 Click **OK** to dismiss the resulting dialog.

  To prevent encountering the "unverified app" screen and having a limit to the number of users, clients can request for verification. Ensure that you have these requirements for app verification:

- The page describing your app's privacy policy must be hosted within the domain.
- You must have verified domain ownership.

  To check if you meet this requirement, go to https://support.google.com/webmasters/answer/35179?hl=en.

  After you completed the requirements, you can request verification of the app's API use:

 a Take note of the used scopes for Export to Google Sheets: https://www.googleapis.com/auth/spreadsheets.readonly and https://www.googleapis.com/auth/spreadsheets.

 b Specify the Privacy Policy URL on the OAuth Consent Tab.

 c Take note of the Client ID.

 d In the Project settings, take note of the Project ID.

 e File a review request of the project's OAuth.

  To access support, go to https://support.google.com/code/contact/oauth_app_verification

Using the Applications tool

The applications tool is for deploying web applications build that includes Mashup SDK. The applications are installed on the M3 UI Adapter server and are publicly accessible.

Installing a web application

1 Create a zip file that contains script and HTML files. The name of the zip file will be included in the path to the web application, so ensure that it is a valid path name.

   Note: The name of the zip file should be in lowercase and should not contain spaces. The name must also have this format: mne/apps/yourname/.

2 From the application menu, select **Administration Tools > H5 Administration > Applications tab**.

3 Click **Install** and browse to the zip file.

4 Click **Upload**.

5 Verify the application by clicking the link under the Installed applications.

   To uninstall an application, click **Uninstall** below the installed application.

Installing multiple web applications

1 Create a zip file that contains script and HTML files. Repeat this step for each application to install.

2 After creating all zip files, create another zip file that will contain all other zip files in one bundle. The name of the zip file will be the path for the web application, so ensure that it is a valid path name.

   Note: The zip file should not contain the top folder, only the files. You must zip the files in a folder and not the folder itself.

3 From the application menu, select **Administration Tools > H5 Administration > Applications tab**.

4 Click **Install Zips** and browse to the zip file that created the bundle.

5 Click **Upload**.

6 Verify the applications by clicking the link under the Installed applications.

   To uninstall an application, click **Uninstall** below the installed application.

Chapter 10: BOD Processor administration tasks

The BOD outbound data flow is based on events that represent predefined data. The Event Hub is the central place where all events are directed to its subscribers. Events from M3 Foundation can be based on program or table operators while M3 BODs are based on table changes.

To trigger events from a specific database table, a rule in Event Analytics must be created. When a rule is activated and the rule session is started, the events that Event Analytics subscribe to are published by M3 Foundation.

M3 BODs has its own set of standard rules in Event Analytics. These standard rules are predefined to recognize changes for specific tables, fields, or values in M3 Business Engine. Event Analytics validates events for BOD and are published as M3BEBOD events, while events that are invalid are deleted.

The purpose of the BOD Processor administration is to configure which companies and divisions in M3 should publish BODs. The valid companies, divisions, facilities, and warehouses are retrieved from M3 through standard MI transactions.

The BOD Processor subscribes to events from Event Analytics, and enhances and publishes them through Event Hub to IEC.

To access BOD processor, select **Administration Tools > BOD Processor Administration** from the Infor M3 Menu in Infor Ming.le. The IFS Security role *M3EC-Administrator* is required to access the **BOD Processor Administration** page.

Enabling BODs

Note: You must manually enable each BOD per M3 Company and Division for M3 to publish outbound BODs.

1   Select **Administration Tools > BOD Processor Administration**.
2   In the **M3 data** tab, click each applicable company link to show the list of available divisions, and select check boxes as needed.
3   Click **Save**. Facility and Warehouse are displayed as reference to inform which Division they belong to.

In the **BODs** tab, you can enable and disable M3 BOD publishing per company and division. You can enable and disable company and divisions based on a BOD, or BODs based on a combination of company and division.

4   Select the check box for the applicable BOD or Division per Company to enable publishing, and clear the check box to disable publishing. Click **Save**.

Custom BODs naming format

You can add custom BODs as required. The custom BOD name must follow the format [Verb][Noun], for example *SyncItemHierarchy*, and must correspond to these characteristics:

- Both Verb and Noun must be specified.
- The supported verbs are "Show", "Sync, "Process", "Load", "Get" or "Update".
- The noun can be arbitrarily chosen.
- Blank spaces are not allowed.

Adding custom BODs

1   Select **Administration Tools > BOD Processor Administration**.

2   In the **BOD** tab, click **Customize**.

3   Specify the custom BOD name, and click **Add**.

4   Click **OK**. The new BOD is displayed in the BODs document list.

To delete a custom BOD, follow the same process and click **Delete** instead of **Add**.

Configuring M3 application settings for ProcessWorkflow

M3 can publish a ProcessWorkflow BOD that can start a workflow in ION.

The ProcessWorkflow BOD publishes this information:

- WorkflowDefinitionCode: This is the name of the workflow in ION that should be started.
- Property/NameValue: M3 data is published in the **/Property/NameValue** element in the BOD, with the M3 field name as attribute and the M3 field value as element. There are two different sets of data:
  - Data that is related to the application message (CMAILB M3 table)
    - Message ID (CBMLID)
    - Message receiver (CBREC2)
    - Message text (CBMSGT): The message text from M3 may be a maximum of 180 characters long, the NameValue element in the BOD a maximum of 64 characters. Therefore, the

message text from M3 is published in up to three different NameValue elements with the MSGT1, MSGT2 and MSGT3 attribute names.

- Data that is related to the M3 record that triggered the application message:
  - M3 publishes the key values for the record that triggered the application message.
    Note: For this set of M3 data, the field name is published without the M3 table prefix. For example, the field name of the item number is published as ITNO and not as MMITNO. This simplifies using the data in the ION workflow to call an M3 API.

Ad hoc reports

Configure these settings on `(AHS110/F)`:

- Ad hoc application message (AHAM): Specify either **2-Current User** or **3-Specific receiver**
- BOD enabled (BODE): Select the check box.
- BOD message type (BDMT): Specify **4-Workflow**
- Workflow name (WFNM): Specify the name of the workflow in ION that is to be started by this application message.

Configure this setting in `'M3 Settings - Application Messages' (CRS424)`:

- Activate Application Message Type 11

Event-based alerts

Set up a valid event subscription in `'M3 Event Subscription. Open' (CMS045)`.

Set up a valid event-based alert in `'M3 Event Based Alerts. Open' (CMS047)`.

Configure these settings on (CMS047/E):

- **Status (STAT)**: Specify **20-Approved**
- **BOD enabled (BODE)**: Select the check box.
- **BOD message type (BDMT)**: Specify **4-Workflow**
- **Workflow name (WFNM)**: Specify the name of the workflow in ION to be started by this application message.

Configure this setting in `'Settings - Application Messages' (CRS424)`:

- Specify 012 in the **Mtp** (message type) field, open this message and select the **Activity code** check box.

M3 application messages

Configure these settings in `'Settings. Application Messages' (CRS424)` for relevant application messages:

- Activity code: Select the check box
- BOD enabled (BODE): Select the check box
- BOD message type (BDMT): Specify `4-Workflow`
- Workflow name (WFNM): Specify the workflow name that should be started in ION.

Publishing BODs

Each BOD must be enabled in the BOD Processor for the relevant accounting entity before BODs can be published.

See

You publish BODs so that ION can retrieve them from your message outbox. You can publish BODs for general use by any product that subscribes to the BODs through Infor ION. After you perform an initial data load from your application, you verify that data is flowing out of the application into ION. For details about a specific integration, see the appropriate integration guide.

For an ION-based integration between this product and another product, see the appropriate integration guide for these instructions:

1   Setting up the data flows.
2   Performing any additional configuration of the applications that is required for that integration.

You can then use the steps in the integration guide to publish the BODs.

Master Accounting Entity

Master Accounting Entity (Master AE) functionality allows the publishing of one BOD for a company regardless of how many divisions the company has. Master AE is represented by the M3 company value, for example, 770. It is not the same as the M3 blank division in the company, which is a real accounting entity and is represented, for example, by 770_.

This functionality is introduced to limit the number of BODs generated. Without Master AE, one BOD per activated division will be published, but with Master AE, only one BOD per company will be published.

Currently, M3 supports Master AE for these Sync BODs:

- ItemMaster
- SupplierPartyMaster
- ShipFromPartyMaster
- RemitToPartyMaster

- CarrierParty

Master Accounting Entity for ItemMaster BOD

If the ItemMaster BOD is triggered with Master AE enabled, only one BOD will be generated. The AccountingEntityID and accountingEntity attributes in the ItemMasterHeader level are displayed as Company (CONO) only. ItemLocation details are displayed for all activated divisions with accountingEntity attribute displayed as Company and Division (DIVI) in CONO_DIVI format.

With Master AE disabled, the AccountingEntityID and accountingEntity attributes in ItemMasterHeader are displayed as CONO_DIVI and the number of BODs published depends on the number of divisions activated in the BOD Processor. ItemLocation accountingEntity is displayed as CONO_DIVI and this item warehouse data is only displayed if it belongs to the same division from which the BOD is published.

Master Accounting Entity for Supplier-related BODs

If supplier-related BODs namely SupplierPartyMaster, ShipFromPartyMaster, RemitToPartyMaster and CarrierParty are triggered with Master AE enabled, only one BOD for each noun will be generated. AccountingEntityID and accountingEntity attribute are displayed as Company (CONO) only. With Master AE disabled, the AccountingEntityID and accountingEntity attributes are displayed as CONO_DIVI, and the number of BODs published for each noun depends on the number of divisions activated in the BOD Processor.

Configuring M3 BODProcessor for Master Accounting Entity

BODProcessor has the BODs and Master AE tabs. The **Master AE** tab contains the check box for the BOD noun that can support the Master AE functionality. The **BOD** tab, on the other hand, has check boxes for divisions within the company to be selected or cleared for a specific BOD.

To enable the Master AE functionality:

1 Select the **Master AE** tab.
2 Select the check box for the BOD noun.
3 Click **Save**.
4 Click **Reload**
5 Click the **BODs** tab.
6 Select the BOD.
7 Click the company where the BOD is to be published.
8 Select the **MasterAE** check box.
9 Click **Save**.
10 Click **Reload**.

Once the check box for the BOD noun in the **Master AE** tab is enabled, the division check boxes for that BOD are not editable in the **BOD** tab. Whether the division is activated or not does not matter in this instance since only one BOD is published on the Master Accounting Entity.

Initial data load

During normal processing, BODs are published when a user changes data or completes a transaction. In some situations, when you start a new integration with another application, you must manually generate BODs that pass the current set of data from your application to another application, so that the base systems are synchronized. The BODs are placed in the message out-box, from which Infor ION retrieves them and passes them to the other application.

In Infor M3, the initial load can be started from `'Initial Load. Open' (EVS006)`:

- Open `'Initial Load. Open' (EVS006)` to start the initial load for a selected set of fields into a Sync or Show BOD.

M3 BOD dependencies

The outbound initial load for Customer-, BillTo-, PayFrom- and ShipToPartyMaster must be carried out in a certain sequence. This is because the BODs refer to each other:

- BillToPartyMaster refers to CustomerPartyMaster
- PayFromPartyMaster refers to BillToPartyMaster
- ShipToPartyMaster refers to CustomerPartyMaster

The correct sequence for sending the initial load is therefore:

1 CustomerPartyMaster (OCUSMA table)
2 BillToPartyMaster (OCUSMA table)
3 PayFromPartyMaster (OCUSMA table)
4 ShipToPartyMaster (OCUSAD table)

Note: This only applies to integrations that use one or more of the Customer-, BillTo-, PayFrom- and ShipToPartyMaster. See the details in the specific integration guides.

Starting an initial data load from (EVS006)

To perform an initial load of selected M3 data into Show or Sync BODs:

1 In M3, open `'Initial Load. Open' (EVS006)`.
2 In **BOD noun**, specify the noun of the BOD, for example **Location**.
3 Optional: In **BOD variant** specify the variant of the BOD noun, for example **Facility**.

4   In **Table**, specify the master table name, for example **CFACIL** to trigger
    M3BE_Out_SyncLocationFacility or M3BE_Out_ShowLocationFacility.

    See information about master tables for initial load in [Master tables for initial load](#) on page 415.

5   Click **Create**.

6   On (EVS006/E), specify the **BOD description**.

7   Optional: Define the **Selection** criteria. The fields that you specified in the selection can be used
    as filters when running the initial load event in `'Initial Load Job. Open'` (EVS007).

    Note: We recommend defining the selection to avoid triggering several events.

8   Click **Next**.

9   Select the `11-'Run'` option and specify these properties:

    Note: DIVI should be included in the selection when running division-specific records.

    •   Filter criteria for the selected tables and fields.
    •   **BOD verb**
    •   Only for Show BODs: **To logical ID**.

    For details on how to use 'Initial Load. Open' (EVS006), see "Initial Load" in *M3 Application
    Foundation User Guide* located in the *M3 Business Engine online documentation*.

10 Click **Next** to run the initial load events.

Chapter 11: Event Analytics Rules administration tasks

To access the **Event Analytics Rules** page, select **Administration Tools > Event Analytics Rules** from the Infor M3 Menu in Infor Ming.le. The *EVENTHUB-Administrator* IFS Security role is required to access the **Event Analytic Rules** page.

Event Analytics Rules UI overview

The Event Analytics Rules is a user interface where you can create Event Analytics Rules based on Event Hub Events. This rule is used when you filter out a specific event and resend it for processing.

Event Analytics Rules is divided into these panels: the Session Tree and the Editor panel.

Session tree

The session tree is composed of several icons that is used for managing sessions:

Session tree icons	Description
Refresh session tree	Refreshes the session tree and returns the sessions to its default
Export sessions	Exports custom sessions to a rulespack file
Import sessions	Imports custom sessions from a rulespack file
Add session	Adds a new session in the tree
Session filter	Filters the session tree and the rules under it

Started filter	• Enabled (green)
	  Shows the started sessions
	
	• Disabled (grey)
	  Hides the started sessions

Session tree icons	Description

Stopped filter	• Enabled (red)
	    Shows the stopped sessions
	• Disabled (grey)
	    Hides the stopped sessions

Filter field	Filters the tree by session name or rule name. If the specified search word matches a session name, only the session is displayed. Select the session to view its rules. If the specified search word matches a rule name, the rule and the session name it belongs to is displayed.

Session editor panel

The session editor shows these icons: Save, Refresh, Delete, Duplicate, Export, Add Rule, Start/Restart, and Stop. It also shows the **Name** and **Description** field. Below the **Description** field are the status and custom indicator. The session status indicator shows whether the session is Started or Stopped.

Note: You cannot find the custom indicator in a standard session. You cannot change the name of an already saved session.

Editor panel icons	Description
Save	Saves the changes in the session editor
Refresh session	Refreshes the session and shows the latest session updates from the server
Delete	Deletes a session **Note:** This icon is enabled for a session which is not started and is disabled for an already started session.
Duplicate	Duplicates a session
Export	Exports the session to a rulespack file
Add rule	Adds a new rule to a session
Start session	Starts a stopped session
Restart session	Restarts a session which is already started
Stop session	Stops a started session

Standard sessions

Infor product development delivers standard sessions. When you duplicate a standard session, it creates a new custom session.

All rules within a standard session are standard rules. These rules are read-only.

Note: You can start and stop standard sessions, but you cannot edit, delete, export, or import them.

Custom sessions

When you add or duplicate a session, it creates a custom session. In the session tree panel, a pencil icon after a session name indicates that the session is a custom session. In the Session editor panel, a custom indicator is displayed under **Description** field. You can edit, delete, duplicate, export, import, and start and stop a custom session.

All rules within a custom session are custom rules. You can edit, delete, duplicate, validate, and activate custom rules.

Rule editor panel

The rule editor shows these icons: Save, Refresh, Delete, Duplicate, Validate, Activate/Reactivate, and Deactivate. It also consists of these tabs: **General Information**, **Subscribed Event**, **Filtering**, **Published Event**, and **Fields**.

Rule editor panel icons	Description
Save	Saves the changes made in the rule editor. When clicked, the Notification is updated stating that the rule is saved successfully.
Refresh	Refreshes the rule to reflect the latest updates from the server
Delete	Deletes a rule **Note:** This icon is enabled for an inactive rule and disabled for an active rule.
Duplicate	Duplicates the selected rule
Validate	Validates the rule. When clicked, the Notification is updated with the validation status. **Note:** You can validate a rule that is not yet saved.
Activate	Activates a valid rule that is currently Inactive

Rule editor panel icons	Description
Reactivate	Activates a valid rule that is currently Active
	Note: You can simultaneously reactivate several edited rules under one session by restarting the session instead of reactivating each individual rule.
Deactivate	Deactivates an active rule

Note: The position of the vertical splitter between the session tree and the editor panel is retained from the previous session.

General Information tab

The purpose of this tab is to manage general data for the rule.

The General Information tab consists of **Name**, **Description**, **Revision**, and **Salience** fields.

Below the **Description** field are the status and custom indicators. The rule status indicator shows whether the rule is Active or Inactive. The Notification messages are displayed at the bottom of the tab. These messages indicate successful activities in blue and errors in red. Whenever a message is given, for example when you validate a rule, that current tab is automatically switched to this tab.

Note: You cannot find the custom indicator in a standard rule.

Fields	Description
Name	Name of the rule.
	You cannot change the name of a saved rule.
	Note: This is a required field.
Description	Description of the rule.
Revision	Revision of the rule.
	The revision number is automatically increased every time the rule is saved.
	This field is read-only.
Salience	Salience is a form of priority where rules with higher salience values are given higher priority within the session.
	For example, a rule with salience = 20 will be executed before a rule with salience = 10 if the rules are under the same session. Only positive integer values are accepted.
	Leave the field blank to disable salience.

Subscribed Event tab

The purpose of this tab is to describe the subscribed, or received, events. The **Subscribed Event**, **Filtering**, **Published Event**, and **Fields** tabs are laid out in a data-flow order.

The **Subscribed Event** tab consists of these groups:

• Subscription
• Field Metadata

This table shows the fields in the Subscription group:

Fields	Description
Publisher	The publisher name for the event subscription, for example "M3". This field is required.
Document	The document name for the event subscription, for example "MITMAS", for database events for the item master table in M3. This field is required.
Operations	The operation or operations for the event subscription. You must select at least one of the operations.

The Field Metadata group consists of a list of fields which are used for the subscribed events. You must create these fields manually.

These fields can be used in the field lists for ISCHANGED Filtering, Key Value, and Copied Fields.

See

Creating a new field in the list

1   In the field list, click **New**.
2   Specify a name for the field. The specified name must be unique within the list. This is a required field.
3   Specify the type from the list.
4   Click **Save**.

Editing a field in the list

1   In the field list, select the field to edit.
2   Click **Edit**. The **Edit Metadata Field** is displayed.
3   Edit the name of the field or change the type of the field.
4   Click **Save**.

Deleting a field from the list

1   In the field list, select the field to delete.

2   Click **Delete**.

3   Click **Yes** to confirm deletion of the field.
   Note: You cannot delete a field in use.

Filtering a field from the list

1   In the field list, specify a search string in the **Search** field.

   Only fields with matching names or types are displayed in the list. The search string only needs to match part of the field names. Filtering is not case sensitive.

2   Clear the **Search** field to show all fields in the list.

Filtering tab

The purpose of this tab is to filter received events. A received event must match all criteria to publish a new event. Event filtering and the publishing of new events, including virtual fields, are the main purposes of Event Analytics.

The Filtering tab consists of these groups:

*   Filtering
*   ISCHANGED Filtering

The Filtering group contains the field **Criteria**. You can specify an expression in this field. This expression must return a boolean value, and must be true to publish a new event. The field is a multi-line text area that is vertically resizable. Click the lower right corner to adjust the height of the text area.

The ISCHANGED Filtering group consists of a list of fields used with the ISCHANGED function. You can use this function for received events with the Update operation. Change at least one of the fields to publish a new event.

You can add ISCHANGED functions to the filtering criteria instead of using the field list.

For example:

```
STAT >= 20 AND STAT < 90 AND (ISCHANGED (field1) OR ISCHANGED (field2) OR
 ISCHANGED (field3)).
```

There can be several changed fields for a received event that must be checked, so this field list is added for convenience.

There is a logical AND operator between the expression in the field Criteria and this field list. There are logical OR operators between the fields in this list.

The fields must first be defined in the Field Metadata field list in the Subscribed Event tab before they can be added to this field list.

For more information about syntax and functions, see [Expressions](#) on page 130.

Adding a field to the list

1 In the field list, click **Add**.

2 Select the field to add. All fields created in the **Field Metadata** field list are displayed, except for fields that are already added to ISCHANGED Filtering.

3 Optionally, you can specify a search string in the **Search** field to filter the list.

4 Click **Add**.

 The selected field is added to the list.

Removing a field from the list

1 In the field list, select the field to remove and click **Remove**.

2 In the confirmation message, click **YES**.

 The field is removed from the list.

3 Optionally, you can add the field to the list again.

Filtering the list

1 In the Filtering field list, specify a search string in the **Search** field.

 Only fields with matching names are displayed in the list. The search string only needs to match part of the field names.

2 Clear the **Search** field to show all fields in the list.

Published Event tab

The purpose of this tab is to manage the published events. A received event must match all criteria given in the **Filtering** tab to publish a new event.

The Published Event tab consists of these groups:

• Event
• Key Value

This table shows the fields that are part of the Event group:

Fields	Description
Publisher	The publisher name is always "EventAnalytics". This field is read-only and is only displayed for clarity.

Fields	Description
Copy document name	Select this option if the document name for the published events must be copied from received events.
Document	The document name for the published events. This field is required if Copy document name is not selected. Otherwise, this field is disabled.
Copy operation	Select this option if the operation for the published events must be copied from the received events. If more than one operation is selected for the received events, published events may have different operations for the same rule.
Operations	Select the operation for the published events. You can only select one from these operations: Create, Update, Delete, Start, Exit, Fail, Request, and Response. This field is required if Copy operation is not selected. Otherwise, this field is disabled.

This table shows the fields that are part of the Key Value group:

Fields	Description
Use default key value	Select this option if the key value field for the published event should use a default value.
Fields	Fields to be included in the key value field for the published events. The fields must first be defined in the **Field Metadata** field list in the **Subscribed Event** tab before they can be added to this field list. If Use default key value is selected, this field list is disabled.

Adding a field to the list

1 In the field list, click **Add**.

2 Select the field to add.

All fields that are created in the Field Metadata are displayed, except for fields that are already added.

3 Optionally, you can specify a search string in the **Search** field to filter the list.

4 Click **Add**.

The selected field is added to the list.

Moving a field in the list

1   In the field list, select the field to move.

2   Click **Up** or **Down**.

Removing a field from the list

1   In the field list, select the field to remove, and click **Remove**.

2   In the confirmation message, click **YES**.

The field is removed from the list.

3   Optionally, you can add the field to the list again.

Filtering the list

1   In the field list, specify a search string in the **Search** field.

Only fields with matching names are displayed in the list. The search string only needs to match part of the field names.

2   Clear the **Search** field to show all fields in the list.

Note: Fields in the **Virtual Fields** field list on the **Fields** tab can be added to the key value field, but these are not managed on this tab.

Fields tab

The purpose of this tab is to manage the fields used for the published events. This tab, together with the **Published Event** tab, is used to define published events.

The Field tab consists of these groups:

•   Copied Fields
•   Fields

Copied Fields group consists of a list of fields to be copied from the received events to the published events.

The fields must first be defined in the Field Meta data field list in the **Subscribed Event** tab before they can be added to this field list.

The Virtual Fields group consists of a list of new fields to be included in the published events. These fields use expressions to set values in runtime. Fields in the received events and hard-coded values can be used with the expressions.

Virtual fields are added to the published events after the copied fields. Virtual fields can also be added to the key value field for the published event.

Adding a field to the list

1   In the field list, click **Add**.

2   Select the field to add.

   All fields created in the **Field Metadata** fields are displayed, except for the fields that are already added.

3   Optionally, you can specify a search string in the **Search** field to filter the list.

4   Click **Add**.

   The selected field is added to the list.

Moving a field in the list

1   In the field list, select the field to move.

2   Click **Up** or **Down**.

Removing a field from the list

1   In the field list, select the field to remove.

2   Click **Remove**.

3   Optionally, you can add the field to the list again.

Filtering the list

1   In the field list, specify a search string in the **Search** field.

   Note: Only fields with matching names are displayed in the list. The search string only needs to match part of the field names. Filtering is not case sensitive.

2   Clear the **Search** field to show all fields in the list.

Creating a new field in the list

1   In the field list, click **New**.

2   Specify a name for the field.

   Note: The specified name must be unique within the lists on this tab. This field is required.

3   Specify an expression for the field. The expression must return a string value.

   For more information about the syntax and available functions, see

   Note: Expressions are required.

4   Optionally, specify a description for the field.

5   Optionally, select **Append to key value** to append the field to the key value field.

   This is indicated by yellow in the field list.

6   Click **Save**.

The new field is created.

Editing a field in the list

1   In the field list, select the field to edit.

2   Click **Edit**.

3   Edit the properties for the field.

4   Click **Save**.

Expressions

You can specify an expression in the **Criteria** field of the **Filtering** tab and in the **Expression** field of the New Virtual Field dialog box. This expression works similar to that in Java, but is simplified.

Examples of Boolean expressions used in the **Criteria** field:

```
ADTH = 1
SUTY <> 5
STCO = 'LANC'
BODE = 1 AND BDMT = 4
STCO = 'PYCD' and STKY <> '' and SUBSTRING (PARM, 1, 2)
E065 = 'BOD' AND OLD (STAI) < '60' AND STAI = '60'
INOU == 2 AND PGRS == 90 AND (RORC == 4 OR RORC == 5)
```

Examples of string expressions used in the **Expression** field:

```
'Sync'
'NOLOOKUP'
'60s'
'SupplierPartyMaster_' + CONO + '_' + SUNO
```

Syntax

This table shows the different syntax used in expressions:

Type	Syntax
Reserved words	• true, false • NOT, not • AND, and • OR, or • LT, lt • GT, gt • GE, ge • LE, le • NE, ne • EQ, eq
Calculation operators	• Addition, + • Subtraction, - • Multiplication, * • Division, / **Note:** Mathematician operator precedence is not implemented. Use parenthesis grouping.
Logical operators	• And: AND, and, && • Or: OR, or, • Not: NOT, not, !
Comparison operators	• Equal: EQ, eq, =, == • Not equal: NE, ne, <>, != • Less than: LT, Lt, < • Greater than: GT, gt, > • Less that or equal to: LE, le, <= • Greater than or equal to: GE, ge, >=
Grouping	(...)
Boolean values	• true • false
Fields	Specified in the Field Metadata or Virtual Fields field lists.
Constants	• Numeric: 1234 • Decimal: 1234.56 • String: 'Abcdef' **Note:** Strings cannot contain the characters ' or ".
String concatenation	+

Functions

A function name is given in capital letters and is followed by a parenthesis. Arguments are given within the parenthesis, separated by commas.

For example:

```
OLD (STAI)
ISBLANK (TRIM (CORI))
SUBSTRING (PARM, 1,2)
```

This table shows the available functions and their descriptions:

Functions	Description
ISBLANK(field)	Returns true if the field is empty; otherwise, the value returned is false.
ISCHANGED(field)	Returns true if the field has a different value than its old value; otherwise the value returned is false.
	This function can only be used for received events with the Update operation.
OLD(field)	Returns the old value of the field.
	This function can only be used for received events with the Update operation.
SUBSTRING(string, beginIndex, endIndex)	Returns a substring of the string.
	The attribute `beginIndex` is zero-based and inclusive. The attribute `endIndex` is zero-based and exclusive. For example, if you want to extract 'CD' from the string 'ABCDE', use beginIndex = 2 and endIndex = 4.
TOUPPER(string)	Converts all characters in the string to uppercase.
TOLOWER(string)	Converts all characters in the string to lowercase.
TRIM(string)	Removes leading and trailing white space from the string.

Adding a session

1   In the Session tree panel, click **Add Session**. The session name must be unique for it to be accepted.

2   In the Session editor, specify the name of the new session.

By default, the name of the new added session is `New_Session_<number>`, where `<number>` is the lowest available number.

3 Optionally, add a description for the session.

4 Click **Save**. The new session is a custom session.

The added session is appended as the last of the session nodes. The session is displayed in its alphabetical order when **Refresh** is clicked.

Duplicating a session

1 In the **Session tree** panel, select the session to be duplicated.

2 In the Session editor, click **Duplicate**.

3 Specify the name of the new session.

By default the new session follows the name of the duplicated session and is appended with _Copy_<*number*>, where *number* is the lowest available number.

4 Optionally, add a description for the new session, or edit the existing description.

5 Click **Save**.

The duplicate session is displayed in the **Session tree** panel. By default, the new session is appended as the last of the session nodes. Then session is displayed in its alphabetical order when **Refresh** is clicked.

Note: When you duplicate a standard or a custom session, the new session is always a custom session. By default, the duplicate session's status is Stopped.

Editing a session

1 In the Session tree panel, select the session to edit.

2 In the Session editor, edit the description for the session.

3 Click **Save**. If you leave the session editor without saving your changes, a confirmation message is displayed.

Note: You can only edit custom sessions.

Exporting a session

1 In the Session tree panel, select the session to export.

2 In the Session editor, click **Export**. A file with the same name as the session and with the file extension rulespack is downloaded to your browser.

3 Move the file to your desired file location.

Note: To export multiple sessions, click the **Export** icon in the Session tree panel. You can only export custom sessions.

Exporting multiple sessions

1   In the Session tree panel, click **Export**. The list of custom sessions is displayed.

If you applied a filter on the session tree, this filter will be applied on the list of custom sessions to export.

2   Select the sessions to export.

3   Optionally, click the More icon.

- Click **Select all** to select all the custom sessions in the list.
- Click **Deselect all** to clear the check boxes.

4   Click **OK**. A file with the name myrules.rulespack is downloaded to your browser.

5   Move the file to your desired location.

Importing sessions

1   In the Session tree panel, click **Import**.

2   Browse to the rulespack file that you want to import and click **OK**. All sessions in the rulespack file will be imported.

If a session name in the rulespack file matches the name of an already existing session, the existing session is replaced. The start/stop state of the replaced session is maintained. The active/inactive state of the rules under the replaced session are maintained. A confirmation message with all the added and replaced sessions is displayed.

3   Take note of the information and click **OK**.

Note: Imported session that did not exist are stopped and all rules under these imported sessions are active. You can only import custom sessions.

Starting a session

1   In the **Session tree** panel, select the session to start.

2   In the Session editor, click **Start**.

3   In the confirmation message, click **YES**.
The status indicator shows the session is Started and the message prompt closes.

Restarting a session

If you have edited rules in a started session, but you have not reactivated the rules, you can restart the session to reactivate all edited rules simultaneously.

1 In the **Session tree** panel, select the session to restart.

2 In the Session editor, click **Restart**.

3 In the confirmation message, click **YES**.

Stopping a session

1 In the **Session tree** panel, select the session to stop.

2 In the Session editor, click **Stop**.

3 In the confirmation message, click **YES**.
The status of the session is changed to Stopped.

Deleting a session

1 In the **Session tree** panel, select the session to delete.

2 In the Session editor, click **Delete**.
Note: The delete icon is only enabled for a session that is not started.

3 In the confirmation message, click **YES**.
Note: You can only delete custom sessions.

Adding a rule

1 In the **Session tree** panel, expand the session.

2 Click **Add Rule** in the **Session editor** panel.
Session editor is changed to **Rule editor**.

3 On the **General Information** tab, specify the name of the new rule.

By default, the name of the added rule is New_Rule_<*number*>, where *number* is the lowest available number. The rule name must be unique.

4 Specify the fields of the Rule editor for the new rule.

See Rule editor panel on page 122.

5 Click **Save**. All mandatory fields must be specified to save the rule.
The added rule is appended as the last of the rule nodes in its session. The rule is displayed in its alphabetical order when **Refresh** is clicked.
Note: You can only add rules in custom sessions.

Duplicating a rule

1   In the **Session tree** panel, expand the session node of the rule.

2   Click the rule to be duplicated.

3   Click **Duplicate**.

4   Specify the name of the new rule.

By default, the new rule follows the name of the duplicated rule and is appended with `_Copy_<number>`, where `<number>` is equal to the lowest available number.

5   Optionally, specify the fields in the Rule editor.

See [Rule editor panel](#) on page 122.

6   Click **Save**. All mandatory fields must be specified to save the rule.
The rule is displayed in the session tree. By default, the duplicate rule's status is Stopped. The new rule is appended as the last of the rule nodes in its session. The rule is displayed in its alphabetical order when the **Refresh** is clicked.
Note: You can only duplicate custom rules under the same custom session.

Editing a rule

1   In the Session tree panel, expand the session node of the rule.

2   Click the rule to edit.

3   Edit the desired fields in the Rule editor.

See [Rule editor panel](#) on page 122.

4   Click **Save**.

If you leave the rule editor without saving your changes, a confirmation message is displayed. Click **YES** to save changes.

Note: You can only edit custom rules.

Validating a rule

1   In the **Session tree** panel, expand the session node of the rule.

2   Click the rule to be validated.

3   Click **Validate**. A notification message is displayed. The edited rule displayed in the rule editor is validated, not the saved rule.

Note: You do not need to save the rule to validate it. You can only validate custom rules.

Activating a rule

1 In the **Session tree** panel, expand the session of the rule.

2 Select the rule to be activated.

3 Click **Activate/Reactivate**.

Note: A validation is automatically done when activating a rule. An invalid rule cannot be activated.

4 In the confirmation message, click **YES**.
The status is changed to Active and a notification that the rule is activated successfully is displayed.
Note: You can only activate custom rules.

Reactivating a rule

After you have edited an active rule in a started session, reactivate the rule so that it becomes available.

1 In the **Session tree** panel, expand the session node of the rule.

2 Select the rule to be reactivated.
The Session editor is changed to Rule editor.

3 Click **Activate/Re-activate**.

Note: A validation is automatically done when reactivating a rule. An invalid rule cannot be reactivated.

4 In the confirmation message, click **YES**.
A notification is displayed stating that the rule is activated successfully.
Note: You can only reactivate custom rules.

Deactivating a rule

1 In the **Session tree** panel, expand the session of the rule.

2 Select the rule to be deactivated.
The Session editor is changed to Rule editor.

3 Click **Deactivate**.

Note: You can only deactivate custom rules.

4 In the confirmation message, click **YES**.
The status of the rule is changed to Inactive.

Deleting a rule

1 In the Session tree panel, expand the session node of the rule.

2   Click the rule to be deleted.

3   In the Rule editor, click **Delete**.

Note: You can only see the delete icon on a rule that is inactive.

4   In the confirmation message, click **YES**.

Note: You can only delete custom rules.

Chapter 12: Event Hub administration tasks

Event Hub 4 and Event Analytics is an M3 system that monitors events for related M3 applications. This version has been improved for better performance, stability and security.

The administrator UI for Event Hub has been updated, with a minor functional change for time correlation in Event Analytics. Event Hub and Event Analytics are managed by Tenant Administrators and are not used by regular M3 CE users. Event Hub is used for outbound M3 CE BODs including EDI BODs, outbound M3 CE FBMs, Data Lake Publisher, Functional Search, etc. Event Analytics is used for filtering of events for outbound M3 CE BODs and FBMs. This guide explains the new technology and latest features.

Event Hub version 4 uses the Apache Kafka® engine instead of JBoss HornetQ which Event Hub 3 used. Event Analytics version 4 uses Apache Kafka® Streams engine instead of JBoss Drools which Event Analytics 3 used. Apache Kafka® persists events on disk in directories called partitions. Partitions are grouped by a name which is called a topic. For Event Hub, the tenant ID plus one or more subscriptions (without the operations) are mapped to topics.

You can specify which partition within a topic stores the event using the "keyValue" element. The element keyValue contains the primary key for the event. The element keyValue is set for all database events published by M3. You can also set it using rules in Event Analytics.

Events are only ordered within their partitions. If you create item A and then update item A two times, these three events will always be delivered to subscribers in the correct order. This is because they all have the same keyValue. If you update item B before item A is updated, the event for the item B update may be delivered in a different order because it has another keyValue and can be stored in another partition.

Each event is identified by an offset. The offset is an index of the event within its partition. Offsets for the latest consumed event within each partition are stored per subscriber. Several subscribers can consume events from the same partition.

All events persist during the retention period. You can browse events within the retention period, by selecting one or more subscriptions and specifying a start time. You cannot purge events in Kafka® but you can skip unconsumed events by setting a new start time for events to consume.

To access **Event Hub**, select **Administration Tools > Event Hub** from the Infor M3 Menu in Infor Ming.le. The *EVENTHUB-Administrator* IFS Security role is required to access the **Event Hub** page.

Event Hub tabs

The Event Hub contains two primary tabs:

Page	Description
Browse Events	This tab shows events based on subscriptions and time.
Administration	This tab shows the publishers, subscribers and subscriptions.

Browse Events tab

The Browse Events tab has two sections:

- Subscriptions
- Events

Each section can be expanded by clicking the v button to the right, or collapsed by clicking the ^ button to the right. By default, the Subscription section is expanded and the Events section is collapsed.

Note: This functionality replaces event recording using the Visualization and Administration UIs for Event Hub version 3.

You can use these buttons to view and filter publishers, subscriptions and events:

Button	Definition
Refresh Subscriptions	Click the refresh button to refresh the list of subscriptions.
Search subscriptions	Specify your search query in this field.
Move buttons	Click a move button to move the selected subscriptions to the other side.
Get events	Specify the start time and click Get Events to show the list of events that meet the specified criteria.
Refresh events	Click the refresh button to refresh the list of events.
Export events	Click to export events in a downloadable file of the specified format.

Getting the events list

You can view events stored in Event Hub by following these steps:

1   Select **Administration Tools > Event Hub**.

2   Select the tab **Browse Events**.

3   If the Subscriptions section is collapsed, expand it.

4   From the available subscriptions list view, select one or more subscriptions.

5   Optionally, use the search function to find and select the subscription by specifying the whole or part of the subscription in the Search field. The search value is case insensitive.

6   Click the **Move right arrow**.

7   Optionally, if you want to remove one or more subscriptions from the Selected subscriptions list view, select one or more subscriptions that you want to remove.

8   Optionally, use the search function to find and select the subscription you want to remove by specifying the whole or part of the subscription in the Search field. The search value is case insensitive.

9   Click the **Move left arrow**.

10 Optionally, you can refresh the list of available subscriptions by clicking the **Refresh** button. Refreshing clears selected subscriptions, start time, and browsed events.

11 Enter a date and a time in the field Start Time. Click the **Calendar icon** to select date and time from a calendar. You can also edit the date and time directly in the field. The format "yyyy/MM/dd HH:mm" must be used.

12 Click the **Get Events** button to get the events.

13 The Subscriptions section is automatically collapsed, and the Events section is expanded. This is to create more space to show the events. You can expand the Subscriptions section to select other subscriptions and/or another start time.

14 Optionally, you can refresh the event list view by clicking the **Refresh** button.

15 Optionally, you can export the events to a CSV or EXCEL file by clicking the **Export** button. Only selected events will be exported.

Converting the downloaded excel file format

You can download events list in .xls file format. When opening the file using Microsoft Excel, you will encounter the warning:

The file format and extension of *filename* don't match. The file could be corrupted or unsafe. Unless you trust its source, don't open it. Do you want to open it anyway?

Click **Yes** and open the file. To save the file in a format that will not show this warning when you open the file, you must convert the file into .xlsx format by following these steps:

1   Open the file.

2   Click the **File** tab.

3   Click **Convert**.

4   Click **OK**.

Events table

This table shows the columns for the Events list view and their descriptions.

Note: You must get the events list first, before you can view the events table contents. The events are shown in an arbitrary order, decided by Kafka®. The events are not shown in chronological order.

Column	Description
Partition	The partition the event belongs to.
Timestamp	The time when the event was received by the Event Hub.
Publisher	The application that published the event.
Document	The name of the data entity.
Operation	The action on the document that triggered the event.
Elements	The name-value pairs for the document data. Old values are shown within parenthesis.
Client Time	The time when the event was published.
Tracking ID	The ID to track a series of related events.
Offset	The position of the event within its assigned partition.

Note: The Events list view uses a so called "infinite scroll pattern". This means that all events might not be shown. When you scroll to the end of the list, more events are automatically fetched and appended to the list.

Administration tab

The Administration tab has two sections:

• Publishers and Subscribers
• Subscriptions

Each section can be expanded by clicking the v button to the right, or collapsed by clicking the ^ button to the right. As default the Publishers and Subscribers section is expanded and the Subscriptions section is collapsed.

Note: The Administration tab replaces the Event Hub Administration UI for Event Hub version 3.

Publishers and Subscribers section

The Publishers list view shows a list of names of all registered publishers for the Event Hub. This section contains two list views:

- Publishers
- Subscribers

The Publishers list view shows a list of names of all registered publishers for the Event Hub. The Publishers and Subscribers list views have several buttons:

Button	Definition
Refresh Publishers	Click the refresh button to refresh the list of publishers.
Filter publishers on selected subscribers	Click to only view the selected subscribers' publishers. **Note:** You can select several subscribers.
Refresh Subscribers	Click the refresh button to refresh the list of subscribers.
Filter subscribers on selected publishers	Click to only view the selected publishers' subscribers. **Note:** You can select several publishers.

To only view publishers publishing events to one or more subscribers, select one or more subscribers in the Subscribers list view and click the Filter button for the Publishers list view. To show all publishers again, click the No Filter button for the Publishers list view.

Note: There is no status available for publishers (producers) in Kafka®.

This table shows the columns for the Subscribers list view and their descriptions.

Column	Description
Subscriber	A list of names of all registered subscribers for the Event Hub.
Status	Number of connected subscriber instances. A green color indicates that the subscriber is OK. A red color indicates that no subscriber instances are connected.
Time lag	Time between the latest published event and the latest consumed event.

Column	Description
Offset lag	Max number of events to be consumed. **Note:** There might be events with non-matching operations persisted in the partitions. There might also be internal administration records. As these events also counts, it is impossible to give an exact figure for number of events to be consumed.

To only view subscribers subscribing to events published by one or more publishers, select one or more publishers in the Publishers list view and click the Filter button for the Subscribers list view. To show all subscribers again, click the No Filter button for the Subscribers list view.

Note: Sessions in Event Analytics are grouped and shown as separate subscribers. Their subscriber names start with "EventAnalytics."

The columns Time lag and Offset lag can be used to determine a possible buildup of non-consumed events for subscribers.

Subscriptions section

This section contains the Subscriptions list view.

Column	Description
Publisher	The application that publishes events.
Document	The name of the data entity.
Operations	The actions on the document that trigger events. Several operations separated by commas can be given for one subscription.
Refresh subscriptions	Click this button to refresh the subscriptions list.

Chapter 13: Data Lake Publisher administration tasks

Data Lake Publisher enables M3 Business Engine to publish data to Infor Data Lake. By selecting documents (equivalent to M3 tables), Event Hub subscriptions are created and events from the selected documents are published to Infor Data Lake through ION Messaging Service.

To access **Data Lake Publisher**, select **Administration Tools > Data Lake Publisher** from the Infor M3 Menu in Infor Ming.le. The IFS Security role *M3-DLSubscriptionAdmin* is required to access Data Lake Publisher.

Configure documents for publishing to Data Lake

1   Select **Administration Tools > Data Lake Publisher**.

2   From the Available list view, select one or more documents to publish in Data Lake.

3   Optionally, use the search function to find and select the document that you want to enable by specifying its document name or document description in the **Search** field.

4   Click the move right arrow.

5   From the Selected list view, click the Save icon.

6   In the confirmation message displayed, click **Close**.

Selecting the move right arrow does not automatically enable specified documents in Event Hub. Ensure to click the Save icon.

7   Refresh the connection point and data flow.

See

Disable publishing of documents

1   Select **Administration Tools > Data Lake Publisher**.

2   From the Selected list view, select one or more documents to disable.

3   Optionally, use the search function to find and select the document that you want to disable by specifying its document name or document description in the **Search** field.

4   Click the move left arrow and click the Save icon.

5   In the confirmation message displayed, click **Close**.

Selecting the move left arrow does not automatically disable specified documents in Event Hub. Ensure to click **Save**.

6   Refresh the connection point and data flow.

See .

Exporting documents

1   Select **Administration Tools > Data Lake Publisher**.

2   From the Selected list view, click the Export documents icon.

This action downloads all enabled documents under Selected list in a .zip file, to your machine. The .zip file contains the .json file that is passed to the server for processing.

Importing documents

1   Select **Administration Tools > Data Lake Publisher**.

2   From the Selected list view, click the Import documents icon.

This action enables you to import documents to the Data Lake Publisher. It overwrites documents in the Selected list based on document name. If the document name is not currently enabled, the document is imported in the Available list.

3   Refresh the connection point and data flow.

See .

Refreshing the connection point and data flow

1   Select **ION Desk > Connect > Connection Points > ehub6_eventhub**.

Note: Connection point may be named differently depending on the environment.

2   Click **Refresh**. Once clicked, ION calls the discovery REST endpoint in EH/DLP.

The selected documents are displayed in the list.

3   Click **Save**.

4   Select **ION Desk > Connect > Data flows > M3_EventHub-Data_Lake M3-Integration**.

Note: The data flow may be named differently depending on the environment.

5   Click the document icon next to the M3/Event Hub connection point.

6   Click the Add icon.

7   Select the new documents from the list and click **OK**.

8   Click **Save**.

9   Click **Reactivate**.

Chapter 14: Infor Enterprise Collaborator administration tasks

Infor Enterprise Collaborator (IEC) is an M3 software component that allows you to create, send and receive electronic documents in XML and flat file formats. In M3, there are a lot of documents that can be sent or received, such as invoices, orders, picking lists, bank payments and others. IEC is the M3 component that builds, sends, and receives these documents in electronic format.

Infor Enterprise Collaborator (IEC) Server is the engine used at runtime when processing messages. If a message fails, access the **Enterprise Collaborator Administration** page to investigate the problem, review received files, and examine logs and routing configurations.

As an administrator, you must monitor the IEC Server and perform administration tasks on Enterprise Collaborator Administration pages to address any business challenge. You are primarily responsible to maintain server system stability, perform system troubleshooting, and provide user access to the server.

Infor Enterprise Collaborator provides message-based collaboration between functions in the Infor M3 enterprise management system and other external applications, in order to enable business process collaboration within and across enterprises.

To access **Enterprise Collaborator Administration**, select **Administration Tools > Enterprise Collaborator Administration** from the Infor M3 Menu in Infor Ming.le.

This table shows the IFS Security roles required for accessing Infor Enterprise Collaborator:

IFS role	Definition
M3EC-Administrator	The M3EC-Administrator IFS role grants administrator privileges in Infor Enterprise Collaborator.
M3EC-User	The M3EC-User role is limited and works with M3 data translations from an integration perspective.

Enterprise Collaborator Administration page

The administration page has several available tabs or pages. These pages allow you to perform IEC monitoring and managing functions:

Page	Description
Messages on page 149	The Messages page allows you to search and view messages. This is your primary monitoring tool.
Server on page 160	The Server page allows you to monitor server activity.
Communication on page 161	The Communication page allows you to monitor incoming and outgoing communication channels.
Events on page 162	The Events page allows you to search and view event logs.
Utilities on page 164	The Utilities page allows you to manage utility tasks and download Business Document Mapper.
Settings on page 168	The Settings page allows you to manage the application settings.
About on page 170	The About page allows you to view version information about Infor Enterprise Collaborator.

Messages

The Messages page provides features for searching messages based on status and retrieving message-specific details. You can view message related activities, filter message views, and messages.

Message states

This diagram shows the message runtime states and the available actions. Standard BOD agreements are delivered automatically.

Message lifecycle states
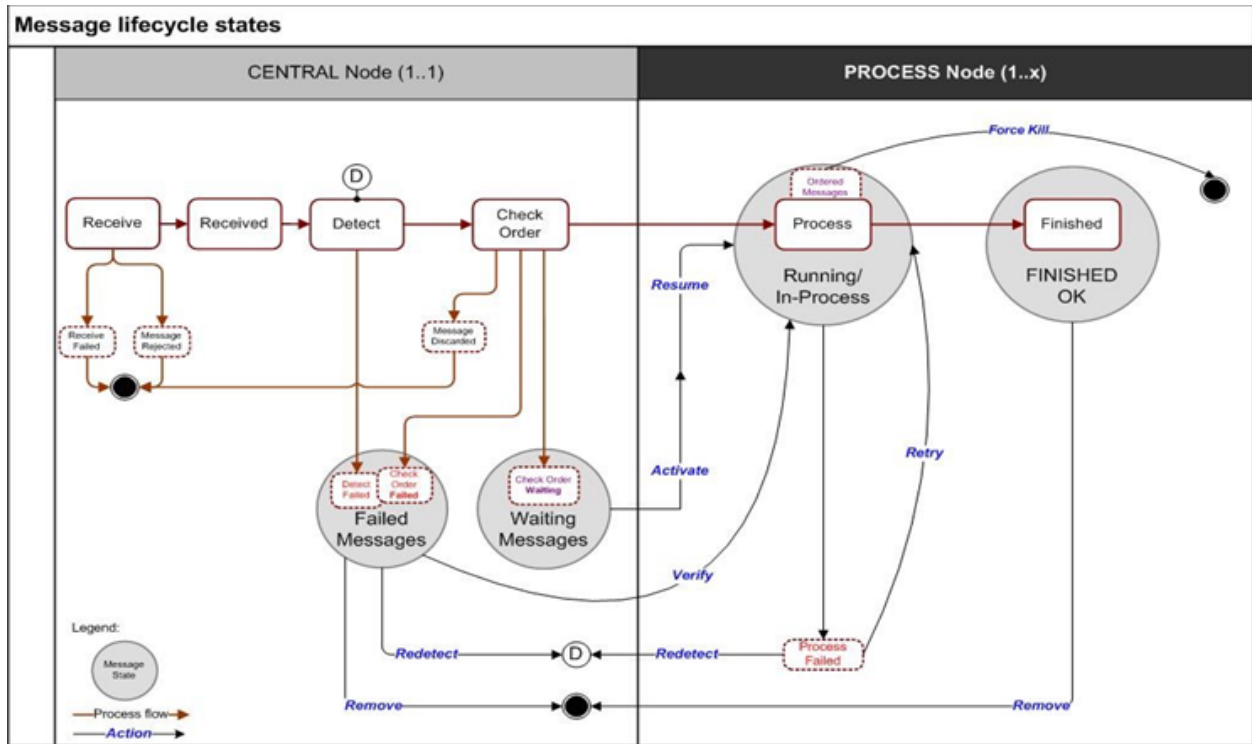
This table shows the description for each message state:

Runtime Message states	Description
Receive	When a communication channel receives a message, the message is put in Receive state before any action is done to the message. A message stays in Receive state until it is fully received or downloaded by IEC.
Received	Message payload and manifests are persisted in the file system. When a message has been fully received into IEC, the message is put in Received state. Messages in Received state are ready for detection.
Receive Failed	If an error occurs while a communication channel receives a message, the message is put in Received Failed state. Most errors in this state are caused in the communication between the message sender and communication channel receiver.
Message Rejected	If the DocIDController is enabled, duplicated messages received by IEC will be rejected and the message is put in Message Rejected state.
Detect	Received messages are identified by IEC based on the message's partner agreements.

Runtime Message states	Description
Detect Failed	If an error occurs in message detection process, for example, the message did not match any configured partner agreements, IEC does not know how to proceed and the message is put in Detect Failed state.
Check Order	IEC checks the message order based on its primary key and variation ID. Check Order is part of the partner agreement process list. If used, a Check Order is recommended to be added only once for ordered messages and it must be the first process in the process list.
Check Order Failed	If a failure occurs in retrieving the message primary keys or variation ID, a message is set in Check Order Failed state.
Check Order Waiting	The message is set in waiting state based on the ordered primary key value.
Message Disregarded	If the variation ID value of a message is lower than the current variation ID value, the message is disregarded.

Note: You can modify the date format and time zone settings in **Settings > Date/Format Settings** page.

The message is transferred from central node to process node:

- Delegate to process

  From the central node, messages are transferred to the process node. When the message is being transferred, it goes through delegate to process state.

- Delegate received

  After the message is received in the process node, the message state becomes delegate received state.

Runtime Message states	Description
Process	In Process state, the message is iterating through the agreement process list. This state is a composite of different steps that are dependent on the partner agreement.
	In a partner agreement, there must be at least one process step. Then, a process counter is used to determine at which step in the process list a message is placed. The process counter is stored in the manifest file.
Process Failed	If an error occurs in a specific process step, the message is put in Process Failed state.

Runtime Message states	Description
Finished	In Finished state, the message is finished iterating through its partner agreement process list.
Retry	Retry means repetition of the last failed process step.
	If a failure occurs during Detect or Check Order, a Retry action restarts processing a message from Detection point.
Redetect	Redetect resets the manifest, sets the state to Received, and starts from Detection point. If Redetect process is used in Partner Admin tool process steps, it is equivalent to sending a message from IEC back into IEC. In Redetect, the current message is redetected and no further processing can be done.

Message lifecycle states

These are the message lifecycle states. You can view the log of message states from **Enterprise Collaborator Administration > Events > Log** page.

Message lifecycles states	Description
Running or In-Process	This state indicates that you cannot take any action on a message, except if the message is also an ordered message.
	For in-process and ordered messages, you can invoke **Force Kill** from the Ordered View. The Force Kill action will attempt to terminate the running process. Use Force Kill on a message that has been running unexpectedly longer than it should and is causing a build up of message into IEC.
	You can access Ordered View from **Enterprise Collaborator Administration > Messages > Ordered** page.
Finished OK	This state indicates messages that have completed its process. You can remove finished messages.
Failed messages	This state indicates a failed message. You can invoke Retry, Redetect, Remove, and Verify actions.

Message lifecycles states	Description
Waiting messages	This state shows after the dependencies are completed. Wait for these messages to be activated before you manually resume. The Resume action puts the waiting message back into process state.

Message lifecycles states	Description
Running or In-Process	This state indicates that you cannot take any action on a message, except if the message is also an ordered message. For in-process and ordered messages, you can invoke Force Kill from the Ordered View. The Force Kill action will attempt to terminate the running process. Use Force Kill on a message that has been running unexpectedly longer than it should and is causing a build up of message into IEC. You can access Ordered View from **IEC Management page > Messages > Ordered**.
Finished OK	This state indicates messages that have completed its process. You can remove finished messages.
Failed messages	This state indicates a failed message. You can Retry, Redetect, Remove, and Verify actions. **Note:** Failed messages are deleted after 60 days and everything else related to these failed messages are deleted after 30 days.
Waiting messages	This state shows after the dependencies are completed. Wait for these messages to be activated before you manually resume. The Resume action puts the waiting message back into process state.

IEC Management Message tasks

These messaging tasks are available in **Enterprise Collaborator Administration > Messages** page. Depending on the state at which a message is currently at, click on a link below to view the tasks.

Status

The Status link shows the latest messages and available options.

On this page, you can filter to view the messages by status such as All, Finished, Failed, and Disregarded states.

You can terminate message processing of any message by clicking the **Terminate** button under the maintenance column. This button will only show for processes involving XML transform. Error processing messages will not have the **Terminate** button. You can remove messages from the system by clicking the **Remove** button.

Note: Terminating message processes that have hanged will sometimes take several minutes.

Autotermination

IEC automatically terminates problematic messages that meet certain criteria involving CPU, Heap and Duration monitors. For example, you can set IEC to automatically terminate messages that consume 10% of a CPU's single-core bandwidth. This feature is used to prevent erring messages from consuming resources and causing issues for other messages and systems.

See [Searching by status filter](#) on page 157 for using status filters and viewing messages that have issues.

Advanced Search

The Advanced Search link allows users to run full-text and wild card searches using UUID, state, and other search details. Use full-text search for better precision results, as only the exact specified text will be searched. Use wild card search to show all results that contain the string within the wild card.

Advanced Search has several search criteria fields you must accomplish to initiate an advanced search:

Field	Definition
Identifier	Identifier is the message UUID or unique identifier. Each message has one.
Status	Status is a filter to limit what is displayed in the search.
Received after	Received after limits the search results based on the time after the messages are received.
Received before	Received before limits the search results based on the time before the messages are received.
Partner Agreement	Partner Agreement features a drop-down menu that contains related search options:
Partner	Partner is the parent folder of an agreement.
Agreement	Agreement is the agreement name.
Received through protocol	Received through protocol is the channel protocol used to receive the message.
Received through channel	Received through channel is the name of the channel.

Field	Definition
Sent through protocol	Sent through protocol is the channel protocol used to send this message out of IEC.
Sent through channel	Sent through channel is the name of the send channel used to send this message out of IEC.

Advanced Search Wild Card

You can do a more specific advanced search using wild cards. Wild cards are search criteria that can contain any text related to the target agreement. These are specified by adding a "%" character before and after the target search string. A wild card search for agreements is shown below.

Note: Searching using wild cards depend on the search criteria specified. Apart from the example shown below, you can also specify a **test** and **ident** wild card and this gives you the option to input a wild card filter for the **ident** field. A wild card can be used in conjunction with a UUID.

1   Go to **Message > Advanced Search**.

2   Specify any available advanced search criteria fields.

3   Specify the wild card search criteria in the **Agreement** field. Your input text must be preceded and followed by the "%" character.

4   Click **Submit**.
IEC will display all the agreements with the specified wild card word in any part of the agreement name.

CPU Usage

You can monitor the CPU resource consumption of IEC messages using the CPU Usage tab. Use this function to visually monitor mappings and determine if there are any problems with CPU consumption. If a mapping shows excessive CPU bandwidth use, you can manually terminate the IEC messaging process. If IEC is processing a lot of messages, the CPU Usage tab will take longer to load.

Note: When "N/A" is displayed instead of a valid CPU percentage (0% - 100%), this means that the message has already ended, terminated or its CPU consumption cannot be determined. You must expect changes to the displayed messages list on the current page, as the messages list refreshes every time you click on or jump to a different page.

You can terminate a message in CPU Usage tab by following these steps:

1   Click the **Show** link beside the message.

2   Click **Terminate** button under the maintenance column.

Retry/Redetect

You can view and manage failed messages using the Retry/Redetect link. All messages are sorted by Receive Time. To filter messages by Partner/Agreement in Retry/Redetect, go to the **Agreement Retry/Redetect Messages > Partner and Agreement > View > Show**.

In Redetect, the current message is redetected and no further processing can be done. Redetect process step is equivalent to sending a message from IEC, back into IEC.

Use these guidelines:

- Messages such as Detect, Send, Validate, XML transform in failed state can be filtered, except for Receive Failed and Message Rejected states.
- Redetect All helps correct message errors that might occur if messages are incorrectly configured in any of these areas:
  - Routing
  - Processing
  - Classpaths for generated transformation

  In message reprocess, incoming messages are stored persistently. In some cases, you do not need to reprocess a message. For example, when a partially processed message caused an external system, such as M3, to update and commit data that has not been rolled back, then you must manually verify the message.
- The Redetect All, Retry All, and Verify All actions apply only to messages in current view.

Agreement Retry/Redetect Messages

The Agreement Retry/Redetect Messages link allows you to view and manage failed messages grouped by Partner and Business or by technical messages. Business messages are suppressed by the partner agreement to summarize error messages. Technical messages are not defined in the partner agreement, but it shows more details about the error.

Ordered

The Ordered link shows any active and waiting ordered messages. There are filters to let you identify what is the current active message in a particular order ID and also what messages are next for processing. Ordered messages result from ordered channels, check order process, and split redetect process.

Variation ID

The Variation ID link (VID) uses a numeric value as basis of process order. These fields let you define your search criteria and results will be displayed in the bottom three fields. You can then update or remove these:

- Channel displays a list of active channels. This is used for channel-based ordered messages and check order process.
- Primary Key searches ordered channel messages using the primary key value "variation-ids". It can also be used to search for the primary key retrieved from the check order process.
  Note: The primary key is dynamic. See Configuring check order on page 207 for more information.

Suppressed Error Reports

The **Suppressed Error Reports** link displays the Suppressed Error document or the summarized Business messages. When an error occurs, run the scheduled Error Aggregation task to show the number of errors according to Error type and Agreement. Messages with errors are sent back to the server for reprocessing. Reprocessed messages keep unique identifiers and no tracking information is lost.

Using message search and view filter

On the **Enterprise Collaborator Administration** page, click **Messages > Status**.

Searching by UUID

1   In the **Identifier** field, specify the message UUID.

2   Click **Show**. The view shows a list of messages matching your specified UUID.

Searching by message

1   From the table list, select a message to view.

2   Click the <u>Show</u> link.

The view shows the manifest details, host,central file system, and comment for your selected message.

You can perform these tasks on this page:

- Click the **Log** icon to view the message content such as log, content file, error report, and manifest information.
- Click the **Save** icon to download the information in the message details window.

Searching by status filter

Different message states can be viewed using status filters. Use this feature to efficiently search for messages that are active or have failed. You can also use the **For Review** filter to view messages and mark them as reviewed, so these messages can be filtered off and you can proceed to review other messages.

Note: IEC will display *There are messages that need your review. Select the "For Review" filter to see the messages* if there are messages that require attention.

1   Click **Messages > Status** in IEC.

2   Select one of these status options:

- **All states** displays all messages.
- **All finished states** displays only the finished and verified messages.
- **All failed states** displays all messages in failed status.

- **All disregarded states** displays all messages in disregarded status.
- **All in-process states** displays all messages currently being processed.
- **All Reviewed** displays all messages that have been marked as reviewed.
- **For Review** displays all IEC highlighted messages that have not yet been marked as reviewed.

3   Click **Filter**.

The view shows an updated table list of messages based on your filter view criteria.

Searching by advanced criterion

1   Click **Advanced Search** button.

2   Select a message status:

All states, All finished states, or All failed states.

3   Specify the advanced search parameters.

4   Click **Submit**. The view shows an updated table list of messages based on your advanced search criteria and parameter specifications.

5   On this page you can also select these actions for the message:

Redetect All, Retry All or Verify All.

6   Click **OK**.

Viewing by ordered messages

On the **Enterprise Collaborator Administration** page, click **Messages > Ordered**. Perform any of these tasks:

- Select Active (non-waiting), then click **Show** to display all active ordered messages.
- Select All, then click **Show** to display all ordered messages.

Processing failed messages

You can manually take action on failed messages to review or resend them.

Note: Messages that have been successfully sent and verified by IEC are kept for 30 days. Messages that have failed are kept by IEC for 60 days, before the messages are discarded. Use this time frame to review the messages.

1   Access the **Enterprise Collaborator Administration** page.

2   Click the Messages link.

3   Click the Retry/Redetect link. The view shows a table list of all failed messages except for the messages in `Received` failed and `Messages Rejected` states.

4   In the **State** field, select a filter option from the drop-down list.

5   Click **Filter**.

6   Perform these messaging tasks from this page:

- Process all messages in the current view:
  - Click **Redetect All** to reprocess the failed messages from the start.
  - Click **Retry All** to reprocess failed messages from when it stopped.
  - Click **Verify All** to reprocess and tag the messages as finished. No further action is performed.
- Process selected message in the table list:
  - Click **Redetect** to reprocess the failed messages from the start.
  - Click **Retry** to reprocess the failed messages.

Note: Retried and redetected messages are not displayed in the table list. The table list is updated to include messages with new status.

7 Click **OK**.

Processing failed messages by agreement

You can manually reprocess failed messages to review or resend them.

Note: Messages that have been successfully sent and verified by IEC are kept for 30 days. Messages that have failed are kept by IEC for 60 days, before the messages are discarded. Use this time frame to review the messages.

1 Access the **Enterprise Collaborator Administration > Messages > Agreement Retry/Redetect Messages** page.

2 In the **Partner Drill** down tree view, select the Partner for the agreement and then select the Business or Technical failed message. The view shows the total of number of failed messages.

3 Select a process action for all the messages in the selected Partner and in the Business or Technical messages:

- Click **Redetect** to reprocess failed messages from the start.
- Click **Retry** to reprocess failed messages.

4 Click **OK**.

Viewing message details

Use **Messages** to show information about transaction from logs, archived files, and error reports.

1 Access the **Enterprise Collaborator Administration > Messages** page.

2 Perform a search by message.
See Using message search and view filter on page 157.

3 View the message details. Select a message to view and click the Show link.

4 View the message logs and select an action to perform:

- Click the **Log** icon to view message content such as log, content file, error report, and manifest information.
- Click the **Save** icon to download the information on the message detail window.

Server

This page monitors the server activities of IEC.

Server allows you to view different properties and reports. Use these links to view this information:

- The **Error Report** link extracts server information and generates a report that will help the support team identify issues that are being encountered by the system. It is recommended when reporting any issue with regards to Enterprise Collaborator, to include an error report.

  An error report requires several fields:

Field	Definition
File name	File name is a search field that specifies the name of the file in question.
Log text	Log text is a search field that will filter the results.
UUID	UUID is optional if you want to include only a specific message, process, or activity.
From and To	From and To time specifies the time frame the error was encountered. When reporting an error, you must add an ample amount of time before and after the issue was encountered.

- The **Group Control Properties** allows you to search and update the values of agreement control properties on group level. Control properties are basically an extra level of configuration available when constructing a partner agreement. This is usually used internally to control behavior or pass values to mappings, channels, and IEC processes.

  Group control properties contain several functions:

Field	Definition
Searching Containing Characters	Searching Containing Characters allows you to search for group names by using the specified criteria.
Saving with Optional Server Reload	Saving with Optional Server Reload allows you to save control properties and reload so that the change will be applied to the system without restarting. Saving without reloading will save the changes in the database, but the changes will not be reflected until the system is restarted.
Optional Server Reload	Tick this box to save you the action of going to the Overview page. **Note:** This function is disabled for tenant administrators.

- **Agreement Control Properties** allows you to provide properties for customer-dependent settings. Group Control Properties are inherited by all agreements and sub-groups within the group.
- **Custom Detection** allows the administrator to search and edit custom detection values. Custom Detection will only show standard agreements that have been configured to use custom detection.
- **Agreement Status** allows the administrator to enable, disable and delete custom agreements. Agreements are active and can be used by default. Inactive agreements will not be usable in the server runtime.
- **Custom Mappings** allows the administrator to search and reload custom agreements. You can search using the group name and agreement name. This tab also shows lists of partner agreements.

Communication

The **Communication** page contains several functions for managing communication channels:

- You can change the status of a channel through the <u>Channel Control</u> link. An icon is displayed on each channel indicating the channel state.

  For specific communication channels, you can view or save a copy of the logs, but log details are limited only from the channel's last startup time. To view the logs per channel from the last restart, access the server's channel control page. There is no role that will allow you to do anything but review logs in the Communication page.

  You can perform these tasks from the Channel control link:

  - Restart newly enabled channels through the <u>Load New Channels</u> link, without changing the existing channels.
  - Restart/start all enabled channels using latest data from the IEC database using the <u>Reload All Channels</u> link.
  - Stop a specific channel from receiving incoming messages using the <u>Pause</u> link. <u>Pause</u>may take several seconds to take effect.
  - Restart a specific channel with the latest available data in the IEC database using the <u>Reload</u> link.
  - Start a specific channel from receiving incoming messages using the <u>Resume</u> link.
  - View the present state of each communication channel:

State	Definition
RUNNING	Shows an active channel that is communicating. Event Hub 4 channels use this term.
ACTIVE / REGISTERED / CONNECTED	Shows an active channel that is communicating. This is specific for Event Hub 3 channels.
ACTIVE / REGISTERED / DISCONNECTED	Shows a channel that has stopped communication. This is specific for Event Hub 3 channels.
DISCONNECTED	Shows a channel that has stopped communication. Event Hub 4 channels also use this term when there is no subscription.

State	Definition
PAUSED	Shows a channel that has paused communication.
FAILED	Shows a channel that has communication issues.

Viewing log events per channel

Use this procedure to view channel-specific logged events on IEC Server's Communication Channels page.

Viewing events logged by a specific channel

1 Access the **Enterprise Collaborator Administration >** <u>Communication</u> page to open the **Channel** control page.

The view shows a table of IEC received channels. A new column is added to the right of the table. This column contains **File** and **Disk** icons for every channel. You can click on the file icons to access the logged events of the indicated channel.

2 Click the **File** icon of a specific channel.

The view shows a new tab containing the channel's logged events. Events from the time the channel was last started or reloaded are logged.

Saving a local copy of events logged by a specific channel

1 Access the **Enterprise Collaborator Administration >** <u>Communication</u> page to open the **Channel** control page.

2 Click the **Disk** icon of a specific channel. A copy of this channel's events is saved to your local Downloads folder.

Events

On this page, you can search for IEC application logs. Several links are available.

Log

The Log link is intended for use as a quick monitoring view. It shows an overview of the latest events registered by the server.

To filter the logs view, select a log level to display from the **Log level** drop-down list. The filter options are **all levels**, **only debug**, **only info**, **only warn**, **only error**, and **only fatal**.

To view messages, select **Show log** from the server start-up box to view the messages created at the time when the server is started.

Log Search

The **Log Search** link allows you to search for detailed server event log. You can use different search criteria and field visibility composing a log event.

Log search contains several search fields:

- **Top** specifies how many records will be returned by the search. The maximum limit is 2000. This search limit is introduced because of possible performance issues when loading a lot of records.
- **UUID** specifies a unique log identifier, used to group similar or related log entries together. For example, each message is assigned a UUID. All logs that are related to a message can be retrieved through this UUID. This also applies to other logs also such as channels and tasks.
- **Server** is information that is masked/hidden for the tenant admin. This specifies what server name or host name that is generated.
- **Priority** allows you to select a standard log level for **DEBUG**, **INFO**, **WARN**, **ERROR**, and **FATAL**.
- **Text Contains** is the actual log message. This field lets you filter based on a match you specified. For example, the log message states "Message has been reactivated from order queue." and a text value of "reactivated" will match this record.
- **Class Contains** searches for the class name that generated the log entry. This is available on the **Log Configuration** page.
- **Thread name Contains** searches for the thread name that generated the log entry.
- **From time** field indicates the start time when searching an entry between a time range.
- **To time** field indicates the end time when searching an entry between a time range.
- **Submit** activates the log search and will generate the results in a separate page.

Log Configuration

The **Log Configuration** provides a runtime control of log levels and default visibility of log searches. With the proper settings, the view can show detailed log information to help trace where a problem occurred in a live system.

Accessing the **Log Configuration** page allows you to submit root level and log level configurations:

- **Set the level for the root logger** is used to determine the level that any new loggers will use.
- **All loggers** set the log level of all current loggers. This does not affect the root logger. Any new logger will still use what the root logger level has.

Note: For Datatranslator, the BEDataTranslatorStarter information about the "Total No. of DataTranslator fetch from BE" will only be shown when the **Events > Configuration** for this class (BEDataTranslatorStarter) is set to **DEBUG**.

You can also access the **Server event log** that contains detailed information about the server operation. The drop-down menu provides several log options used to submit a log configuration:

- **DEBUG** shows fine-grained informational events that are most useful to debug an application.
- **INFO** shows informational messages that highlight the progress of the application at coarse-grained level.
- **WARN** shows potentially harmful situations.
- **ERROR** shows error events that might still allow the application to continue running.
- **FATAL** shows severe error events that might lead the application to abort.

Note: Important warning and error messages always show in the **Log** page, regardless of **Log Configuration** settings.

Utilities

Utilities allow you to perform reloading tasks, property checks, and translation data utilities. Click the **Utilities** link in the **Enterprise Collaborator Administration** page to access this page. There are several pages available.

Datatranslator

The **Datatranslator** page allows you to load translations using M3 Data Translator. You can use reloading functions available in the M3 Data Translator page to retrieve records from M3 Business Engine (M3 BE) and add priorities for the M3 Data Translator schedule.

Access **Enterprise Collaborator Administration > Utilities > Datatranslator** page.

String data going to (outbound), or received from (inbound) a partner can be translated or converted from one form to another. This is called data translation and applies only to string data. It does not apply to the numeric value of a string data. For example, the received string data "KG" can be translated to "kg", but their corresponding numeric values do not change.

On this page where Translations are loaded, you can utilize M3 Data Translator (former known as MECDataTranslator) Controls.

Datatranslator has several features:

- Click the **Reload** link to reload DataTranslator and return immediately.
- Click **Check properties** to include parameters such as user environment, reloading schedule, and translation data expiration in the Datatranslator printout.

- Click **Dump translation data** to return a link to save the translation data in a CSV log file format.

Note: IE retrieves 20,000 records only when downloading translation data. All records beyond the 20,000 limit will be disregarded.

The reload operation retrieves the data from M3 and saves it in the IEC database. This process runs in the background and automatically updates the cache.

When you click **Dump Translation data**, this operation writes the cache to a file for download.

DataTranslator Controls

DataTranslator features several controls that you can use to manage messages, properties and other data.

Reload
Reload is used to manually reload records stored in memory. When running, the updated number of data translations in Business Engine (BE) will be automatically added or deleted in the tenant's database. When performing Dump Translation data, the updated number of data translations will be saved in a .csv file.

You can view the events and reloads in the **Events > Log** page. You do not need to use the manual reload function if you already have reloading scheduled in the schedule tasks list.

Check properties
Displays the current date and time stamp.

Dump translation data
Dump translation data is used to download and save the data translation records stored in the memory.

MBMTRN Update

The **MBMTRN Update** page allows you to generate data in the M3 program Business Message Data Translation. You can Specify Displ (CRS881) for the selected mappings, using the standard API Reference.

Updating MBMTRN

You can generate data in the M3 program Business Message data translation. Specify `Displ (CRS881)` for the selected mappings using the standard API Reference. CRS881 is a Business Engine program that manages records. MBMTRN update is responsible for generating metadata for M3 Business Engine data translations. All user functions in all published business message mappings in Infor Enterprise Collaborator are scanned for data translations when you update MBMTRN.

1   Access **Utilities > MBMTRN Update**.
2   Select **Cumulative**.
    Note: Select the **Cumulative** option to prevent BE data loss.
    Ticking the **Cumulative** checkbox will add or update MBMTRN records. Not ticking the box will delete unused MBMTRN records when you click the **Update** button.
3   Select to update at least one of the API Reference names.
    You can only add or update Mappings with data translation in it.
4   Select a Mapping Name to update.
    Note: All mappings are processed when there are no mappings specified.
5   Click **Update**. A link to the task page is displayed.

Mapper Delivery

The **Mapper Delivery** page allows you to download the Infor Business Document Mapper in either 32-bit or 64-bit version.

You will find the latest version of the IEC Mapper here, including the host and port that serves the download request.

Business Document Mapper tool

The Business Document Mapper (Mapper) client tool is a component within the Infor Enterprise Collaborator (IEC) product. Business Document Mapper is used to create and deploy mappings.

Downloading Mapper

You must download the zipped installation file for Business Document Mapper before you can install it. Follow these steps from the **Enterprise Collaborator Administration** page.

1 Go to **Utilities > Mapper Delivery**.
2 Select the 32-bit or 64-bit version.
3 Click **Download**.

Business Document Mapper client requirements

Business Document Mapper must be installed on premises. This table shows the hardware and software that you must install on your system before you can run and configure Mapper:

Hardware component	Quantity	Notes
Memory	8 GB or more	DDR2 or newer.
Hard disk space	8 GB or more	Hard drives with faster read/write cycles, such as solid state drives, will improve performance.

Software component	Version	Notes
Operating system	Windows 10	English is the preferred language for installing Windows. The ability of Infor to support non-English installations is limited.

Installing Mapper

1 Copy the `mapper-all*win*.zip` file to the machine where Mapper should be installed.

2 Unpack the `mapper-all zip` file to a local folder on the machine. The folder will be referred to as *MAPPER_ROOT* in the installation instructions.

Starting and verifying Business Document Mapper installation

Perform this verification procedure right after installation.

1 Start `eclipse.exe` from the *MAPPER_ROOT* folder.

2 Verify that the Mapping Development perspective is displayed:

    a Select **Window > Open Perspective > Other**.

    b Double-click **Mapping Development**.

    c Specify a user password.

Upgrading Mapper

You can upgrade an existing installation of Business Document Mapper (Mapper) using the Mapper plugins update. If there are changes required to the eclipse.ini or jre subfolder inside the eclipse mapper-all, you need to download and install the available 32-bit or 64-bit installers of Mapper from the Mapper Delivery page.

Use this procedure to upgrade Business Document Mapper.

1 Go to **Enterprise Collaborator Administration > Utilities > Mapper Delivery** page.

2 Click the Mapper plugin updates radio button.

3 Click **Download**.

4 Open Mapper.

5 Click **Help > Install New Software**.

6 Click **Add…**

7 Click **Archive…**

8 Select the mapper-site file downloaded from the Mapper delivery page in the local file system.

9 Click **Open** and **OK**.

10 Select the Mapper feature(s) to be installed.

11 Click **Next** and **Next**.

12 Click I accept the terms of the license agreement radio button and click **Finish**.

13 Wait for the mapper features and plugins to be installed.

14 Click **Yes** to restart mapper.
The updates will take effect after the program restarts.

Configuring Business Document Mapper

You need to configure Business Document Mapper before you can connect to the server. See Business Document Mapper tool configurations and settings on page 177.

Uninstalling Business Document Mapper

Before uninstalling the Mapper, you can export your existing Server preferences by using the export function of Eclipse for later reference.

1   Close the Mapper.

2   Delete the *MAPPER_ROOT* folder.

Settings

You can change communication encryption and date and time settings for IEC Tenant Administrator. A password is needed for changing Encryption/Decryption and this password also serves as the encryption key. Ask your tenant administrator for this password before changing any settings.

Configuring time zone settings

Tenant administrators can change the date format and time zone settings for the Enterprise Collaborator Administration pages by using the Date/Format settings page. Changes will take effect immediately.

Note: The Date/Format settings require tenant administrators to have the IFS role of M3EC-Administrator.

IEC supports Daylight Saving Time (DST). To use daylight saving time, you must use Time Zone Database (TZDB) zone IDs. Do not use three-letter abbreviations. For example, Eastern Standard Time (EST) and Standard Time do not observe DST. These are full time zone names that are used as part of a time zone. Choose a time zone that observes EST with United States daylight saving time rule, such as America/New_York.

You can configure time zone settings by following these steps:

1   Go to **Settings > Date Format Setting**.

2   Specify this information:

Date Format
The time format in year, date, month, hours, minutes and seconds.

Date Locale
The parameters for the language and region of the user interface.

Date Time Zone
The time zone location.

3   Click **Update** on the specified settings.

Encryption/Decryption

The encryption/decryption is a function that allows you to encrypt or decrypt specific elements in your messages. Encrypted values can only be decrypted (and vice versa) using the specified password, which must be set on this page. The supported key size is currently fixed to 128-bits only. 192 and 256-bit encryptions are not supported.

Note: Encrypted values can only be decrypted using a specific password. If the password was lost or changed, then the old messages will not be decrypted.

You must follow these Encryption/Decryption parameters to use the utility.

Aspect	To Encrypt	To Encrypt	To Decrypt	To Decrypt	To check if a string is encrypted
Usage syntax	CryptographUtility.encrypt(plainText)	CryptographUtility.encrypt(plainText, varPwdIterations)	CryptographUtility.decrypt(encryptedText)	CryptographUtility.decrypt(encryptedText, varPwdIterations)	CryptographUtility.isEncrypted(encryptedText)
Parameter	plainText - Is any string/element selected to be encrypted.	• plainText - Is any string/element selected to be encrypted. • The number of iterations to be used when deriving the key.	encryptedText	• encryptedText - Is any string/element selected to be decrypted. • The number of iterations to be used when deriving the key.	encryptedText - Any string that is to be checked, if it is encrypted or not.
Return value	encryptedText	encryptedText	decryptedText (or String)	decryptedText (or String)	true - If encrypted; otherwise, false.

You can encounter different error messages depending on the issues encountered by the utility.

Error Message	Description
Please check the input element.	This error shows during decryption, when the utility is unable to understand or decode the encrypted text.
Unexpected input element format.	This error shows when the element to be decrypted is not in the expected format.

Error Message	Description
Encoding is not supported.	This error shows if the element contains an unsupported character which the utility is unable to handle.
This element might have been encrypted with a different key. Unable to decrypt.	This error shows when the key used to encrypt is not the same as the key used to decrypt.
Password Error.	This error shows when there is a problem with trying to retrieve the password.
Password is missing or empty.	This error shows when the password has not been defined in IEC.
An error occurred while encrypting/decrypting.	This error shows when a general, but technical or internal problem during the encryption or decryption process occurs.
Key generation error.	This error shows when a technical internal problem occurs during key generation.
Initialization error.	This error shows when a technical internal problem occurs during initialization.
IV length is incorrect.	This error shows when the used Salt or IV does not meet the specifications.

You can use encryption/decryption by following these steps:

1   Go to **Settings > Encryption/Decryption**.
2   In the Set Encryption/Decryption Password, specify this information:

Password
Specify the encryption/decryption key. Do not lose this information as you will be unable to decrypt messages encrypted using this key.

3   Click **Save**.

About

On this page you can view the version information of Infor Enterprise Collaborator.

Chapter 15: Business Document Mapper user tasks

This chapter provides an overview of the functions, usage, and management of the Business Document Mapper (Mapper) client tool.

This tool is a component within the Infor Enterprise Collaborator (IEC) product.

These topics are for Mapper client tool business consultants, application engineers, and Java developers who create mappings for transformation of XML documents on the runtime server.

To use Mapper, you must be knowledgeable and experienced in these areas:

- Mapping and its prerequisites
- Programming concepts such as functions and input/output parameters, loops, and execution flow control
- XML concepts such as elements, attributes, namespaces, and schemas
- Java programming
- Eclipse IDE Framework

You can use this as a primary reference on how to create mappings for transformation to XML documents on the runtime server. These transformations are available through the IEC Server.

To access **Business Document Mapper**, download the installer from Enterprise Collaborator administration and install in your local machine. Follow the steps in Business Document Mapper tool on page 166 for installation.

Business Document Mapper conventions and terminologies

This chapter uses specific text conventions and terminologies, which are detailed in this section.

Name

When prompted for names, do not use white spaces or quotation characters within names.

Naming example:

- Acceptable: `MyMapping, my_mapping, myMapping`

- Not acceptable: `My Mapping, Peter's Mapping`

Selection

A target icon or a dashed border indicates a selected mapping element.

Search filter

Some tasks in this tool use wizard pages or views with **Search** filter option.

When you search for document elements, you have these options:

- Specify a name that will locate all element that matches that name.

- Specify a path on the `/A/B` format (as denoted by the `A/B = path` expression before the filter box).

The second option, for example, "`/SyncCreditTranfer/DataArea`" will locate all elements under that path, for example, "`/SyncCreditTranfer/DataArea/AnElement`" and `/SyncCreditTran fer/DataArea/AnotherElement`".

Search filter guidelines:

- To display all data, leave the filter field blank and press Enter.

- Use an asterisk (*) character in place of a sequence of unspecified characters, for example, `MyMapp*`.

- For filters used together with a multicolumn list, the filter is applied across columns.

  For example, to see all mappings containing the phrase `AutoBank` in the `PBM` category, specify the expression `AutoBank*PBM`.

Name	Version	Category	ID
AutoBank_XYZ	1.0	PBM	137
AutoBank_QRT	1.0	PBM	149
MyMapping	1.0	TEST	2

This table shows the traffic light icons that indicate the status of writing output to the Mapping Console view:

Traffic light icon	Status
Red light	The last command failed.
Yellow light	The command is running, or the last command gave a warning.
Green light	The last command was successful.

For example, these light icons are displayed when you publish a mapping:

- While the mapper server publishes the mapping, the traffic light is temporarily **yellow**.

- If successfully published, the traffic light turns **green**.

- If not successfully published, the traffic light turns **red**.

The state of the traffic light stays until you run the next command. The traffic light always reflects what is logged in the Mapping Console view. For example, if you run the command to clear the console, the traffic light will always be green because the clear command is always successful.

Some **Validation** and **Broken Links** tabs show the result for a document or a mapping validation. If issues are detected, a table list of category label and detailed descriptions is displayed. Select an issue to display more details about that issue.

Business Document Mapper is used to create and deploy mappings. On a high level, a mapping is a description of how to transform or translate messages or documents that are sent between internal and external resources. On a detailed level, a mapping is an XML file expressing a transformation between input and output elements, and attribute structures defined by XML schemas representing the input and output messages and documents. Business Document Mapper provides the capabilities to define mappings, for example, for OAGIS BODs, EDI, EBS, financial messages, and others.

Mappings are organized in projects. These projects can contain new or imported mappings. To define the transformation, we use a graphical editor called Mapping Editor.

In the Mapping Editor, the left and right panes show the elements and attributes of the input (left) and output (right) schemas as tree structures. The middle pane shows the transformation expressed as a sequence of connected mapping elements that are categorized as functions, variables, or constants. An empty mapping has a blank middle pane.

After a mapping is completed, deploy the mapping to the Mapper Server runtime environment. Deployment is the process of publishing a mapping to a Mapper Server. The Mapper Server stores the mapping in a Mapper Database and generates a runtime Java class. The Mapper Server uses activated runtime Java classes for message translation. Mappings can also be imported to the mapper from the Mapper Database through the Mapper Server.

This diagram shows the interaction among Business Document Mapper, database, server, and messages:
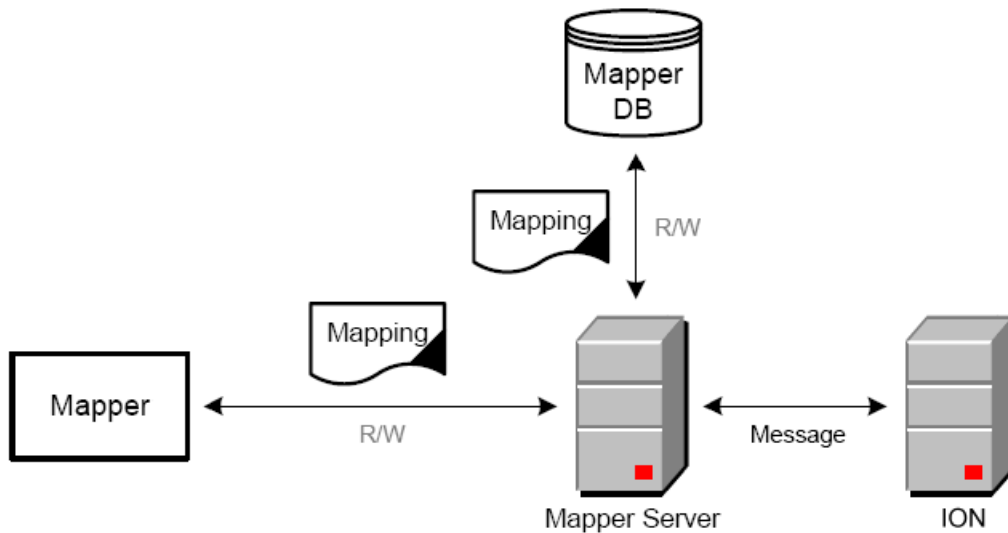
The Business Document Mapper can help you achieve these tasks:

- Create mappings.
  - Select predefined functions or create new user-defined functions to map elements and attributes in the input schema to a number of input parameters in the functions.

    The output from the functions is mapped to the elements and attributes according to the output schema.
  - A completed mapping is internally saved in XML format, that is the .map file.

    The file contains all the information needed by the Mapper Server runtime environment.
- Deploy mappings to the Mapper Server.
  - Publish the mapping to a Mapper Server. The Mapper Server stores the mapping in the Mapper Database and generates a runtime Java class.

These concepts explain the mapping generation and publish lifecycle states of a mapping.

This overview diagram shows that you use Business Document Mapper to create and publish a custom mapping. You must configure IEC to use it through Infor Ming.le.

Process steps	Details
1. Create a mapping.	• Create a mapping in Business Document Mapper, or import an existing mapping from a .zap file or from a mapper server. • The mapping now exists as files in your Eclipse workspace. • The mapping is still a standard mapping. The mapping property Mapping. Type is equal to Standard. When renaming, copying or editing the mapping it will become a custom mapping. The mapping property Mapping. Type will then be equal to Custom.
2. Save the mapping to the mapper server.	Optionally, save a custom mapping to the mapper server without generating or publishing it. • The mapping now exists as metadata in the mapper database. • At this stage, if this is a new mapping, no runtime mapping class exists. **Note:** A standard mapping cannot be saved to the mapper server.

Process steps	Details
3. Generate the mapping.	Optionally, generate the mapping. The mapping is automatically saved in the mapper database. • The mapping now exists as metadata in the mapper database. • At this stage, if this is a new mapping, no runtime mapping class exists. • A runtime mapping class is generated and compiled, but the runtime mapping class is not inserted into the mapper database. Generate is used for testing purposes only. Use this to verify that you can generate the mapping, compile the mapping, and that the data translations are correctly defined. Generate does not affect runtime. Although, the mapping metadata in the mapper database is updated.
4. Publish the mapping.	The mapping is automatically saved in the mapper database. • The mapping now exists as metadata in the mapper database. • A runtime mapping class is generated and compiled. • The runtime mapping class is inserted into the mapper database.
5. Update the mapping in Business Document Mapper.	• Use this to update the mapping in Business Document Mapper. • The updated mapping now exists as files in your Eclipse workspace. **Note:** If you change the mapping's version number, the updated mapping will be handled in the same manner as a new mapping; see step one. When updating a standard mapping, it is automatically converted into a custom mapping.
6. Save the updated mapping to the mapper server.	Optionally, save the mapping to the mapper server without generating or publishing it. • The updated mapping metadata is now stored in the mapper database, but the runtime mapping class is still generated from the old **Source**.
7. Generate the updated mapping.	This step is optional. See step three.
8.Republish the mapping.	The mapping is automatically saved in the mapper database. • The mapping now exists as metadata in the mapper database. • A new runtime class is generated and compiled.

This diagram shows the mapping states and the available actions for custom mappings:
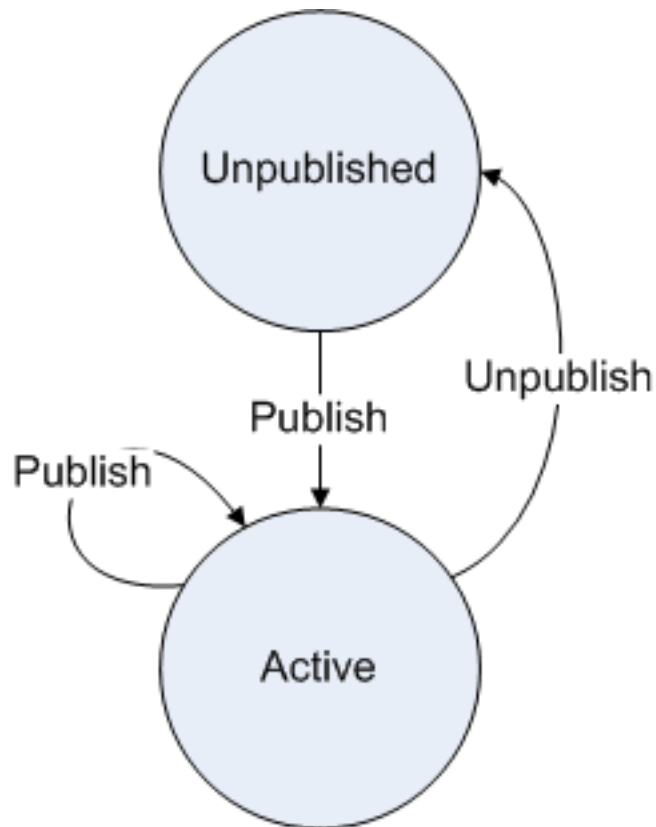
Business Document Mapper tool configurations and settings

Mapper tool use requires you to know several configuration and setting procedures.

Business Document Mapper configuration overview

This table shows the settings that you need to complete after you install Business Document Mapper:

Setting	Description
Memory Settings	Control the maximum memory setting that can be allocated at the start to ensure that Mapper has sufficient amounts of memory.

Setting	Description
Server Connectivity	Set the server connection to save, remove, generate, and publish a mapping to Mapper Server.

Business Document Mapper configuration

This table shows the settings that you need to complete to help with your mappings:

Settings	Description
Type Definitions	These are data type definitions you can use on your functions parameter, variables, and constants.
XML Document Editor	You can use the external XML schema editor in addition to Mapper editor.
Import/Export Settings	You can use default folders when importing or exporting mappings.

Configuring memory settings

Mapper is preconfigured with a maximum memory allocation of 1GB. In most cases, 1GB should be sufficient. If you work with huge mappings, you might encounter an OutOfMemory error. If this happens, increase the Mapper memory allocation.

1   Create a shortcut link to this file: `MAPPER_Install_Folder`/eclipse.exe.
2   Edit the Target property of the link. Use this format:

`MAPPER_Install_Folder`/eclipse.exe -vmargs –Xms*MIN*m –Xmx*MAX*m

where:

- *MIN* is the minimum amount of RAM in megabytes, for example 512
- *MAX* is the maximum amount of RAM in megabytes, for example 1000

Example Target format: `IBDMapper/eclipse.exe –vmargs –Xms512m –Xmx1000m`

Saving Business Document Mapper preferences

When you have made a change on the **Mapper preference** page, you must save it. Follow these guidelines:

- Click **Apply** to save your changes without leaving the current preference page.

  Or

- Click **OK** to save your changes and close the preferences dialog.
- If you switch to another preference page without clicking on **Apply**, your changes will not be saved.

- If you click **Cancel** to close the preferences dialog, without first clicking **Apply**, your changes will not be saved.
- If available, click **Restore Defaults** to restore the default settings for the preference page.

Setting the server connectivity

Set one or several server connections to save, remove, generate, and publish a custom mapping to a Mapper Server, and to import a standard or custom mapping from a Mapper Server.

The server connection is also used to save, remove, activate, and deactivate a custom flat file definition to a Mapper server, and to import a standard or custom flat file definition from a Mapper server.

OAuth2 User Authentication

Business Document Mapper uses OAuth2 specification to delegate access using Ming.le credentials to the application. This added layer of authentication in addition to authorization, allows for the management of end user information control and consent. OAuth2 has the benefit of adding security against access tokens injected by an outside party and, provides differentiation between authentic access tokens and malicious attack tokens introduced by attackers trying to impersonate a user.

Configuring Server Preferences

Business Document Mapper requires you to setup OAuth2 in **Preferences** before using it for the first time.

There are two requirements for setup:

1   The ION API file.
2   Valid Ming.le credentials.

A limitation with using the OAuth2 protocol is the 1-minute ION gateway connection timeout between the client and server. If the connection lasts longer than 1 minute, ION gateway will terminate the connection. After the connection has been terminated, Mapgen will still continue to process the request until it succeeds or fails. Token-based connection addresses this issue by constantly checking for connectivity. If the connection fails, Mapgen will no longer continue to process the connection and you won't have to wait for the connection timeout issue.

A token-based asynchronous connection to Mapgen is used for server operations that may take longer than one minute to complete, for example, to publish a large mapping. If needed, you can extend the timeout to wait for Mapgen to complete the operation, just as before. When the Mapgen processing is complete, the result is shown in the mapping console as usual. The 1-minute ION gateway connection timeout should not affect this functionality.

Note: The IFS security role M3EC-ClientDesigntime is required for a user to connect the mapper to a server. See Required IFS security roles on page 16 in the M3 CE Core Administration Guide.

Follow these steps to configure Server Preferences.

1   Select **Window > Preferences > Mapper > Servers**.

2   Click **New**.

3   Specify this information:

Tenant ID
The Tenant ID is automatically retrieved from the ION API file.

Timeout (ms)
Specify the timeout in milliseconds, before the connection registers a timeout connection failure.

Load ION API File
Use the ION API file that has been downloaded from the Infor ION API web page. See the M3 CE
Core Administration Guide for more details on how to download the ION API file.

4   Click **OK**.

5   Select your Configured server location.

6   Click **Test Connection...**.

7   Login using your IFS credentials.

8   Click **Allow**.
The test results will be shown.

9   Click **OK**.
You now have an access token.

Adding a custom authorized application

You need to manually create a custom authorized application to download the ION API file, when you
configure OAuth2 for the first time. You will find this option in the Authorized Apps page in the Infor
ION API page in Infor Ming.le. Follow these steps to create your custom authorized application.

Note: You must have access to ION API for this option to be available. You have to create a custom
authorized application to be able to download the ION API file, when you configure OAuth2 for the first
time. This needs to be done once per tenant.

1   Go to **Infor ION API** page.

2   Click **Authorized Apps**.

3   Click **+**.

4   Specify this information:

Name
Specify the name of the authorized application.

Type
Select **Web** as the type of application.

Description
Enter a short description of the application.

Redirect URL
Specify **https://www.infor.com**. This URL is not mandatory and you can use any arbitrary URL.

Authorized JavaScript Origins
Specify * in this field.

OAuth2 Access Token
Specify 2 hours by default.

5   Click the **Save** icon.

Downloading the ION API file

You must download the ION API file when you configure OAuth2 for the first time. You will find this file from the Authorized Apps page in the Infor ION API page in Infor Ming.le. Follow these steps to get the ION API file.

Note: The ION API file is used by all users that need to connect the mapper to the server. You do not need to create or download a separate ION API file for every user. You need to create one ION API file per tenant. You must have access to ION API for this option to be available.

1   Go to **Infor ION API** page.

2   Click **Authorized Apps**.

3   Click or search for Infor Business Document Mapper (M3), or the name that is specified in step 4 in Adding a custom authorized application on page 180.

4   Click **Download Credentials**. Before downloading your access credentials, you can specify the timeout duration of the access token, to extend the period of access timeout.

Performing server connectivity tasks

Set one or several server connections to save, remove, generate, and publish a custom mapping to a Mapper Server and also to import a standard or custom mapping from a Mapper Server.

Perform these tasks from the **Server Connectivity** page:

1   Edit an existing server connection:

   a   Select a server connection to modify and click **Edit**.

   b   In the New window, specify the timeout and click **Load ION API File**.

   c   Select the ION API file and click **Open** and **OK**.

   d   Verify that the new connection is listed under the Configured server locations and click **Apply** and **OK**.

2   Remove a server connection:

   a   Select a server connection to disconnect and click **Remove**.

   b   Verify that the removed connection is no longer listed under the Configured server locations.

3   Set a default connection:

   a   Select a server connection and click **Make Default.**

   b   Verify that the connection is listed as the Default location.

4   Copy to clipboard:

Use this function to copy **ION API Preference** to the clipboard.

5 Test a connection.

This option will verify the connection to the Mapper Server. The version of the Mapper Server is shown.

Type definitions property

You can add predefined type definitions through the `mapping.types.default` property in the `mapper.properties` file.

Setting the type definitions

Use this procedure to add new or edit the existing type definitions for parameters, variables, and constants.

1 Select **Windows > Preferences**.

2 Expand the view of Mapper and select **Type Definitions**.

3 Click **Add**.

4 In the Type Definitions table, select **New Type** to display the type definitions page.

5 Specify the name and description for the new data type.

6 Optionally, click **Edit** to change the value of an existing type definition.

7 Optionally, click **Remove** to delete an existing type definition.

8 Optionally, click **Restore Defaults** to restore the default type definitions. All type definitions that you have added will be deleted.

9 Click **OK** to close the **Preferences** window.

If a mapping uses a type definition that does not exist, the type definition is automatically added when the mapping is opened.

Setting the XML Document Editor

Use this procedure to define an external XML schema editor to open your mapping as an XML document. This is the recommended way to edit a schema.

1 Select **Window > Preferences**.

2 Expand the view of Mapper and select **XML Document Editor**.

3 In the **Editor** field, specify the complete path to your installed external XML schema editor.

4 Click **OK** to add the external editor.

Getting started with Business Document Mapper

When you start Business Document Mapper, Eclipse opens to the Mapping Development perspective with customized views for working with mappings. The first time you start Business Document Mapper, you must manually select the Mapping Development perspective through **Window > Open Perspective > Other**.

This diagram shows Mapping Development Perspective with a mapping in full view:
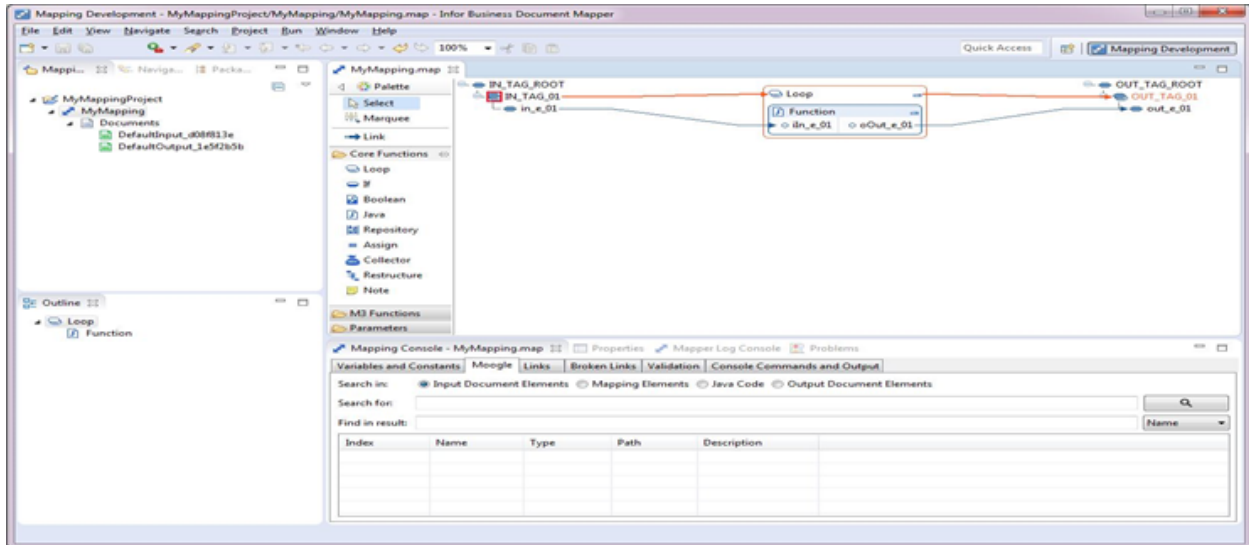
Note: To optimize loading time, the Mapping Editor shows only the required and top-level schema elements that are connected to mapping elements.

- To see all connected elements, expand the tree view.

- To drill down and see non-connected elements, expand individual schema elements.

- To define a mapping, access the Palette and select tools from the drawers.

- Running an older Mapper version on a newer workspace will cause a 'Could not create the view...' message. Avoid this issue by not using older versions of Mapper on a newer version workspace.

Using the Editor area layout

On the Mapping Editor, you can view the input, mapping, and output sections of a document. Allocation for view space is changed in the `Mapping.Layout` property on the Properties tab. Changing view space allocation is useful for documents, for example, with long element names.

1 Click the **Properties** tab.
2 Double-click `Mapping.Layout` to edit the spacing.

   Three sets of numeric values represent the editor area split view in percentage:

   <INPUT_AREA_SIZE MAPPING_AREA_SIZE OUTPUT_AREA_SIZE>

3 Specify the values for the mapping editor spaces.

For example, specify the values *20 60 20* to display 20% for input, 60% for mapping, and 20% for output.

Note: A more convenient way to set this property is to select **Set Width** from the Input and Output Document context menu to change the *<INPUT_AREA_SIZE> and <OUTPUT_AREA_SIZE>* view space.

Mapping context menus

Depending on the level of information that you want to display or work on, you can use the Mapper context menus to help you complete mapping tasks.

You can use these options in the Mapping Editor context menu:

Menu option	Description
Properties	Shows the properties
Mapping Console	Shows the Mapping Console
Hide or Show Variables and Constants	Hides or shows all assigned variables and constants
Hide or Show Unconnected Parameters	Hides or shows unconnected function parameters
Expand All	Expands the complete mapping **Note:** Collapse all loops, functions, and document nodes one by one to collapse an expanded mapping.
Show In Mapping Explorer	Finds the mapping in the Mapping Explorer view Select **Show In Mapping Explorer** to find the current mapping in the Mapping Explorer view. You can use this menu to find the mapping, for example, when you have several mappings with similar names or several mappings with the same name that are located in different projects.

You can use these options in the input and output document's context menu:

Menu options	Description
Hide or Show Unconnected Elements	Hides or shows unconnected elements and attributes in the document

Menu options	Description

Expand or Collapse Document	Shows complete input/output tree view:
	
	 1 Select input or output document.
	
	 2 Select an element view option.
	   • Select **Expand Document** to show all connected schema elements in the tree, plus all schema elements that you have expanded while editing the mapping.
	   • Select **Collapse Document** to show only the top two level elements of the tree.

Set Width (%)	Sets the width of the document areas: **1** Select input or output document > **Set Width**. **2** Optionally, you can select **Default**. The width of both the input and output document areas is set to 20% each, with the remaining 60% for the mapping area. The percentage corresponds to the default value 20 60 20 for the mapping property `Mapping.Layout`. These settings are stored with the mapping. Use these rules for the width option: • Minimum document width is 15%. Anything less than 15% is not usable. • Maximum document width is 50%. Exceeding 50% in one area makes the two other areas too narrow. • Minimum width for the mapping area is 20%. Anything less than 20% is not usable. • Total width for input document area + mapping area + output document area must always be equal to 100%. Some widths are dependent on the width setting of other document areas.

Mapping Explorer view

In the left pane, **Mapping Explorer** tab shows an artifacts-oriented view of mappings. The artifacts shown are mapping projects with mappings. Each mapping has a Documents node that shows the input and output schema documents. Each artifact, or node, has a context menu.

The project context menu includes these options:

- Select **Refresh** to update the mapping project with recent changes.
- Select **Refresh All** to update all mapping projects in the workspace with recent changes.

Business Document Mapper views

Use these procedures to display views to help you monitor or find your way around your mappings. The tabs in each view contain features to help you search for elements in a mapping, or filter log expressions.

Displaying the Mapping Console view

Navigate a mapping or display the progress of mapping operations through the Mapping Console view. This view opens automatically when you open a mapping feature with progress feedback, for example, when you import mappings from a Mapping Server.

1   Right-click an area in the Mapping Editor and select **Mapping Console**.

The **Mapping Console** view is displayed as a new tab with sub-tabs. In every sub-tab, except for Console Commands and Output, you can use **Search** to filter the list.

This table shows the available tabs and a description on how to use the tab:

Tab name	Description
Variables and Constants	The default opening tab for **Mapping Console** view. Access the tab to maintain variables and constants, and to search for variables and constants in a mapping. **a** Click **Search** to filter the variables and constants in the mapping. **b** Double-click a row to reveal the location of a variable or constant.
Moogle	Contains search functionality for the mapping. This tab replaces the tabs **Input Document Elements, Functions and Parameters**, and **Output Document Elements** in previous versions of Mapper. Searching in a mapping on page 187
Links	Contains all the links in the mapping. Access this tab to select a link in the editor when working with huge and complex mappings. To select a link, double-click a link in the Mapping Editor. To delete a link use the context menu.

Tab name	Description
Broken Links	Access this tab to show the broken links in a mapping.
	A link cannot be made when listed under the Broken Links tab. Failure to link may be due to a missing source or target element after a document was changed for mapping. The missing source or target element is indicated by a question mark, and the detailed information is displayed in the list. Broken Links tab also shows the information about the internal IDs of the missing elements and the cause of the missing elements.
	For advanced users, use the internal ID information to verify the reason for the missing source or target element. Use the internal ID information with a previous version of the mapping file (.map).
	Double-click a link to select the existing source or target in the Mapping Editor.
	The broken links remain in the mapping until you discard them with the Discard option in the context menu.
	Note: Saving a mapping to database will fail if there are broken links.
Validation	Shows the result of document or mapping validations. If issues are detected, a list with detailed descriptions is displayed. Select an issue from the list to show more specific details about the issue.
Console Commands and Output	Access this tab to view mapping operation progress and to access additional commands, if needed.

2   Save your work.

3   Optionally, close the **Mapping Console** view.

Searching in a mapping

Moogle tab contains search functionality for the mapping.

1   Click the **Moogle** tab in the **Mapping Console** view.

2   Select a category for the Search in field:

Category	Search in field
Input Document Elements	To search in elements and attributes for the input document.
Mapping Elements	To search in mapping elements, for example loop, function, and parameter names.
Java Code	To search in the Java code for all UVJ and UVB functions.
Output Document Elements	To search in elements and attributes for the output document.

Note: When you select a category, a new search is completed. You can use the same search string in different categories.

3 Specify a search string in the Search for field:

Search for field	Search string
Search for document element name or a parameter name.	The search string is case insensitive and matches part of element names, for example, a search for **bc**, matches ABCD.
Search for unknown string.	Use the wildcard character * (asterisk) in place of an unknown string, for example, a search for **b*d** matches ABCD, bed, and abd-001.
For advanced users, search using regex.	Regex are case sensitive. For example, to search for all elements with the name ID, specify the search string **#.*/ID** where: • **#** is the regular expression indicator. • **.*** indicates any character zero or more times. • **/ID** indicates the end of the XPath.

The paths for the elements are used for searching, not the element names. For example, if the name of a loop matches the search string, all functions and their parameters and inner loops within the loop are returned as search result because the name of the loop is part of their paths.

4 Click **Search**.

The search result is displayed in a table list. The number of matches is shown in the status bar at the bottom of Eclipse.

5 Optionally, to further refine the search result:
   a   Specify a search string for the search result in the field **Find in result**.
   b   Click the drop-down arrow to select all fields or a specific field.

This search filter applies to one or all of the columns. The search string is case sensitive and will match part of the column values. You can use the wildcard character (*) asterisk in place of an unknown string.

Search result

This table shows the search areas in the **Name** field drop-down list and the information in each search area:

Search area	Details
Index	This area shows the element icons and the chronological index order of the element. You cannot search data in this column. Example of icons include a repeating element, a UVJ function, or a parameter.

Search area	Details
Name	This area shows the element names in a mapping. This name does not need to match the search string. This name can be for a parent element in the element's path that matches the search string.
Type	This area shows the type of the element, for example, Doc Input or Doc Output.
Path	This area shows the path of the element. An element in this path matches the search string.
Description	This area is the description of the element. Multiline descriptions are shown in tool tips.

To find an element in the mapping editor, double-click an item in the table. The mapping editor expands the view of parent nodes, scrolls to the element, and selects the element.

Data for input and output document elements and attributes are read into memory only when they are used. Document elements and attributes in use means that they are connected or expanded in the current editing session. Also called "lazy loading", this method is used to avoid too much memory usage, for example, when you store data for a complete BOD schema in memory.

When searching for input or output document elements, search runs on both the memory and schema including all unexpanded elements.

These are the indications and details for search match on elements and attributes:

- In memory, search matches are indicated by the icons as used in the document tree.
- In the schema, search matches are indicated by a document icon. The same icon is used for schemas in the Mapping Explorer view.
- In the schema, search matches are not shown if they are in use.

When you pause on the field headers, tool tips are displayed with brief details on how to run search.

When you pause on an item in the table, more information on the item such as description or data type and length may be shown in a tool tip.

When you search on schemas, all schemas known for a mapping are searched including custom schemas, if any. If there are matches in such schemas, the returned paths will be relative to the root element(s) of that schema and not relative to a root element of the input or output schema. In this case, the path is regarded only on the schema where the element is defined. If you double-click the result row, search does not locate the element in the mapping editor.

Mapping Console commands

The commands in this view are aids in error analysis and are intended for advanced users and tool developers only. This is not intended to be used by regular end-users.

For command details, write `help`.

Note: Some of these commands still use the old way of communicating directly with the mapper database, and not the current communication directly with the mapper server.

Command	Details
clear	Clears the console
genadm	Lists stats for or purges generator connections
genloc	Lists and selects the known mapping generator locations
save	Saves a mapping to file using normal or classic format
version	Prints the version of the Mapper Editor Feature
xdstat	Analyzes and prints stats about an XML schema (tool developer aid)
linkstat	Prints supplementary information about broken links
history	Prints the history of the Mapper Editor feature
editor	Enables or disables visual guides. border_on displays a border around the sequence
dump	Dumps the Epoxy model to file (tool developer aid)
utildoc	Opens an external browser with Java Documentation of the EC Utility classes
mcu	Short for "mapping clean up" Cleans up a mapping from unused replacement elements and schemas developed from within custom namespaces.
keys	Prints a list of the current shortcut keys available in the Mapping Editor
scroll	Enables or disables auto scrolling of the output document. Sometimes it is convenient to disable scrolling when you create links where the source and target are vertically far apart
paths	Dumps all the paths of a document
settings	Prints the editor settings
pmu	Exports and imports mapping elements
validations	Lists the known validations done by the mapper
jsa	Lists the JSA inclusion list in the command console

Command	Details
`jsa u`	Retrieves the latest JSA inclusion list from the default server location and lists them in the command console
`migrate`	Converts on-premise API to compatible IEC Cloud API.

Viewing the Java documentation and IEC utility classes

You can access the Java documentation though Business Document Mapper to view the utility classes. Follow these steps to access the online documentation:

1   Open a mapping project in Business Document Mapper.
2   Right-click the mapping project and click **New Mapping** or open an existing mapping.
3   Click the **Console Commands and Output** tab.
4   Optionally, specify **help** in the console input and press **Enter**.
    This will display the list of help commands and their corresponding functions.
5   Specify **utildoc** in the console input and press **Enter**.
    This will open the Java documentation page using your default web browser.

Displaying the Mapper Log view

The commands in this view are for tracing and debugging purposes only. These commands are intended for advanced users and tool developers only. This is not intended for regular end-users.

1   Select **Windows > Show View > Other**.
2   Expand the view of Mapper, then select **Mapper Log Console**.
3   Click **OK** to open the **Mapper Log Console view** tab.

    This table shows the tabs and the information on tasks that you can do on each tab:

Tab name	Description
Log	On this tab, you can perform these tasks: • View the path to the log file. This path is editable, but do not change the default value. Click the **Load** button for the Log file field to show the current log. • Select **Search** to filter your logs. • Select **Insert mark** to put in a marker in the log file before you generate large amounts of log traces. A marker helps trace the actions causing an issue. • Select **Clear log** to remove logs and traces. • Select **Copy to clipboard** to save a copy of the log file to another location. • Select **Save current filter as default** to reuse this filter details as default.
Loggers and Configuration	On this tab, you can perform these tasks: • View the path to the Configuration file. • View the path to the Default log level file. • Specify search filter strings to short list your logs. • Set the logger level and select to apply to the current active or to all mappings. Logger level options are FATAL, ERROR, WARN, INFO, and DEBUG. • Select **Save Current Levels as Default** to reuse this filter details as default.

Mapping projects

Before you start working with mapping projects, ensure that you have configured the Mapper.

Creating a mapping project

Create a mapping project to contain the mappings that you will create later.

1   Right-click an area on the **Mapping Explorer** tab and select **New Project**.

2   Specify a name for the project.

3   Click **Finish**.

On the **Mapping Console** tab, the status of the new project is displayed. If there is an error, the message is displayed on this tab.

On the **Mapping Explorer** tab, the new project is listed.

4   Optionally, click the **Navigator** tab and expand the view of your project to view the default contents.

Note: When you create the first mapping project, an additional project called `MappingUtils` is automatically created if this folder does not exist in the mapper workspace.

MappingUtils is a resource project that is referenced from your mapping projects. You can ignore this project. If you delete `MappingUtils`, it is restored the next time you restart mapper or create a new mapping project. This project is not displayed in the Mapping Explorer view.

On the **Mapping Explorer** tab, you can perform these project-related tasks:

*   Create several mapping projects and later select the project where to add mappings.

    See

*   Import existing mappings into your project.

    See

Mappings

Before you start working on mappings, ensure that you have configured the Mapper.

Standard Mappings are mappings provided as business solutions. These mappings cannot be updated or removed from the Mapping Database, nor they can be republished or unpublished in the Mapper Server. However, they can be imported from server and modified in the mapper workspace by renaming or copying. In this process, they are converted to Custom Mappings.

Custom Mappings are mappings that are created by the user, imported from mapping export (zap) file, renamed, or copied from Standard Mappings for custom usage such as a modified business solution to meet specific business requirements. You can save, remove, publish, and unpublish mappings in the Mapper Server.

Opening mappings

You can open mappings in Mapping Editor or in XML Editor.

Opening a mapping in Mapper Editor

1   On the **Mapping Explorer** tab, right-click a mapper and select **Open Mapping**.

A new tab for this mapping opens in the Mapper Editor area.

2   Edit the mapping.

3   Save the mapping.

Opening a mapping in XML editor

The option to open the mapping in the external XML editor is only for advanced users. Directly editing the map file can destroy the mapping.

1   Define an external XML editor in the **Window > Preference** menu option.
2   On the **Mapping Explorer** tab, right-click and select <u>Open With External XML Editor</u>.
    The metadata file for the mapping is displayed in the external XML editor.

Mapper key preferences

You can view and edit Mapper key shortcuts used throughout Business Document Mapper. Follow these steps in Mapper to access the window.

1   Select **Windows > Preferences**.
2   Click **General > Keys**.
3   Select **Mapper** from the Scheme drop-down menu.
4   Click **Apply**.
5   Click **OK**.
    The mapper key shortcuts will be active after the settings are applied.

Creating a mapping

Before you can create a mapping, you must have a project to contain mappings. Additionally, you must read and understand the requirements for using complex schemas in a mapping.

Schemas can be complex and can import or include other schemas. If you use complex schemas in your mapping, the mapping must meet these requirements for the Mapper to correctly import all the relevant schemas into the Mapping project:

*   All referenced schemas must be present on disk. References to non-local schemas are not valid, for example, `http://some/url/somewhere/aschema.xsd`.
*   All references must be relative to the folder with the root schema.

    For example, if root schema `A.xsd`, import schema `B.xsd`, and `C.xsd` are located in sub-folders `FB` and `FC` to the folder `A`, the reference paths must be `/FB/B.xsd` and `/FC/C.xsd` respectively.

    Relative paths against the root schema location are also valid, for example, if schemas A, B, and C are located in this structure, then B may import C as `../C.xsd`

    `MY_FOLDER/A.xsd`

    `MY_FOLDER/SUB_FOLDER/B.xsd`

    `MY_FOLDER/SUB_FOLDER/C.xsd,`

Follow these guidelines:

*   If the schemas do not meet these criteria, you must edit the schemas before you use the schemas.
*   The mapping name must be a valid Java class name.

Creating a new mapping

1 In the **Mapping Explorer** view, select a project to contain this mapping.

Right-click the project, select **New Mapping**.

2 Specify this information:

Project

Specify the project name.

At the start, if you right-click a project, the **Project** field is automatically populated with the project that you selected.

To specify another project, click the browse button for the **Project** field. Click **OK**.

Mapping name

Specify a descriptive name for the mapping. The default name is `MyMapping`.

Input Schema

This is a required field with a default schema name: `DefaultInput`.

To accept the default input schema, click the **Default** button for the Input schema (mandatory) field.

To change the default input schema, click the **Browse** button for the Input schema (mandatory) field. Click **Open**.

Output Schema

This is an optional field with a default schema name: `DefaultOutput`.

To accept the default output schema, click the **Default** button for the Output schema (optional) field.

To change the default output schema, click the **Browse** button for the Output schema (optional) field. Click **Open**.

3 Click **Next**.

4 Select a root element for the input and output schemas.

The input and output schemas may contain multiple element definitions. Select a root element for the input data structure and a root element for the output data structure.

For example:

Default Element `IN_TAG_ROOT`

Default Namespace `http://www.intentia.com/MBM`

5 Click **Finish**.

In the **Mapping Console** view, the status of the new mapper and error messages, if any, are displayed.

In the **Mapping Explorer** view, the new mapping is listed under the project where it belongs. To view the contents of your new mapping, expand the nodes under the project folder.

When successfully completed, the new mapping is added to your selected project. On the Mapping Editor area, the input schema in the left pane and the output schema in the right pane are shown. The space in the middle is for a sequence of mapping functions that you will add later on.

Setting the mapping schema categories

When you create a mapping, its default values for the properties `SchemaIn.Category` and `SchemaOut.Category` are set to **Other**. Use this procedure to set and change the default schema categories.

1 Open a mapping in the Mapper Editor.

2 Right-click an area in the Mapper Editor. Select **Properties**.

3 To set the mapping schema:

  a Click the **Properties** tab, and select the `SchemaIn.Category` property.

  b Select a value type for this property.

  c Repeat steps **a** and **b** for the `SchemaOut.Category` property.

  Note: If a schema is a BOD schema, its category should be set to BOD. This will enable some functionality in the mapper. The schema categories are also important for customized BOD mappings.

Mapping key shortcuts

You can utilize these shortcuts while editing mappings. These shortcuts are set using the Mapper key preferences on page 194.

Function	Hot key
Move up	Ctrl + Shift + U
Move down	Ctrl + Shift + D
Add input parameter	Ctrl + Shift + I
Add output parameter	Ctrl + Shift + O
Rename	Ctrl + Shift + F2

Copying mappings

Use this procedure to copy a mapping from another or the same mapping project. The progress and result of this action are logged on the **Mapping Console** tab.

Note: The new mapping name must be a valid Java class name.

1 On the **Mapping Explorer** tab, right-click a mapping to copy. Select **Copy**.

  The **Copy Mapping** window is displayed.

2 Click **Browse** to select a Mapping Project for this copy.

3 Type a new name for this copy, or accept the default name prefixed with "**CopyOf** *<name>*".

4 Click **Finish**. When copying a standard mapping, the new mapping will automatically be a custom mapping.

Saving a mapping

Created mappings must be saved on your workspace even when the mappings are not yet completed. In the Mapping Editor or in the Java editor, an asterisk after the mapping name or function indicates that the editor content has not yet been saved. To save your created mappings, select **File > Save** or use the shortcut key **Ctrl+S**.

To generate or publish a mapping, it must be saved to the Mapper Database because the Mapper Server reads mapping metadata from the Mapper Database. The mapping is automatically saved when you generate or publish the Mapper Database. You can also save the mapping to the Mapper Database without generating or publishing it.

The default way of saving a mapping to the Mapper Database is to save it to the server. Then, the mapping will be sent to the Mapper Server and saves the mapping to the Mapper Database.

When you save a mapping, the progress and result is logged in the **Mapping Console** view.

Note: If you replace a mapping that already exists in the Mapper Database, only the mapping is saved and not the schema files. Follow the standards in changing and saving a schema file:
- See Managing custom schemas on page 224.
- See Editing schemas on page 218.

If you change the XML schemas for a mapping, it is recommended that you give the mapping a new version number or rename the mapping. A new mapping is created in the Mapper Database.

When you save a standard mapping, it is automatically converted into a custom mapping. A custom mapping can have the same name as a standard mapping, but it is recommended to change the name for custom mappings, for example, by adding a suffix to the mapping name.

Saving to workspace

Use this procedure to save the changes made on a mapping. The mapping is saved in its current name and workspace location.

1   Edit the mapping in Mapper Editor.
2   Select **File > Save**.

The updated mapping is saved to the current workspace. A saved standard mapping is automatically converted into a custom mapping.

Saving to server

Use this procedure to save a mapping to the server. All saved activity and information are logged on the **Mapping Console** view.

1   Open the mapping in Mapper Editor.
2   Right-click a blank space in Mapper Editor. Select **Save to Server**.
3   Select a mapping server location.
4   Optionally, click **Test Connection** to check the connectivity to the server.

If an error occurs, a message is displayed. Check your server connectivity to correct the error. Click **OK** to proceed.

5 Click **Finish**.

Note: If the save to server action fails, the database is restored and the mapping is not saved. Take note of the error details and review your mapping or save logs.

6 If there is a validation error or warning, a confirmation dialog is displayed. Click Yes to continue saving the mapping. Click No to abort the operation. If there are validation errors or warnings, you can right-click the mapping and select Validate to view the validation errors and warnings.

The mapping is automatically validated before it is saved.

Note: To disable automatic validation on save, go to **Windows > Preferences > Mapper**. In the Validation group, clear **Validation On Save**.

7 If the mapping with the same name and version does not exist in the database, click OK to confirm to add the mapping to the database.

If the mapping with the same name and version does not exist in the database, follow this step:

• Click **OK** to confirm to add the mapping to the database.

If the mapping already exists in the database. Follow these steps:

Select to overwrite the existing mapping in the database or to update the version of the mapping. Update creates a new version of the mapping in the database.

a Click **Overwrite**.

b Click **Update Version** to update the version of the mapping before saving it to the database.

In the new window, specify a new version for the mapping. By default, 0.1 is added to the current version. For example, if the current version is 1.1, the proposed new version is 1.2. Change the new version, if necessary, and click **OK** to create a new version of the mapping in the database.

The mapping version is updated before it is saved to the database.

Note: Use Update Version only when you want to create a new version of the mapping. While developing a mapping, overwrite the existing mapping in the database because there is no need to keep intermediate versions of the mapping.

When creating a new mapping version, it is a good practice to first update the version of the mapping then overwrite the new mapping version, until the development is complete.

An additional dialog is displayed if a schema file has been added to or removed from the mapping since the mapping was saved in the Mapper Database the last time. The added and removed files will be listed. Click **Confirm** to update the Mapper Database with the added or removed files. The mapping does not need to be unpublished. You cannot save a standard mapping to the server.

Deleting mappings

The standard way of removing a mapping from a Mapper Database is to remove it from the server. A command will be sent to the Mapping Server, which in turn removes the mapping from the Mapper Database.

Note:

- Input and output XML schemas are saved in the database. If two or more mappings use the same XML schema, the XML schema is only stored once in the database.

  To remove an XML schema from the database, you must delete all mappings that are using the XML schema. When the last mapping that uses an XML schema is deleted from the database, the XML schema is also deleted from the database.

- **Delete from workspace** removes the complete mapping folder and not only the mapping file.

  This action can be reversed by using the Eclipse standard feature Restore from Local History on the Navigator tab.

- **Remove from server** has no reverse function.

  Before you remove a mapping, you must verify the name and version number of the mapping to remove. While you still have the mapping in your workspace, note that the mapping in the Mapper Database could differ from the mapping in your workspace.

  Before you can remove a mapping from the server, you must unpublish the mapping.

- You cannot remove a standard mapping from the server.

Removing mappings from server

Before you can remove mappings, you must verify that the correct mapping version is saved in the server (database) and exists in a workspace.

1 Open the mapping in Mapper Editor.
2 Right-click a blank space on Mapper Editor, then select **Remove from Server**.
3 Select a mapper server location where this mapping is stored.
4 Optionally, click **Test Connection** to check the connectivity to the server.
    a If an error occurs, a message is displayed. Review the error details and check your connectivity settings to correct the error.
    b Click **OK** to continue.
5 Click **Finish**.

  A remove failure message is displayed only if the mapping is not in the database. Review the error message and click **OK** to terminate remove action.

6 In the confirmation dialog, click **OK** to remove the mapping.

  You cannot remove a published mapping from the server and you must first unpublish the mapping. You can also remove the mapping from the server when you unpublish the mapping.

  You cannot remove a standard mapping from the server.

Deleting from workspace

Before you can delete mappings, you must verify that the correct mapping version exists in a workspace and your selected mapping is not open on the **Mapping Editor** tab.

1 On the **Mapping Explorer** tab, right-click a mapping to delete. Select **Delete**.
2 Click **OK**.

If your selected mapping is open in Mapping Editor, a message is displayed. Click **OK** to close the Mapping Editor and proceed.

If you delete a standard mapping from the workspace, the mapping still exists in the Mapper Database. It may have been updated to a newer version.

Validating document

The validation process checks the consistency of the mapping details. Validation checks the encoding, references, and checksums on a document.

1   In the **Mapping Explorer** view, expand the view of a mapping node to validate.
2   Right-click a Documents node and select **Validate Documents**.
    All XML schemas for the mapping are validated.

3   Optionally, right-click a single Document node to validate only that schema.
4   On the **Mapping Console** tab, review the detailed report.
    If an error occurs, click the link and review the error details.

Updating document checksums

A document can contain a checksum value as part of the document name. You must update checksums when document contents are changed. You can change the documents by using the External XML Editor.

1   In the **Mapping Explorer** view, expand the view of a mapping node to validate.
2   Right-click a Documents node and select **Update Document Checksums**.
3   Click **OK**.

Validating mapping

Validation checks the consistency of the mapping details and the documents. You can validate a mapping at any time. When the mapping is saved to the server, generated, or published, it is automatically validated.

1   Open a mapping in the Mapping Editor.
2   Right-click an empty area on the Mapping Editor and select **Validate**.
3   On the **Mapping Console** tab, review the detailed report.
    If an error occurs, click the link and review the error details.
    This table shows some of the checks performed when you run Validate on a mapping:

Checks performed	Check details
Loops	Loops are controlled by a function element. If the controlling function element is a **UBJ** function, then it must be the first child of the loop.
Data links	Data links are made between source and target elements that use the same or compatible data type.
Input/output root elements	The input/output tree root element is defined in the input/output schema.
Variables and constants	Validate action checks for unused or unconnected variables and constants.

Note: You can perform more checks when validating a mapping. To list all possible validation errors and warnings, run the command **Validations** in the console.

Renaming mappings

From Eclipse menu, the **Refactor > Rename** option does not apply to mappings. You can run **Rename** only through the Mapper context menu in the **Mapping Explorer** view. The new mapping name must be a valid Java class name.

1   In the **Mapping Explorer** view, right-click a mapping. Select **Rename**.

   Your selected mapping must not be open in the Mapper Editor.

2   Click **OK** to close the Mapping Editor for this mapping and proceed.

3   Specify a new name for this mapping. The new mapping name must be a valid Java class name.

4   Click **OK**.

   In the **Mapping Explorer** view, a new name is assigned to your selected mapping.

   Note: If the mapping is stored in the Mapper Database, the name of the mapping in the Mapper Database will not be changed. To change the name of the mapping in the Mapper Database, you must first remove the mapping with the old name from the server, then save the mapping with the new name to the server.

   When you rename a standard mapping, it will automatically be converted into a custom mapping.

Changing documents and root element

Use this procedure to update or change the input/output schema or document root elements of a mapping. The progress and result of this action are logged in the **Mapping Console** view. If the input/output schema contains a wildcard node, remove first the replacement element before changing documents.

See

Note: If you change documents for a customized BOD mapping, the input and output root element names must not be changed. The mapping properties **SchemaIn.Category** and **SchemaOut.Category** must not be changed either. If these mapping properties are changed, it will not be possible to use the customized BOD mapping instead of the standard BOD mapping.

1   In the **Mapper Explorer** view, right-click a mapping and select **Change Documents**.

Your selected mapping must not be open in the Mapping Editor. If open, click **OK** to close the editor and proceed. The fields are automatically populated with default values.

2   Click the **Browse** button for a specific field to change the default value. Specify this information:

Project

The current project name is the default value.

Copy or Overwrite

Optionally, select <u>Copy</u> or <u>Overwrite</u> to select how to apply your edits. <u>Copy</u> is the default selection.

Mapping Name

Specify a new name for the mapping. The mapping name must be a valid Java class name.

Note:
•   This is a potentially risky operation so your current mapping should be preserved. When Copy is selected, a new mapping folder is created by copying your current mapping and applying the changes on the mapping with new name.
•   When <u>Overwrite</u> is selected, the current mapping will be overwritten and hence not preserved.

Input schema

Specify a new input schema, or accept the default value.

Output schema

Specify a new output schema, or accept the default value.

3   Click **Next**.
4   Select the root elements for the input or output tree.
5   Click **Finish**.

The new mapping opens in a new tab in the Mapping Editor. Document elements linked to mapping elements may have disappeared if the new schemas have different content as compared to the old ones. The difference in schema content results in an inconsistent mapping with broken links. The new mapping will be a custom mapping.

Navigating around mappings

You can show mappings and navigate around the **Mapping Explorer** view.

1   In the **Mapping Explorer** view, expand the view of a project.

Mappings are indicated by a double-end diagonal arrow icon.

2   Double-click a mapping to show it in the Mapper Editor.

The mapping is displayed in a compressed state. Schemas are displayed as hierarchical tree structure.

Solid blue bullets indicate required schema elements, hollow blue bullets indicate optional elements.

Layered bullets indicate repeating elements. Smaller round bullets indicate element attributes.

Note: To optimize loading time, the Mapping Editor shows only the required schema elements and top-level schema elements connected to mapping elements.

Follow these guidelines:

- To see all connected elements, expand the elements view.

- To drill down and see non-connected elements, expand individual schema elements view.

3 Optionally, use the search feature to filter the list to view.

In the **Mapping Console** view, click the **Moogle** tab and search in the input or output document.

See

If an entry is used multiple times, select the desired occurrence in the presented alternatives.

4 Perform these navigation tasks:

Task	Steps
To display the complete input/output tree	From the **Mapping Editor** context menu, select any of these options: • Select **Expand Document** to display all the connected schema elements in the tree. • Select **Collapse Document** to display only top-level elements of the tree.
To locate mapping elements	**a** Open the **Mapping Console View**. Click the **Moogle** tab and search for elements in the input or output document. **b** Double-click a row in a list to show the corresponding element in the Mapping Editor.
To display the structure of a mapping	Select **Window > Show View > Outline**. The **Outline** tab shows a hierarchical tree view of the mapping.
To hide/reveal variables and constants	From the **Mapping Editor** context menu, select any of these options: • Select **Hide Variables and Constants** to hide connected variables and constants. • Select **Show Variables and Constants** to display connected variables and constants.

Mapping import and export

MEC Mapper allows you to import and export existing mappings to and from other mapping projects.

Exporting mappings

1   On the **Mappings Explorer** tab, right-click one or several mappings and select **Export**.
2   Specify the destination folder.
3   Click **Finish**.
4   Click **OK**.

Importing mappings

Add new mappings or import existing mappings to your mapping project.

Note: You must have a working Mapper server connectivity.

Import from server

1   On the **Mapping Explorer** tab, select a project to contain your imports.
2   Right-click and select **Import > From Server**.
3   Select a server location to use.
4   Optionally, click **Test Connection...** to verify the connectivity to the server.
    If an error occurs, a message is displayed. Review the error details and check your server connectivity. Click **OK** to proceed.
5   Click **Next**.
6   Select the mappings to import, you can use **CTRL-click** or **SHIFT-click**.
7   Click **Finish**.
    Note: If a mapping with the same name and version number already exists in the project, a message is displayed. Select whether to overwrite the existing mapping.

Import from archive file

1   On the **Mapping Explorer** tab, select a project to contain your imports.
2   Right-click and select **Import > From Archive File (.zap)**.
3   Click the browse button for the **Project** field. Select a project to contain imported classic mappings.
    This is automatically derived from the Project that you select in step 1.
4   Click the browse button for the **File** field. Select an archive .zap mapping file to import.
5   Click **Finish**.

The imported mapping will be a custom mapping.

Mapping deployment

A mapping is automatically saved in the Mapper Database associated with the Mapper Server when it is generated or published. Before a mapping can be used by the Mapper Server, it has to be deployed through the Publish action.

- When you generate a mapping, the mapping is built and published.

    Note: When you generate a published mapping, it replaces the current version of the mapping in the Mapper Database. The mapping source used in runtime is not replaced, though. The mapping source used in runtime is not replaced. When you publish a mapping, the Mapper database is updated to include the mapping. Successfully published mappings become available for use in the Mapper Server. Before you deploy mappings, you must define a connectivity to the Mapper Server. You cannot generate or publish a standard mapping

See

Generating mappings

Use this procedure to verify that a mapping can be translated into Java code that is accepted by the **Mapper Server**.

1 Double-click a mapping to open in the Mapper Editor.

2 Right-click the Mapper Editor and select **Generate**.

3 Select a server location where the mappings can be deployed.

4 Optionally, click **Test connection** to check the connectivity to the server machine. If there is an error, take note of the error details to correct later. Click **OK** to proceed.

5 Click **Finish**.

The mapping is automatically validated and saved to the server before it is generated.

If there is a validation error, a confirmation dialog is displayed. Click **Yes** to continue saving the mapping. Click **No** to abort the operation. For example, if the mapping only has validation warnings and document checksum warnings, no confirmation dialog is displayed. You cannot disable this automatic validation. If there are validation errors or warnings, you can right-click in the mapping and select Validate to view the validation errors and warnings.

Then, the mapping is saved to the server. The same dialog for confirmation of adding or overwriting the mapping in the Mapper Database is used as when saving the mapping to the server.

See

The mapping is generated on the server and you cannot generate a standard mapping.

Publishing mappings

Mappings are published by using a **Mapper Generator** that is associated with the **Mapper Server**.

When you publish, the Mapper Database is updated to make updates. The mappings are loaded into the Mapper Sever and are ready to run.

Published custom mappings are used in the Mapper Server using the **Server > Custom Mappings page**.

1   Double-click a mapping to open in the Mapper Editor.
2   Right-click the Mapper Editor view and select **Publish**.
3   Select a server location.
4   Optionally, click **Test connection** to check the connectivity to the server machine. If there is an error, take note of the error details to correct later. Click **OK** to proceed.
5   Click **Finish**.

    The mapping is automatically validated and saved to the server before it is published.

Unpublishing mappings

Unpublish action updates the Mapper Database to make the mapping unavailable for the Mapper Server. Published mappings are unloaded from the Mapper Server if these mappings are not used or run by any Partner agreements.

1   Double-click a mapping to open in the Mapping Editor.
2   Right-click the Mapping Editor and select **Unpublish**.
3   Select a server location.
4   Optionally, click **Test connection** to check the connectivity to the server machine. If there is an error, take note of the error details to connect later. Click **OK** to proceed.
5   Click **Finish**.
6   On the **Confirm Unpublished wizard** page, select either `Unpublish`, `Unpublish and Remove`, or `Cancel`.

    The mapping is automatically validated if it is not used by any Partner agreement. If there is none, unpublish activity will push through.

    Unpublish causes the mapping to be unloaded from the server. **Unpublish** and **Remove** causes the mapping to be unloaded from the server and additional confirmation will be asked before permanently removing the mapping in the Mapper Database. **Cancel** terminates the activity.

Adding control properties

The control properties tab allows you add, edit and delete control properties' names and values for related maps and corresponding agreements. Use these steps if you are creating a custom map through import of a standard map or a new map. Control properties are needed for the runtime parameters that the map will use during execution.

1   To add a control property, go to **Mapping Console > Control Properties tab**.

2   Right-click and add new properties.

3   Edit the value of Name and Value to suit your requirements.

4   Press Ctrl+S to save or click the **Save** button on the tool bar.

To delete control properties, right-click and click delete.

hasCustomDetection and *isAgreementInactive* are not allowed to be included in the generated agreement's properties.*isAgreementInactive* will always default to true. This makes the agreement inactive.

The values for Control Properties can be modified through the SAT page in **Server > Agreement Control Properties**.

Configuring check order

Check order is an error-checking process used to determine if a Sync BOD received by the IEC Server is updated. Check order starts by checking the variationID for inbound Sync BODs. The variationID is an attribute containing a version number for the BOD. This version number refers to the identification (ID) of the BOD, for example, the item number for a SyncItemMaster BOD. If a Sync BOD is received, but another Sync BOD for the same noun and identification, having a higher (or the same) variationID, has already been received, the new Sync BOD will be disregarded by the IEC server. This is because the data in the BOD is outdated. There is a check order process in the agreement to handle this. You must set up the check order process for inbound Sync BOD mappings to generate custom agreements.

You can set up how to find the BOD ID by using the check order configuration window. This is the primary key for the BOD. You must also set up how to find the variationID (VID) in the BOD. XPaths are used for these settings.

When importing a standard mapping for an inbound Sync BOD from the server, the check order settings are automatically copied from the standard agreement. You must use these settings as basis for custom BODs, as it can be difficult to make the correct settings. The IEC server stores variationID values for all primary keys. The check order settings should not be changed, once in production, as this can create inconsistent data.

Note: Check order configuration only applies to inbound Sync agreements.

1   To access check order, open an inbound Sync BOD mapping and right-click the mapping work area.

2   Click **Configure Check Order**.

3   Select the box **Include Noun for Primary Key** to include the BOD noun in the primary key stored in the IEC server. This is to separate BODs having different nouns but the same ID.

4   Specify the default namespace for the BOD in the field **Default Namespace**.

The standard namespace for Infor BODs is "http://schema.infor.com/InforOAGIS/2".

5   Specify a default namespace prefix in the field **Default Namespace Prefix**. The check order process uses this to indicate the default namespace in XPaths. Standard value is "dns".

6   Click **Add** to add a new XPath for the primary key.

At least two XPaths are used to pinpoint the primary key: the accounting entity and the document ID. The window shows **PrimaryKey Retrieval Settings**. Specify the XPath for the element or

attribute in the field **PrimaryKey Xpath**. An ID element is repeating to give the possibility to use several alternative IDs. This is indicated by the attribute schemeName. As the original ID should be used for the primary key, the instance of the element without a schemeName attribute should be used. If this functionality is required, enter the name of the attribute that should not exist for the element in the field **PrimaryKey No Attribute Existing**. Standard value is "schemeName".

7   Click **Edit** to edit the selected primary key XPath. You can also double-click the XPath to edit it.

8   Click **Delete** to delete the selected primary key XPath.

9   Specify the XPath for the variationID attribute or element in the field **VID Xpath**. This field must have a value, otherwise the variationID cannot be found in the BOD.

10 An ID element is repeating to give the possibility to use several alternative IDs. This is normally indicated by the attribute schemeName. As the variationID for the original ID should be used, the instance of the element without a schemeName attribute should be used. If this functionality is required, enter the name of the attribute that should not exist for the VID element in the field **VID No Attribute Existing**. Standard value is "schemeName".

11 Click **OK**.

12 Right-click the mapping work area and click **Validate** to validate the changes made.

Check order settings currently do not have an edit function in SAT.

You must provide the control properties and check order properties when publishing a custom agreement for the first time. Control properties are applicable to all types of BODs, regardless of direction (inbound/outbound) or verb (Acknowledge, Load, Process, Show, Sync, Update). Check order is only applicable to inbound Sync BODs. The agreement will be generated successfully after following these steps. The custom agreement will be updated accordingly when the mapping is republished if the control properties or check order settings are updated for a published mapping.

Check order settings

You must input certain settings when configuring check order for standard BOD mappings. An example is shown below.

Note: The XPath syntax used by IEC is different from the standard XPath syntax. The standard XPath for an attribute is "element@attribute", for example "ID@variationID". The corresponding syntax in IEC is "element[@attribute]", for example, "ID[@variationID]". In standard XPath syntax this XPath would select all ID elements that have an attribute named "variationID". This is not how IEC works.

Example settings for the standard BOD mapping M3BOD_BillOfMaterials_Sync_In_2_13_2:

*   Include Noun for primary key: Selected
*   Default Namespace: http://schema.infor.com/InforOAGIS/2
*   Default Namespace Prefix: dns
*   Primary Key Setting #1:

    Primary Key Xpath: /dns:SyncBillOfMaterials/dns:DataArea/dns:BillOfMaterials/

    dns:BillOfMaterialsHeader/dns:DocumentID/dns:ID

    Primary Key No Attribute Existing: schemeName
*   Primary Key Setting #2:

Primary Key Xpath:
/dns:SyncBillOfMaterials/dns:DataArea/dns:BillOfMaterials/dns:BillOfMaterialsHeader/

dns:DocumentID/dns:ID[@location]

Primary Key No Attribute Existing: schemeName

- Primary Key Setting #3:

    Primary Key Xpath:
    /dns:SyncBillOfMaterials/dns:DataArea/dns:BillOfMaterials/dns:BillOfMaterialsHeader/

    dns:DocumentID/dns:ID[@accountingEntity]

    Primary Key No Attribute Existing: schemeName

- VID

    Xpath: /dns:SyncBillOfMaterials/dns:DataArea/dns:BillOfMaterials/

    dns:BillOfMaterialsHeader/dns:DocumentID/dns:ID[@variationID]

    VID No Attribute Existing: schemeName

In the BOD example shown below, the primary key is "BillOfMaterials" (noun) + "1234" (ID) + "105" (location) + "770_AAA" (accountingEntity). The Variation ID (VID) is "20". Note that there is no schemeName attribute for the ID element.

```
<BillOfMaterials>
 <BillOfMaterialsHeader>
 <DocumentID>
 <ID location="105" accountingEntity="770_AAA" variationID="20">1234</ID>
```

Working with BOD mappings

IEC Mapper allows you to create custom maps. Follow these topics and use cases if you are creating a custom map through a new map or by importing existing maps. See BOD Processor administration tasks on page 113 for more information on BODs.

Creating a new custom BOD

You can create a new custom BOD, that is not a standard BOD, using this feature. To create custom BODs, you must follow these general steps which require you to perform tasks in Event Analytics Rules (EAR).

1   Create a new custom session with rules in EAR, for outbound BODs. See Activating a rule on page 137.

2   Add a new custom BOD in BP, for outbound BODs. See Enabling BODs on page 113.

3   Create a new custom BOD mapping in IEC. Publish the mapping and generate a custom agreement. Activate the custom agreement.

    The prerequisite of this step is that the mapper is already downloaded and configured to connect to the IEC server.

You need to download and install the MEC mapper to create custom BODs. The Downloading Mapper on page 166 gives information on MEC Mapper installation.

Once the mapper setup is complete, you must create a new map by following the steps in Creating a mapping on page 194.

There are restrictions when creating a new map as this needs to be followed when creating a custom BOD map.

a   Provide naming rules. See BOD naming rules on page 210.

b   Provide details on Adding control properties on page 206

c   Provide details on Configuring check order on page 207 settings

Once the map is ready, publish the map to the IEC server. Publishing mappings on page 206 explains how you can publish the mappings. If you are working on a valid BOD map based on the rules and guidelines provided previously, the IEC server will also generate a custom BOD partner agreement. This will be stated in the confirmation message.

To activate the custom BOD, see Activating a new partner agreement on page 210. This action will automatically trigger a reload of the needed IEC communication channels and the IEC server can start accepting messages for this agreement and map.

4   New messages will start using this custom BOD agreement. Check the result in ION for outbound BODs or in M3 for inbound BODs.

Completing these steps will result in a new custom BOD that works just as a standard BOD of the same type.

Activating a new partner agreement

The newly created partner agreement is inactive by default. To activate the new partner agreement:

1   Go to **server administration > agreement status** page.

2   Search for the custom BOD agreement.

3   Activate the BOD agreement.

Deactivating a custom BOD agreement

You can deactivate an existing BOD partner agreement by deactivating the specified partner agreement.

This will deactivate the specified agreement only. If there was a corresponding standard agreement, then this action will leave both agreements inactive. This does not happen all the time as sometimes, there will be no corresponding standard agreement and in that case, there will be only one.

BOD naming rules

You need to follow certain rules when naming BODs.

The naming rules for main BOD maps are:

<Any Prefix>_<BOD Noun>_<BOD Verb>_<In or Out>_<Any Suffix>

The naming rules for error BOD maps are:

<Any Prefix>_<BOD Noun>_<BOD Verb>_Error_<In or Out>_<Any Suffix>

These are some examples of BOD types:

Description	Sample	Classification	Result
Conforms with [{Any_String}_[Valid Verb]_[Valid Direction]]	M3BOD_Shipping_Ac-knowledge_In	M3 BOD	Supported if the BOD type has the verb "Process" and direction "In"
Starts with any prefix, other than M3BOD. Conforms with [{Any_String}_{Any_String}_[Valid Verb]_[Valid Direction]]	Test_NonAck_Acknowl-edge_Out	NON M3 BOD, but valid BOD	Supported if the BOD type has the verb "Process" and direction "In"
Does not conform with [{Any_String}_{Any_String}_[Valid Verb]_[Valid Direction]] pattern	MyMapping My_Mapping	NON-BOD	Not supported

Asynchronous and Synchronous Process in BODs

An inbound Process BOD can be either synchronous or asynchronous. For a synchronous inbound Process BOD, an Acknowledge BOD will be returned by the Process agreement. The mapping then reads the input Process BOD, processes it and creates an output Acknowledge BOD. For an asynchronous inbound Process BOD, the Acknowledge BOD will be returned by a separate outbound Acknowledge agreement, having the same noun. The mapping in the Process agreement then reads the input Process BOD, processes it but does not create an output Acknowledge BOD. In both cases, if something goes wrong in the Process BOD mapping, an error Acknowledge BOD is returned by the error handling in the Process agreement. An error Acknowledge mapping is always needed for inbound Process BODs. It can either be an existing standard mapping or a custom mapping.

When you publish an inbound Process mapping for the first time, you will be asked which agreement type to generate, either a synchronous agreement or an asynchronous agreement. The agreement type cannot be changed for existing agreements.

Selecting synchronous or asynchronous agreement type determines what type of agreement is generated:

Agreement type	Definition	Processes tab contents	Error handling tab contents
Synchronous	Generates an agreement where the Acknowledge BOD is returned by the Process IEC agreement. This is the most common agreement type.	• XML Transform • Apply Envelope • Send	• Create Confirm-BOD • Send • XML Transform • Apply Envelope • Send
Asynchronous	Generates an agreement where an Acknowledge BOD is not returned, except for error Acknowledge BODs. M3 needs to process the request separately before a non-error Acknowledge BOD can be created. A separate outbound Acknowledge agreement will then be triggered by M3. This is used for the standard inbound ProcessItemMaster BOD.	• XML Transform	• Create Confirm-BOD • Send • XML Transform • Apply Envelope • Send

Getting a synchronous process agreement

If you have generated an asynchronous agreement by mistake, no Acknowledge BODs will be sent to ION, except for error Acknowledge BODs. To get a synchronous Process agreement, you need to perform these steps:

1 From the **Enterprise Collaborator Administration** page, go to **Server > Agreement Status** page.

2 Deactivate the custom Process agreement if it is active.

3 Delete the custom Process agreement.

4 Republish the Process mapping in the mapper.

Select the agreement type `Synchronous`.

Deleting a custom agreement

You can delete existing custom agreements using the delete custom agreement function. Once an agreement is deleted, it is permanently removed and you cannot undo the deletion. You can only delete inactive agreements, so you must deactivate the selected agreement before deletion.

Note: You can only delete inactive custom agreements. The other types of agreements cannot be deleted.

1 From the IEC Server administration page, go to **Server > Agreement Status**.

2 Click **Delete** on the listed agreement in the **Partner: Custom** table.

3 Click **Proceed**.

Update a custom BOD

You can update an already existing custom BOD. To update custom BODs, you must follow these general steps which require you to perform tasks outside IEC in Event Analytics Rules (EAR).

1 Update the custom rules in EAR, for outbound BODs. See Reactivating a rule on page 137.

2 Update the custom BOD mapping in IEC and keep the same mapping version.

3 Republish the mapping.

Import the custom BOD map into the server. See Importing mappings on page 204 and Mapping management on page 228.

It is very much recommended to always import a fresh copy of a map from the server before doing any modifications. This makes sure that the properties are in Sync with the server copy. If you are working on an old copy of a map in your workspace, it will most likely be outdated compared to the properties saved in the server.

Once imported, the user can proceed with actual modifications to the map. For this use case, there are no changes to the map version. For changes to the map version, refer to the other use cases. The user also needs to make sure and confirm that the control properties defined are correct. See Adding control properties on page 206. If needed, the user also needs to confirm the check order settings. See Configuring check order on page 207.

Once the map is ready, publish the map to the IEC server. See Publishing mappings on page 206.

There are 2 sub cases that can happen when you publish a custom BOD map:

- In the case where the custom BOD map is currently assigned to the custom BOD agreement, the updated published map will be updated to its currently assigned agreement in the server. The control properties will also be updated in the custom agreement. If check order settings are required by the agreement, these will be taken and updated as well in the server.

- In the case where the custom BOD map is not assigned to the agreement, then the map will just be published without any modifications to the agreement.

This information will be provided in a future update.

If the custom agreement this map is assigned to is active, then the next message that will be processed will use this new and updated map.

If the custom agreement this map is assigned to is inactive, then you need to activate the custom BOD agreement for this map to be in use. See instructions on activating/deactivating agreements in the server administration pages.

You might need to update control properties before proceeding. Identify which BOD type needs check order settings and check order settings are updated before publishing. See Adding control properties on page 206.

4   New messages will start using this custom BOD agreement. Check the result in ION for outbound BODs or in M3 for inbound BODs.

Completing these steps results in an updated custom BOD that works just as a standard BOD of the same type.

Creating a new version of a custom BOD

You can create a new version of an already existing custom BOD. To create a new version of a custom BODs, you must follow these general steps which require you to perform tasks outside IEC in Event Analytics Rules (EAR).

1   Create new versions of the custom rules in EAR by using Duplicate, for outbound BODs. See Activating a rule on page 137 and Deactivating a rule on page 137.

2   Create a new version of the custom BOD mapping in IEC.

You must publish the new mapping and regenerate the custom agreement.

See Update a custom BOD on page 213. You need to a different set of prerequisites for creating a new version of a custom BOD.

The same steps apply exactly the same except for this use case where the user will update the version of the map.

The version of the map is now critical for the updates. Only the latest version will be automatically assigned to the agreement. If the version is updated to a lower version than the current, the map will just be published into the server but will not be assigned to the agreement. See versioning rules for BODs for more information.

3   New messages will start using this custom BOD agreement. Check the result in ION for outbound BODs or in M3 for inbound BODs.

Completing these steps results in a new version of the custom BOD that works just as a standard BOD of the same type.

Stop using a custom BOD

You can stop using a custom BOD. The steps are the same as replacing a custom BOD with a new standard BOD, but no corresponding standard BOD exists.

1   Outbound BODs only: Stop the custom session in EAR. See Stopping a session on page 135.

2   Outbound BODs only: Disable the custom BOD in BP See Enabling BODs on page 113.

3   Deactivate the custom agreement in IEC.

See instructions in server administration pages to activate or deactivate agreements.

Completing these steps will result in the custom BOD not being used, but it is not deleted.

Activating duplicate agreements

A message will inform you if you are attempting to activate an agreement that is similar to an activated agreement. When your selected agreement has an activated duplicate agreement, a message appears asking you if you want to activate the selected agreement. The selected agreement for activation shows in the message along with the activated duplicate agreement. Click **Proceed** if you want to activate the selected agreement and deactivate the previous duplicate agreement.

Delete an unused version of a custom BOD

You can delete an unused version of a custom BOD.

1   Delete used inactive rules in the custom session in EAR, for outbound BODs. See Deactivating a rule on page 137.

2   Unpublish and remove the unused version of the custom BOD mapping in IEC.

    See Publishing mappings on page 206 and Unpublishing mappings on page 206.

3   Verify that unused rules are deleted in EAR for outbound BODs, and that the unused mapping version for the custom BOD is deleted in IEC server.

    Completing these steps will result in the deletion of the unused version of the custom BOD.

Delete an unsupported custom BOD

You can completely delete a custom BOD.

1   Delete the custom session in EAR, for outbound BODs. See Deleting a session on page 135.

2   Delete the custom BOD in BP, for outbound BODs. See Adding custom BODs on page 114.

3   Delete the custom agreement in IEC. Unpublish and remove all mapping versions of the custom BOD.

    Deleting custom agreements is not yet supported. You can unpublish, remove mappings and, deactivate the custom agreements as needed.

    You can

4   Verify that the custom session is deleted in EAR, that the custom BOD is deleted in BP for outbound BODs and that the custom agreement and all mapping versions for the custom BOD are deleted in the IEC server.

    Completing these steps will result in the deletion of all versions of the custom BOD.

Custom mapping error messages

When working with BOD mappings, you can encounter several possible error messages:

Error message	Description
An agreement cannot be generated when a multiple acknowledge error map having the same noun exists.	This error happens when multiple versions of the same noun, (e.g. Itemmaster) exist. To solve this, you must ensure that there are no other similar nouns.
An agreement cannot be generated yet. No available Acknowledge Error Mapping for Inbound Process Mapping with noun, exists.	This error happens when you try to publish a Process_In mapping without a matching Acknowledge_Error mapping. A similar prefix is not required for a mapping to be published, it only requires the same noun. To solve this, you must have the same noun. For example: Noun=ItemMaster
An agreement cannot be generated yet, please create a custom inbound process mapping for the noun.	This error happens when you publish the error map first. To solve this, you must have a custom agreement to go with it, such as a Process_In map.

Multiple mappings deployment

A mapping is saved in a Mapper Database associated with the Mapper Server before it can be used by the Mapper Server. Saved mappings are then deployed through the Publish action. You can simultaneously publish multiple mappings or simultaneously generate multiple mappings.

Saving a mapping to database is part of the deployment process and is made automatically.

Generating multiple mappings

You can simultaneously generate multiple mappings from the **Mapping Explorer** view. The generate activity and information are logged in the **Mapping Console** view.

1   Select one or several custom mappings from one or several mapping projects in the **Mapping Explorer** view.

2   Right-click one of your selected mappings and select **Generate...**

3   Select a server location.

4   Optionally, click **Test Connection** to check the connectivity to the server machine.

If there is an error, review the error message displayed and click **OK** to proceed.

5   Click **Finish**.

If a mapping already exists in the server's database, a confirmation dialog will be displayed. Click **Overwrite** to overwrite the current mapping in the database with the mapping in the mapper. Click **Cancel** to skip generation of that mapping.

In the console, the view shows the progress of the ongoing generation. When the processing is finished, a summary is displayed. Standard mappings cannot be generated.

Publishing multiple mappings

You can simultaneously publish multiple mappings from the **Mapping Explorer** view. The publish activity and information are logged in the **Mapping Console** view.

1   Select one or several custom mappings from one or several mapping projects in the **Mapping Explorer** view.
2   Right-click one of your selected mappings and select **Publish**.
3   Select a server location.
4   Optionally, click **Test connection** to check the connectivity to the server machine.

    If there is an error, review the error message displayed and click **OK** to proceed.

5   Click **Finish**.

    A confirmation dialog is displayed if a mapping already exists in the server's database. Click **Overwrite** to overwrite the current mapping in the database with the mapping in the mapper. Click **Cancel** to skip publishing of that mapping. Standard mappings cannot be published.

    In the console, the view shows the progress of the ongoing generation. When the processing is finished, a summary is displayed.

Unpublishing multiple mappings

You can simultaneously unpublish multiple mappings from the **Mapping Explorer** view. The unpublish activity and information are logged in the **Mapping Console** view.

1   Select one or several mappings from one or several mapping projects in the **Mapping Explorer** view.
2   Right-click one of your selected mappings and select **Unpublish**.
3   Select a server location.
4   Optionally, click **Test connection** to check the connectivity to the server machine. If there is an error, take note of the error details to connect later. Click **OK** to proceed.
5   Click **Finish**.

    On the **Confirm Unpublish wizard** page, select either **Unpublish**, **Unpublish and Remove**, or **Cancel**.

    The selected mappings will automatically be unpublished (and removed) from the server if they are not used by any Partner Agreement.

    Click **Unpublish**, the mapping will be unloaded in the server. Press **Unpublish and Remove** the mapping will be unloaded in the server and additional confirmation will be asked before permanently removing the mapping in the Mapper Database.

    Click **Cancel**, the activity is aborted.

In the console, the view shows the progress of the ongoing activity. When the processing is finished a summary is displayed.

Schemas

Schemas can be complex and may import or include other schemas. If you use complex schemas in your mapping, the schemas must meet the listed criteria for the Mapper to correctly import all the relevant schemas into the Mapping project. If the schemas do not meet the criteria, you must edit the schemas before you use them.

Refer to these criteria in using complex schemas:

- All referenced schemas must be present on disk.

  References to non-local schemas are not valid, for example, `http://som/url/somewhere/as chema.xsd`.

- All references must be relative to the folder with the root schema.

  For example:

  - Root schema `A.xsd`; Import schema `B.xsd`, and `C.xsd`

  - Location sub-folders `FB` and `FC` to folder A.

  The reference paths must be `/FB/B.xsd` and `/FC/C.xsd` respectively.

  Relative paths against the root schema location are also valid, for example, if schemas A, B, and C are located in this structure, then B may import C as `"../SUBFOLDER_C/C.xsd"` :

  `MY_FOLDER/A.xsd`

  `MY_FOLDER/SUB_FOLDER_B/B.xsd`

  `MY_FOLDER/SUB_FOLDER_C/C.xsd`

Editing schemas

The Infor Business Document Mapper does not provide functionality for editing of the input or output schemas. You can edit the schemas using an external XML editor. Before you can use an external XML editor, you must specify the path to an external XML editor in **Window > Preferences > Mapper XML Editor**.

Note: Editing the map file in an external XML Editor can destroy the mapping. You might not be able to open the edited mapping or to save the edited mapping to the server.

To properly edit the schemas, follow these guidelines:

1 Edit the schemas in an external XML editor.
2 Update the schema checksums.

   Note: Ensure that the mapping editor is closed before this operation.

   a In Mapping Explorer view, right-click a Document node and select **Update Document Checksums**.

    b   Click **OK** to proceed with updating the document checksums.

        Note: If an included or imported schema is edited, both the checksum for the edited schema and the checksum for the schema including or importing the edited schema will be changed.

3   Open the mapping to see your changes in the mapping editor.

    When opening the mapping there may be broken links if you have changed existing element or attribute names in the schema, or if you have changed a namespace.

    Note: The next time you save the mapping to the server, the old schemas will be listed as removed, and the new schemas will be listed as added. This is because the schema names have been changed due to the updated checksums. If you save, the changed schemas will be changed in the Mapper Database.

Changing SchemaIn and SchemaOut version

You can update a mapping's input and output schema versions under the mapping properties in IEC Mapper. This function allows you to specify the version of the schema in the mapping file for saving.

To change the versions of Mapping, SchemaIn and SchemaOut, you must:

Note: If you need to adjust the SchemaIn and SchemaOut version, you must update the mapping version.

1   Open the mapping file on the mapping editor.

2   Click on the Properties tab.

3   Specify the versions on these fields:

    Mapping version
    Shows the current version of the mapping

    SchemaIn
    Shows the current version of the input schema

    SchemaOut
    Shows the current version of the output schema

4   Save the mapping.

5   Right-click on the mapping editor.

6   Click Publish.

7   Specify the server location and click Finish.

    If the mapping to be modified with an updated schema version is already existing in the server, you must increment the mapping version to save the updated schema versions. You will not be able to save the updated schema versions if you only overwrite the existing mapping with the same mapping version.

Setting the schema location in Mapper

You can enter the schema location for an output document in a user-defined field. The IEC Mapper SchemaOut.Location is used to define the schema location if a mapping has an output document. The user-defined field contains null value by default.

To define the schema location, you must:

1 Click on any empty space of the mapping editor.

2 Go to **Mapping Console > Properties** tab.

3 Click the SchemaOut.Location value field.

4 Input the schema location.

Wildcard elements

Wildcard elements explain how to use the XML schema element `any`. The XML schema construct, `any`, is used, for example, in Infor BOD UserArea extensions. Mapper follows the W3C standards. The specification for ION BOD UserArea extensions is a subset of the W3C standards.

The XML schema element `any` enables the XML document to be extended with elements that are not specified by the schema. This makes the data structure for the `xs:any` element to be unknown, even if data mapping requires known data structures to map data from and to. The solution is to define the elements that will replace the wildcard construct in instance documents. When the elements are defined for the mapping, these are shown in the document trees and can be used similar to any other element that is declared in the main schemas.

See

Wildcard elements limitations

The schema element `xs:any` has two attributes.

Namespace

This attribute specifies the namespaces containing the elements that are used as replacement elements. Mapper assumes that elements from any namespace are allowed. This is the default behavior for namespaces.

processContents

This attribute specifies how the XML processor, which receives the output document, must use validation against the replacement elements. Mapper assumes that the XML processor must obtain schemas for the used namespaces and validate the replacement elements. This is the default behavior for processContents.

This XML code shows how to write the element `xs:any` :

```
<xs:any namespace="##any" processContents="strict"/>
```

For the element `xs:any`, the attributes **minOccurs** and **maxOccurs** are reflected when showing wildcard elements in the input and output document trees.

Infor BOD UserArea extensions

UserArea extensions provide features to manage elements and custom schemas.

UserArea

UserArea is a construct provided by OAGIS that allows data to be associated with a noun. This data is not previously defined for a noun. UserArea is built with a type `xs:any`, which means that any defined element can be used inside `xs:any`. For more information on UserArea elements, see the Infor ION technical documentation.

The UserArea element has a wildcard element as its only child. This complex type is used by all UserArea elements:

```
<xs:complexType name="UserAreaType" block="restriction">
 <xs:sequence>
  <xs:any namespace="##any" processContents="strict" minOccurs="0" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

A UserArea element may have one or more replacement elements as child elements.

Note: Use replacement elements in a custom BOD mapping to extend a standard BOD mapping.

For Infor BODs, you can use these types of replacement elements:

User Defined Fields

User Defined Fields are always published using the Property element which belongs to the default namespace.

BOD instance example:

```
<UserArea>
  <Property>
    <NameValue name="ln.Priority" type="NumericType">999</NameValue>
  </Property>
  <Property>
    <NameValue name="ln.Owner" type="StringType">SFC</NameValue>
```

```
    </Property>
</UserArea>
```

Custom Elements

Custom elements may not be added by standard development teams. Custom elements are new elements that are defined in a custom namespace.

BOD instance example:

```
<UserArea>
  <cust:AdditionalInformation>
    <cust:ReceivingWarehouse>WHAMS1</cust:ReceivingWarehouse>
    <cust:InterfaceUpdate>No</cust:InterfaceUpdate>
  </cust:AdditionalInformation>
</UserArea>
```

ION Connect restrictions

Follow these restrictions:

- Only one custom namespace is allowed for every customer. Several custom schemas can be used for different replacement elements for different UserArea elements.

- Each custom schema can only have one top-level element declaration which is a replacement element. That element can be a complex element, but there must not be any includes, or imports, in the custom schema.

- Only one custom schema can be used for every UserArea element. One custom schema can be used by several UserArea elements.

Icons

The element icons for wildcard elements and replacement elements are marked with an asterisk.

A wildcard element has no name, only the element icon with an asterisk is shown. Although a wildcard element is shown to have its own level in the document tree, this level does not represent a level in the data structure. It is used as a placeholder for the wildcard element's replacement elements, for example, the relative XPath for a Property replacement element is `UserArea/Property`, not `User Area/ /Property`.

Custom namespace

A custom namespace is derived from a custom schema's target namespace. Several custom schemas using the same target namespace can be used per custom namespace. The custom namespace must not be the same as the main schema's target namespace, any other namespace defined in the main schema, or another custom namespace.

Note: You can use custom namespaces, custom schemas, and replacement elements in custom BOD mappings to extend standard BOD mappings.

Managing a custom namespace

Custom namespaces are global per document. You can use any wildcard element to manage custom namespaces.

Adding a custom namespace

1 Right-click a wildcard element and select **Manage Custom Namespaces**.
2 In custom namespace section, click **Add**.
3 In custom schema section, click **Add**.
4 Select a custom schema (xsd file) to import.

   The target namespace must not already exist in the main schema or as another custom namespace.

   The namespace URI is displayed.

5 Specify a unique custom prefix and click **OK**.

   The selected custom schema is added to the mapping.

6 Click **OK** to close the Manage Custom Namespaces dialog box.

   Note: If the mapping is already saved in the database, you must confirm adding of the custom schema to save the mapping to the database.

   The custom schemas and the mapping are saved to the database. You can increase the mapping's version number and publish a new version of the mapping, instead.

Removing a custom namespace

Before removing the custom namespace, all associated replacement elements for the custom namespace must be removed.

1 Right-click a wildcard element and select **Manage Custom Namespaces**.
2 Select the custom namespace and click **Remove**.
3 Click **OK** to remove the custom namespace.
4 Click **OK** to close the Manage Custom Namespaces dialog.

   Note: If mapping is published, you must first unpublish the mapping to save the mapping to the database because the schemas are changed.

   When the mapping is saved, the custom schemas for the deleted custom namespace are removed from the mapping's file system. Although, a custom schema is not removed if it is also used for the other document (input or output), or by another mapping.

Custom schemas

A custom namespace must have at least one custom schema. The target namespace in the custom schema defines the custom namespace. Several custom schemas with the same target namespace can be used for a custom namespace. Top-level element declarations from these custom schemas for a custom namespace can then be used as replacement elements.

Custom schemas are handled in the same way as the main schemas are handled in the file system. Custom schemas are copied from their original locations and stored together with the mapping metadata files.

When you save a mapping with custom schemas to the mapper database the custom schemas are also saved.

Similar to main schemas, the custom schemas follow the lifecycle of the mapping. When you import and export a mapping, the custom schemas are also imported and exported. You can add custom schemas to custom BOD mappings.

Managing custom schemas

You can add or remove a custom schema to an existing custom namespace.

The first custom schema is added when the custom namespace is created.

Before you can remove a custom schema, you must ensure that a custom namespace has more than one custom schema. A custom namespace must always have at least one custom schema.

Adding a custom schema

1   Right-click a wildcard element and select **Manage Custom Namespaces**.
2   Select the custom namespace and click **Edit**.
    Note: After the custom namespace is created, you cannot change the prefix.
3   Click **Add**.
4   Select an .xsd file custom schema to import.
    Follow these guidelines:
    •   The target namespace for the selected custom schema must be the same as the existing custom namespace URL.
    •   The selected custom schema must not contain a top-level element that is already declared in another custom schema for the custom namespace.

    The added custom schema or schemas are displayed. Repeat steps 3 and 4, if necessary.
5   Click **OK** to confirm.
6   Click **OK** to close the **Manage Custom Namespaces** dialog box.

Removing a custom schema

If there is only one custom schema for a custom namespace, you cannot remove the custom schema. A custom namespace must always have at least one custom schema.

1   Right-click on a wildcard element and select **Manage Custom Namespaces**.

2   Select the custom namespace from where a custom scheme will be removed and click **Edit**.

3   Select a custom schema to remove and click **Remove**.

    Repeat this step to remove more custom schemas.

4   Click **OK** to remove the custom schemas.

5   Click **OK** to close the **Manage Custom Namespaces** dialog box.

Managing replacement elements

Replacement elements are handled for every wildcard element. Before you can manage replacement elements, you must find the wildcard element for the replacement element.

Adding a replacement element

1   Right-click the wildcard element and select **Add Replacement Element**.

    A list of available replacement elements is displayed. The list shows the namespace URI, prefix, and name for each replacement element. Replacement elements that are already added for the current wildcard element are omitted. You can use the search feature to find the correct replacement element.

    The search string can be part of the desired value and is not case sensitive. You can use an asterisk (*) in the search string as a wildcard character, replacing zero, and one or many unknown characters. Use the drop-down arrow box for the search field to select the columns to search in.

2   Select the replacement element and click **OK**.

    The replacement element is added as the last entry under the wildcard element.

3   Right-click the replacement element and select **Move Up** to move the replacement element.

    When you move replacement elements, its grouping properties are automatically set, but it can be updated.

Updating replacement elements

1   Expand the wildcard element to show its replacement elements.

2   Right-click the replacement element and select a direction to move the element.

3   Change the grouping properties of a replacement element.

    a   Select the replacement element and click the **Properties** tab.

    b   Select the values for the **Mandatory** and **Repetitive** properties to change their grouping properties.

The replacement element's icon is updated in real time.

Note: Child elements to a replacement element cannot be moved or updated because they are defined in the schema for the replacement element.

Removing replacement elements

1   Expand the view of wildcard element to show all replacement elements.

2   Right-click the replacement element and select **Remove**.

3   Click **OK** to remove the replacement element.

Adding property replacement elements

Infor BODs use the replacement element Property, declared in the default namespace for User Defined Fields. The default namespace is in `http://schema.infor.com/InforOAGIS/2`.

1   Right-click the wildcard element.

2   Select **Insert Property Element**.

For all wildcard elements, this context menu option is displayed only if the document's schema category is set to **BOD**. The schema categories are defined by the mapping properties `SchemaIn.Category` and `SchemaOut.Category`.

The Property element, which is declared in the default namespace, is automatically inserted first under the wildcard element. Its cardinality is set to non-mandatory and repetitive.

Mapping Example

Replacement elements and their child elements and attributes are managed similarly as any other element and attribute in a mapping. They follow the same mapping rules.

Note: You cannot link from or to a wildcard element.

This diagram shows a mapping example where data from an Infor BOD UserArea extension is retrieved:
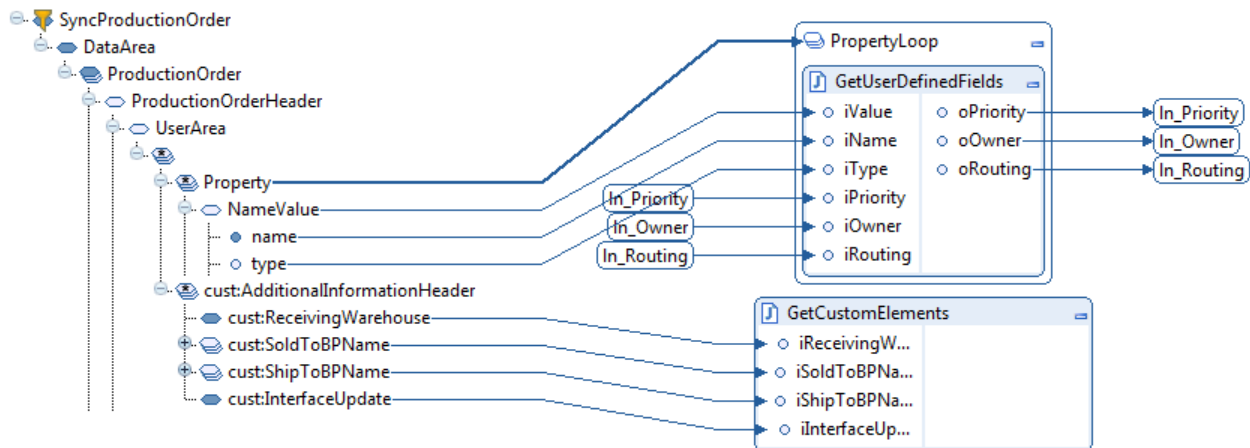
Data Translations

Infor Enterprise Collaborator (IEC) runtime provides built-in functionality for data translations. The M3 database stores translation data, which is managed by the M3 programs Business Message Data. Displ (CRS881) and Business Message Data .Translate (CRS882).

Data translation is normally required between an in-house system and an external system. Data translation allows the string data going to (outbound) or received from (inbound) systems to be translated from one value to another.

For example, between two partnering companies, the same things may have different representations, such as units of measure codes. One company uses a string data code STK, while the partner company uses the string data code PCS. Both codes refer to the same single unit of measure which means "styck" for Swedish and "pieces" for English.

Data Translation Configuration in Mapper

Data translations are defined in the mappings by using Java code. The user must identify which elements need to be translated. For the data translations, you specify the message identification and the message direction (inbound or outbound). This is the metadata used to identify the data translations.

Translation Data

When a mapping containing data translations is run for the first time, the data translations and their values are loaded from the M3 Database into memory. To use new translation data, you must reload the data translations.

For information on how to reload data translations, see the topic on IEC Management Pages in *Infor Enterprise Collaborator Tenant Administration Guide*.

Translation Logic

This logic applies to every data translation:

1 Find translation data for the given message context and the given partner. Given message context can be standard, type, or element path.
2 Find translation data for the given message context and the generic blank partner. Given message context can be standard, type, or element path.
3 Use the input value.

Using DataTranslator

The DataTranslator is required when communicating between an in-house system and an external system. Data translation allows the string data going to (outbound) or received from (inbound) systems to be translated from one value to another.

For more information about how to use the Java class DataTranslator, see the Javadoc available in the mapper Java editor.

Mapping management

A mapping contains only input and output document elements. Your task is to describe how the input document is translated to the output document by defining the complete transformation process. Describe the translation by adding a sequence of functions and linking them to the input and output document elements.

Mapping elements overview

The function sequence is executed from top to bottom every time an XML document is transformed. Data values are fetched from linked input elements, processed, and then sent to linked output elements, constituting the output document.

Use these guidelines in defining the transformation sequence:

• Create a transformation.

  You create transformation by adding a sequence of function elements in the space between the input and output document trees.

- Create links between document elements and function elements.

- Create links between function elements.

- Create links between variables or constants, and functions or documents elements.

Note: An output document is not mandatory, for example, in many inbound M3 messages, no output document is used because the data is stored in M3 by calling M3 APIs in the sequence. No response message is required.

Updating a MI program

If MI programs or BE programs are outdated in your mapping, you may have issues during publishing. For example, when you have a mapping or MI program with outdated fields, you will have "fields does not exist" errors during publishing. To avoid this publishing error, you should check if the mapping is updated.

Mappings that contain MI programs should be updated in order to function properly at runtime.

Update this and update all searches for the updated BE programs needed for your mappings to work. Update this, updates one BE program. Update all, updates all BE programs. Once you update, you can publish the updated mappings properly. Use these steps as a way to troubleshoot publishing errors.

To update a MI program using update this:

1   On the specified map, right-click the MI program.
2   Click **Update this**.
3   Specify the MI program
4   Click **Next**.
5   Specify the connecting BE.
6   Click **Next**.
7   Click **Finish**.

Updating multiple MI programs

If you want to update multiple MI programs efficiently, you must use update all.

To update multiple MI programs using update all:

1   On the specified map, right-click the MI program.
2   Click **Update all**.
3   Specify the MI programs.
4   Click **Next**.
5   Specify the connecting BE.
6   Click **Next**.
7   Click **Finish**.

Function elements overview

Functions define how to transform data from an input document and how to send the transformed data to the output document.

You can use input data and functions for calling information and the information returned by the function can then be written to the output document. Data from an input document can be, for example, an item number regarding a price request on a product. Other sources can also be used in functions.

Functions can be predefined or user-defined. For user-defined specific transformations, use Java code.

To define the transformation sequence, drag the elements from the Functions palette and drop them in the space between the input and output-schema trees. This action will create a stack of elements performed from top to bottom.

Follow these rules in defining the transformation sequence:

*   When connecting elements in the Mapping Editor, you can drop elements on acceptable areas only.
*   To complement the **Move Up/Down** menu options, use drag and drop to move function and loop elements up and down in the mapping.
*   Output data from functions can be used as input data in functions later on in the function sequence.

See

Core functions

You can manage maps and mapping elements using several core functions:

Loop

A loop is an iteration element controlled by a link. Loop functions can contain nested loops or other function elements. Use loops to create repeating elements in the output document based on conditions that are determined by the type of source elements of the connected link.

A loop can be dropped on the mapping area or on another Loop function, where the latter will create a nested loop.

If

An If function is a condition element controlled by a link. If functions can contain loops, other If functions, or other function elements. Use If functions to create elements in the output document based on conditions that are determined by the type of source element of the connected links.

An If function can be dropped on the mapping area, on a Loop function, or on another If function where the Loop and If functions will create a nested structure.

Boolean

Boolean function (UBJ) controls the loop iterations based on user-defined criteria. These functions are also known as "loop-controlling functions".

Note: The Boolean and Java functions are almost the same, except that Boolean function's output is a Boolean value. You define the criteria by writing Java code in the same way.

Java

Java functions can be dropped on the mapping or a Loop. A user-defined Java function (UVJ) processes data in the form of input parameters and sends the result to output parameters. Java code is used to define the processing that the function will do.

Repository

Repository Functions are reusable Java and Boolean functions stored in Function Repository that is part of a Mapper database. Reusable functions can be dropped on the mapping area or a loop.

Assign

Use this mapping element to make a direct assignment from an input document element to an output document element.

You can drag and drop an input element directly to an output element. In the Core Functions pane, you can also select and drag Assign in the Mapping Editor area and drop it in the mapping. Then, attach the input and output elements by dragging and dropping them to the Assign widget.

Note: The Assign function only assigns one value at a time. It can only be used with data elements or attributes in the input and output documents. A data elements is an element which do not have child elements.

Collector

A Collector function serves like a UVJ function with regard to input parameters. The input document elements and attributes, preceding functions' output parameters, and global variables and constants can be linked to Collector function.

Collector function stores your altered data into memory, so it can be sorted before the outmost restructure loop starts.

Restructure

A Restructure function serves as a UBJ function. Restructure function controls the loop iterations and can only be created in a loop. Only one Restructure function can be made within a loop and it must be the first in the loop's sequence, controlling the loop..

Note

Use this mapping element to insert specific textual descriptions in relevant places in your mapping. When you move the mouse pointer over a note, the note contents are shown. Inserted notes can be edited by double-clicking on the note function or through its context menu.

Variables and constants

Variables are used as intermediate data holders to transfer data to, or transfer data from functions. Use Variables to transfer values from functions on the upper part (earlier) to the lower part (later) of the execution sequence.

Constants are data holders for fixed data values to be assigned to function input parameters.

Palette behavior

Palette behavior contains tools and functions for mapping.

Using Tools

You can use Select, Marquee, and Link tools from the Tools palette:

Using the Select tool

Use this tool to select functions, parameters, elements, and links in the mapping.

- When you select a function, parameter, or element, the selected tool's icon and all connected widgets are highlighted, including the links.
- When you select a link, its connected function, parameters, and element are highlighted recursively.

Aside from the **Select** tool, you can use the **Links** tab in the Mapping Console to select links. It is more convenient to use the **Select** tool to create links in the mapping.

Use this example to create links in the mapping using **Select** tool:

1 In the Palette tool, click the **Select** tool.
2 In the mapping area, select the element, function, or parameter to link from.
3 Drag your selection to the element, function, or parameter to link to.

Use this example to select multiple items:

1 In the mapping area, press control-click or shift-click to select multiple items.
2 Right-click and select an action to perform on your selected items.

   For example, to delete many functions or parameters simultaneously, select **Delete**.

Using the Marquee tool

The **Marquee** tool works like the **Select** tool, but with the **Marquee** tool, you mark areas to select multiple items simultaneously.

Using the Link tool

The **Link** tool is used to create links in the mapping. Use the **Link** tool only to create a control link from a Boolean function to a Loop. A control link is automatically created when you insert the Boolean function to a loop.

Create links in the mapping using the **Link** tool:

1 In the Palette tool, click the **Link** tool.
2 Click the link from, element, function, or parameter.
3 Click the link to, element, function, or parameter.

   You can repeat steps 2 and 3 to create more links.

Note: A function, tool, or parameter is selected until you select another function, tool, or parameter.

Using Functions

When you add functions in the editor pane, a grey widget is displayed. The grey widget indicates where you can create a function.

These are the ways to use the functions in the palette:

- Click and drag a function in the palette to the editor pane.
   1 Hold-down the mouse button and point where you want to create the selected function.
   2 Release the mouse button to create the function.

   After this operation, the **Select** tool will always be selected in the palette.

- Click to select a function in the palette.
   1 Move the mouse over the editor pane to create a function.
   2 Click to select a function.

   After you click and select, you can create more functions of the same type without accessing the palette for each new function.

   The function is selected in the palette until you select another function, or tool.

Note: A function, tool, or parameter is selected until you select another function, tool, or parameter.

Adding input/output parameters

Use this procedure to add input/output parameters in a mapping. These parameters contain input or output arguments and can be dropped on a Java function, or on a Boolean function. Boolean functions should have input parameters only.

1 Double-click a mapping to open it in the Mapper Editor.

2   From the Core Functions pane, click and drag a function to add in your mapping.

   Note: When connecting elements, you can only drop elements where it is acceptable.

3   From the Parameters pane, click and drag an input or output parameter to add in a function in your mapping.

4   Right-click on a parameter and select **Properties**.

5   Specify this information:

   Description

   Provide a brief description for the parameter.

   Name

   Specify a descriptive name for the parameter.

   Default value: InParameter1

   Type

   Select the data type to use. `String, int, long, float,` and `more` .

   Default value: String

6   Select **File > Save** to save the parameter into your mapping.

To link an input document element to a function, or to link an output document element to a function, drag and drop the document element to the function, to create parameters. A parameter will be created with the same name as the document element with a link.

Adding a loop function

A loop can only be dropped on a mapping or on another loop function where the latter yields a nested loop.

See <ins>Function elements overview</ins> on page 230.

Note: When connecting elements, you can drop elements only where it is acceptable.

1   Double-click a mapping to open it in the Mapper Editor.

2   From the Core Functions pane in the **Palette** tab, select and drag **Loop** in your mapping.

3   Drag and drop the circular function element to the space between the input and output document trees.

4   Connect a controlling link for your loops.

   A loop can be controlled by a repeating input document element, a UBJ function, or a repeating external function (M3 API).

   Use the loop to create repeating elements in the output document by linking the loop to a repeating output document element.

   For example, you can drag a repeating input document element to a loop and then drag the loop to a repeating output document element. You can also drag the output element to the loop. The output repeating element will repeat as many times as the input repeating element does.

This table shows how the source element type of the link that controls the loop defines the loop condition:

Loop condition	Possible loop action
The source is a repeating input document element:	• Execute for each occurrence of the element (WHILE_TRUE, Execute loop FOR EACH element)
The source is a UBJ function:	• Execute while the function returns true (WHILE_TRUE, Execute loop WHILE function is TRUE) • Execute while the function returns false (WHILE_FALSE, Execute loop WHILE function is FALSE)
The source is an external function (M3 API):	• Execute for each record returned by the function (WHILE_TRUE, Execute loop for EACH record returned by a list program)

5   Right-click a loop and select **Properties**.

6   Select the condition for your loop.

   Note: Expand the Condition property to select the `LoopExecution` value.

7   Specify the information for your parameter properties:

   Condition

   See loop conditions in the table above.

   Description

   Provide a brief description.

   Name

   Specify a name.

   Default value: Loop

8   Select **File > Save** to save the loop into your mapping.

   Note: If you select a condition so that the loop will only run once, the Loop function becomes an If function.

Adding an If function

An **If** function can only be dropped on a mapping, or on another **Loop** or **If** function where the latter yields a nested **If** function.

See [Function elements overview](#) on page 230.

Note: When connecting elements, you can drop elements only where it is acceptable.

1 Double-click a mapping to open it in the Mapper Editor.

2 From the Core Functions pane on the **Palette** tab, select and drag **If** function in your mapping.

3 Drag and drop the circular function element to the space between the input and output document trees.

4 Connect a controlling link for your **If** function.

An **If** function can be controlled by an input document element or a UBJ function.

This table shows how the source element type of the link that controls the **If** function defines the **If** condition:

Condition	Possible If function
The source is an input document element, single or repeating	• Execute once if the elements exists (If_TRUE, Execute ONCE if element exists) • Execute once if the element does not exist (If_FALSE, Execute ONCE if no element exists)
The source is a UBJ function	• Execute once if the function returns true (If_TRUE, Execute ONCE if function is TRUE) • Execute once if the function returns false (If_FALSE, Execute ONCE if function is FALSE)

5 Right-click an If function and select **Properties**.

6 Select the condition for your **If** function.

Note: Expand the Condition property to select the `IfExecution` value.

7 Specify the information for your parameter properties.

Condition

See **If** conditions in the table.

Description

Provide a brief description.

Name

Specify a name.

Default value: `If`

8 Select **File > Save** to save the **If** function into your mapping.

Note: If you select a condition so that the **If** function will run multiple times, the **If** function becomes a **Loop** function.

Adding Java function

Java functions can be dropped on a mapping or a loop.

1   Double-click a mapping to open it in the Mapper Editor.

2   From the Core Functions pane, click and drag **Java** in your mapping.

    Drop it in the space between the output document trees to add a **UVJ** function.

    Note: When connecting elements, you can drop elements only where it is acceptable.

3   Define the processing to be done by the function.

    See [Editing Java and Boolean functions](#) on page 238.

4   Add links to input/output parameters to receive/output through **UVJ** function.

    With the parameters, you can now receive and output data through the links.

5   Save the function in the Java Editor and in the Mapping Editor.

    Note: You must save the changes in both Java Editor and Mapping Editor to save the changes made in your mapping.

Adding Boolean function

Use this procedure to add boolean function in a mapping. Also known as a loop controlling function, **UBJ** function controls the loop iterations. This function can be dropped on a loop. There can only be one boolean function within a loop and it must be the first in the loop's sequence, controlling the loop. The same is valid for **If** functions.

Note: The methods for **UBJ** and **UVJ** functions are almost the same, except that the **UBJ** function outputs a Boolean value, as well as output parameter values. You define the criteria by writing Java code in the same way.

See [Function elements overview](#) on page 230.

1   Double-click a mapping to open it in the Mapper Editor.

2   From the Core Functions pane, click and drag **Boolean** in your mapping.

    Drop it in the space inside a loop to add a **UBJ** function.

    The loop will automatically get the condition "Execute ONCE if function is TRUE". You can see this by the changed loop icon. If you want the loop to repeat many times, you must change the loop's condition to "Execute WHILE function is TRUE".

    Note: When connecting elements, you can drop elements only where it is acceptable.

3   Add input parameters to receive input data.

4   Define the processing to be done by the function.

    See [Editing Java and Boolean functions](#) on page 238.

5   Save the function in the Java Editor and in the Mapping Editor.

    Note: You must save the changes in both Java Editor and Mapping Editor in order to save the changes made in your mapping.

Editing Java and Boolean functions

Use the instructions here to edit Java and Boolean functions. Java compliance level must be 1.8 or better.

Note: The Mapping Editor and the Java Editor are two separate entities.

When you edit and save your code in the Java Editor and then from the Java Editor you switch back to the Mapping Editor, the Mapping Editor becomes cluttered because it detects that the Java code was edited. At this stage, it is not enough to only save the code in the Java Editor. You must also run save in the Mapping Editor to include your code changes with the mapping.

1 Double-click a mapping to open it in the **Mapper Editor**.
2 Double-click on the Java or Boolean function to edit the Java code. Alternatively, you can right-click on the function and select **Edit Java Code**.

   A new tab opens with a name format: `<Function_name>.java`

   This is the standard Eclipse Java editor.

3 Write your Java code.

   Use the standard content assist and formatting features in the body of the method.

   Note: Do not specify any code outside of the named method.

   All of the surrounding code is generated each time you edit the Java function, so any changes here are lost. This includes all automatic imports that the Java editor may do. If a class is not imported by the default imports, you have to write the Java package name before the class name, for example, `java.util.ArrayList`.

4 Save the Java code in the Java editor.

   Select **File > Save** or press **CTRL-S**.

   You cannot use all standard Java constructions due to security reasons. Only the most commonly used Java constructions can be used. Prohibited Java constructions are not marked in the Java editor. When you generate or publish the mapping, you will get an error message if you have used prohibited Java constructions.

Getting help from Content Assist commands

Before you can get help from the Content Assist command (Ctrl+Space), you must select Java Proposals.

1 Select **Window > Preferences > Java > Editor > Content Assist > Advanced**.
2 In Default Proposal Kinds pane, select the Java Proposals check box and click **Apply**.
3 Click **OK**.

   By default, Java Proposals is not selected. You can also check other proposal types and see how these are cycled through when content assist is repeatedly invoked.

Important notes on saving a mapping and java code

Refer to these notes:

- If the mapping was previously saved, notice that the mapping is unsaved, as indicated by an asterisk before the mapping name on the mapping tab. When you save the Java code in the Java editor, the code is inserted into the mapping, but the mapping is not saved.

  An invisible `.java` file is created in your project but the file is not used by the mapping. To save your Java code it is important to also save the mapping.

- If you add, remove, or rename function parameters, global variables, or constants, save your work and then close the Java editor tab. Then, edit the Java code again to reflect these changes.

  The Java skeleton is generated every time you select **Edit Java Code** for a function.

Creating and adding repository function

Repository Functions are reusable Java functions (UVJ and UBJ) stored in Function Repository which is part of a Mapper database. Reusable functions can be dropped on a mapping or a loop.

See [Function elements overview](#) on page 230.

Note: When connecting elements, the Mapper Editor will allow you to drop elements only where it is acceptable.

Before you start, you must define a Mapper Server connection to use repository functions.

Creating repository functions

1 Double-click a mapping to open it in the Mapper Editor.
2 From the Core Functions pane, click and drag the **Boolean** or **Java** function element.

  Drop it in the space between the input and output document trees to add a **UBJ** or **UVJ** function.

3 Add parameters to your function.
4 Name the function and its parameters, then edit the Java code.
5 Save the mapping with the new functions.
6 Right-click on the function and select **Save to Repository**.
7 Select a server location to contain this repository function and click **Finish**.
8 Confirm to insert a new repository function into the repository, or to overwrite an existing repository function.
9 Review the function version to save to the server and click **OK**.

  Note:

  - The version for a new repository function is `1.0`.

  - The category for a new repository function is `Default`.

Adding repository functions

1 Double-click a mapping to open it in the Mapper Editor.

2 From the **Palette** tab, select the **Repository** function element.

Drag and drop it to the space between the input and output document trees.

3 Select the Mapper server location to use and click **Next**.

4 Select a reusable function.

Optionally, click **Next** to view the input and output parameters.

5 Click **Finish**.

When completed, the selected function is inserted in the mapping complete with input and output parameters that you can now link to other elements.

Changing a repository function category

If you have a repository function in your mapping, use this procedure to change its category.

1 Right-click on the repository function and select **Properties**.

2 Click the **Properties** tab and change the `Category` value.

3 Save the mapping.

4 Right-click on the repository function and select **Save to Repository**.

5 Select a server location to contain this Repository Function and click **Finish**.

6 Confirm to overwrite an existing repository function.

Inserting a new version of a repository function

If you have a repository function in your mapping, use this procedure to create a new version of it.

1 Right-click on the repository function and select **Properties**.

2 Click the **Properties** tab and change the **Version** number.

3 Save the mapping.

4 Right-click on the repository function and select **Save to Repository**.

5 Select a server location to contain this repository function and click **Finish**.

6 Confirm to insert a new repository function into the repository.

Deleting a repository function

Deleting from database has no undo function. Before you delete, you must verify the category and version of a repository function.

Deleting from server

1   In the Mapping Editor, right-click on a repository function and select **Delete from Server**.

The **Remove Function from Repository** window is displayed.

2   Select a server location containing the repository function to delete.

3   Optionally, test the connection to check server connectivity.

4   At the successful Connectivity Test prompt, click **OK**.

Note: This action only tests for the connectivity to the server.

5   Click **Finish**.

6   At the confirmation prompt, click **OK**.

Note: The repository function is identified by its name and version. To delete another version of the repository function, you must first change the version of the repository function.

Deleting from workspace

In the Mapping Editor, right-click on a repository and select **Delete**. The selected function is now removed from workspace.

Links overview

When the sequence of function elements are run, data is transformed and transferred from the input document to the output document. The source and target of this data transfer is defined by connecting document elements and function elements with Data Links. In addition to Data Links, you can also use Control Link to control the iteration of a loop function.

This table shows the element connection details when you add links:

Element	Connection details
Document elements	Elements with multiple multiplicities (repeating elements) can be connected to (input) or from (output) a Loop. This means that the element multiplicity will control the iteration of the loop (input element) or the population of an output document (output element).
	Elements with single multiplicity can be connected to (input element) or from (output element) input/output parameters.
Boolean function	Must be connected to a loop.
Variables	Can be connected to input parameters, to output parameters, to input document elements or attributes, or to output document elements or attributes.
Constants	Can be connected to input parameters or to output document elements or attributes.

Adding links between function elements

You can create Data Links and Control Links between function elements.

Follow these rules:

- Data Links between UVJ and Repository functions output parameters to input parameters of UBJ, UVJ, and Repository functions

- Control Link between a function (M3 API or Database) and a loop

  This means that the occurrence of retrieved records controls the number of times that the loop will iterate.

- Control Link between a UBJ function and a loop

  This means that the Boolean result controls if the loop shall iterate or not.

Assigning variables or constants to function elements

You can assign variables or constants to function elements.

Follow these rules:

- Drag variables to input/output parameters of UVJ, UBJ, and Repository functions.

- Drag constants to input parameters of UVJ, UBJ, and Repository functions.

Note: You can also drag variables and constants directly to the documents.

Adding and removing links

Mapping elements contain constants and variables that you may link in the Mapper Editor.

Adding links

1   Double-click a mapping to open it in the Mapper Editor.
2   Click and drag a source document element to connect to the target function element.
3   Save the mapping.
   Note: The editor is context-sensitive to connections and will only allow you to create meaningful connections.
   Follow these limitations:
   - Crossing links to the output document is not allowed.

     Crossing output links will generate erroneous output documents. Crossing links from the input document is allowed.

   - Create Control Links from a repeating document element to a loop function, and the other way around.

     This means that the occurrence of a repeating document element controls the number of times the loop will iterate.

- Create Control Links from a loop to a repeating output document element.

  This means that the number of times the loop iterates controls the occurrence of the repeating element.

- Create Data Links from document elements to input/output parameters of UVJ, UBJ, and Repository functions or vice versa.

Removing links

When you delete a link between a variable or constant and a mapping element, only the connection is removed.

See Deleting variables and constants on page 245.

1 Double-click a mapping to open it in the Mapper Editor.
2 Right-click a connection, and select **Delete**.
3 Save the mapping.

Adding and deleting variables and constants

Variables and constants are used as parameter elements in the Mapping editor.

Adding variables and constants

1 Double-click a mapping to open it in the Mapper Editor.
2 Click the **Variables and Constants** tab in the **Mapping Console** view.
3 Right-click on a blank space on the **Variables and Constants** tab and select any of these options:

- Select **Create Variable** to create a variable with a default name `NewVariable1`.

  A **V**-icon indicates a variable. Assigning values to Variable are optional.

- Select **Create Constant** to create a constant with a default name `NewConstant1`.

  A **C**-icon indicates a constant. Assigning values to Constant are mandatory.

Note: A unique index is added to the default variable or constant name.

4 Specify `Value Type`, `Value` and `Description`.
5 Drag a variable or constant from the **Variables and Constants** tab to a parameter element in the Mapping Editor. Now, the variable or constant has been assigned to the parameter. You can also drag a variable or constant from the **Variables and Constants** tab directly to a document.
6 Save the mapping.

Adding variables and constants from parameters

An existing variable or constant can also be assigned to a parameter through the parameter's context menu. You can also create a new variable or constant through the parameter's context menu.

1   Double-click a mapping to open it in the Mapper Editor.

2   Right-click a parameter for a function in the Mapping. Select any of these options:

Option	Description
Assign Variable	Select this option to assign a variable to the parameter. The variable must have the same name and type as the parameter.
	If that variable does not exist, it is automatically created and assigned to the parameter. The variable will have the same name, data type, and description as the parameter.
	If that variable already exists, a confirmation dialog is displayed. Click **Reuse existing variable** to assign the existing variable to the parameter. Click **Create a new variable** to create a new variable to be assigned to the parameter. The new variable will have the same name as the parameter name with a unique index added, and the same data type and description as the parameter.

Option	Description
Assign Constant	Select this option to assign a constant to the parameter. This option is only available for input parameters, because a constant must have a fixed value and have the same name and type as the parameter.
	If the constant is non-existing, it is automatically created and assigned to the parameter. The constant will have the same name, data type and description as the parameter. The constant's value is set before it is created. A dialog is displayed where you set the new constant's value and a default value is given. The default value depends on the data type, for example, an empty string for the data type String. You can only set values that are valid for the parameter's data type. Click **OK** to create the constant.
	If the constant exists, a confirmation dialog is displayed. Click **Reuse existing constant** to assign the existing constant to the parameter. Click **Create a new constant** to create a new constant to be assigned to the parameter. The new constant will have the same name as the parameter name with a unique index added and the same data type and description as the parameter. The view shows a dialog where you set the new constant's value, as described above.

3 Save the mapping.

Deleting variables and constants

If you delete a link between a variable or constant and a mapping element, only the connection is removed and not the variable or constant.

1 Double-click a mapping to open it in the Mapper Editor.

2 Click the **Variables and Constants** tab in the Mapping Console View.

3 Right-click a variable or constant and select **Delete**.

4 Optionally, select more than one variable or constant to delete simultaneously.

Right-click any of the selected items then select **Delete**.

The selected variables or constants are removed from the current mapping.

5 Save the mapping.

Note: Click in the editor pane to undo the deletion of variables or constants.

Editing mapping element properties

Use this procedure to edit mapping elements in Eclipse Properties view.

The information in the contextual Properties tab change, depending on your currently active selection in Mapping Editor. It can be properties on functions, elements, or mapping. In this procedure, the Mapping Loop is used as an example.

1 Double-click a mapping to open it in the Mapper Editor.

2 Right-click a loop to edit and select **Properties**.

3 Select and expand the view of Condition property.

4 Select a value to use for **LoopExecution** property.
  Select from these options:
  - Execute loop FOR EACH element
  - Execute loop ONCE if element exist
  - Execute loop ONCE if NO element exist

5 Select **File > Save**.

Customizing message partner IDs

You can set the partner name using the IEC Mapper. This function requires you to use the `setMani` `festInfo (ManifestConstants.MAP_PARTNER, "partner_ID_name")`. Follow these steps to search messages and display results from a specific partner ID.

1 Double-click a mapping to open it in the Mapper Editor.

2 Add `setManifestInfo (ManifestConstants.MAP_PARTNER, "partner_ID_name")` in the code where applicable. "partner_ID_name" must be replaced with the name of the specific partner.

3 Save your changes.

4 Save the mapping.

5 Republish the mapping.
  The specified partner will show among the search results. Clicking on **Show** of the specific message will also show the partner ID in the **unfiltered** message manifest information.

Restructure function

This section guides you on how to use the Restructure function using three examples. References to in-depth technical details are cited in the examples.

See Restructure concepts on page 257.

In these examples, data from the input document is used directly. You can also use Collector functions to alter the data before it is restructured.

See Collector function on page 264.

When mapping data between two different documents, the document structures are not always the same. While the input document contains all data needed for the output document, it is structured in a way different from what is required for the output document.

These are the example scenarios that we will use to describe the Restructure function:

- Example 1
- Example 2
- Example 3

Example 1

In these examples, we will use time reporting to restructure data in different ways.

The structure of the input data:

```
Timesheets
    Employee
        FirstName
        LastName
        EmployeeNumber
        Time
            Date
            Project
            Hours
            Comment
```
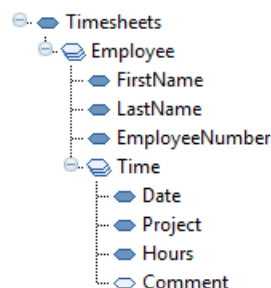
In this example, we want the time transactions in a flat structure:

```
TimeTransactions
    Transaction
        EmployeeNumber
        Date
        Project
        Hours
        Comment
```
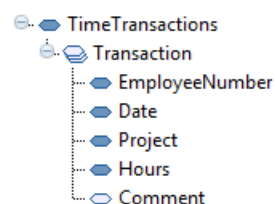
This can be done using manually created standard Java code in Java functions. The task can be complicated and technically advanced. This is true if the output document requires a lot of nested loops

with grouping and maybe summing, as in our next examples. The mapping will be cluttered with technical functions such as `AddToRecord`, `AddToArray`, `SortArray`, `CheckBreak`, and `GetFromArray`.

The Restructure function does all these without the need for the user to create a single line of Java code. At runtime, all the processing is completed in a similar but more complex way. It is similar to solving the problem with a manually created Java code in the mapping.

See Restructure concepts.

Example 1 steps

1 Create loop.

    a We need to loop through all time transactions to create the repetitive Transaction element in the output document. Start by creating a loop that controls the output element *Transaction*.

    b Because we want to restructure the input data, we also insert a Restructure function in the loop. It will automatically control the loop.
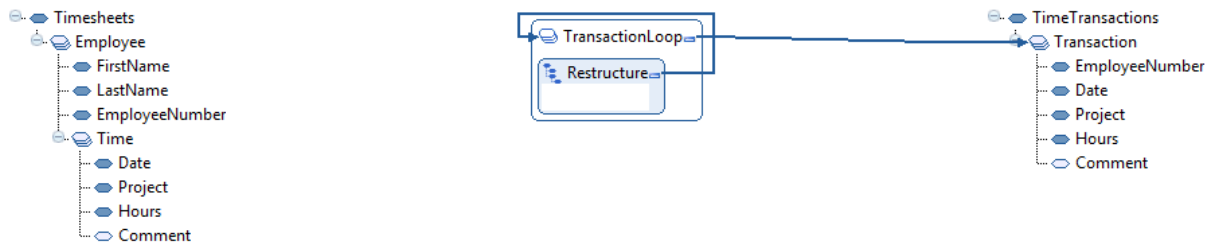
2 Select data.

    a Drag and drop all data elements to be used from the input document to the Restructure function.

        At this stage, we do not need to think about the repeating elements in the input document.

    b Link the corresponding output parameters to the output document.
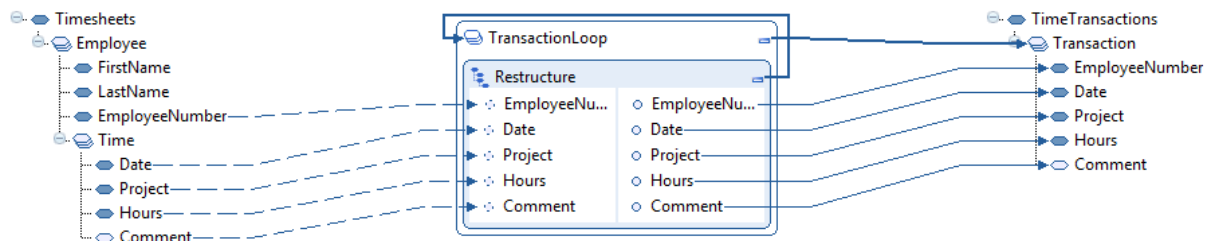
3 Select input loop levels.

You must instruct the Restructure function which repeating elements in the input document we want to loop on when reading the input data.

In this case, this is what we want to do:

• Loop on Employee, because we want data for all employees.

• Loop on Time, because we want data for all time transactions per employee.

This table shows the movement for the elements:

Element	Movement
Both *Employee* and *Time*	• From the input document • To the Restructure function's header
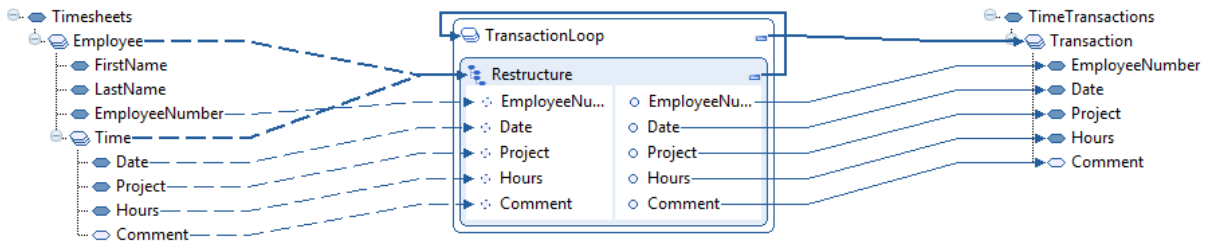
4 Sort the data.

To restructure the data, you must sort it and decide the sorting sequence.

See information about sorting in.

To sort, set the property Sorting Sequence for the input parameters.

Follow this sorting sequence for the input parameter:

Input Parameter	Sorting Sequence
EmployeeNumber	1
Date	2
Project	3

5 Complete the procedure by performing these steps:

    a Save the mapping to the mapper database.
    b Publish the mapping.
    c Set up an agreement.
    d Other required procedures, then test the mapping.

Example 2

Our second example is more complicated. In this example, we will use the same input structure to group the time transactions per day (*date*) and employee.

See Grouping topic in Restructure concepts.

This diagram shows the output structure:
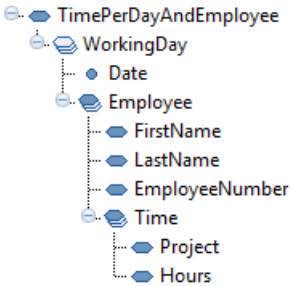
1 Create loops.

In this example, the loops will control the three repetitive elements in the output document.

This diagram shows that the nested loops are controlled by Restructure functions.
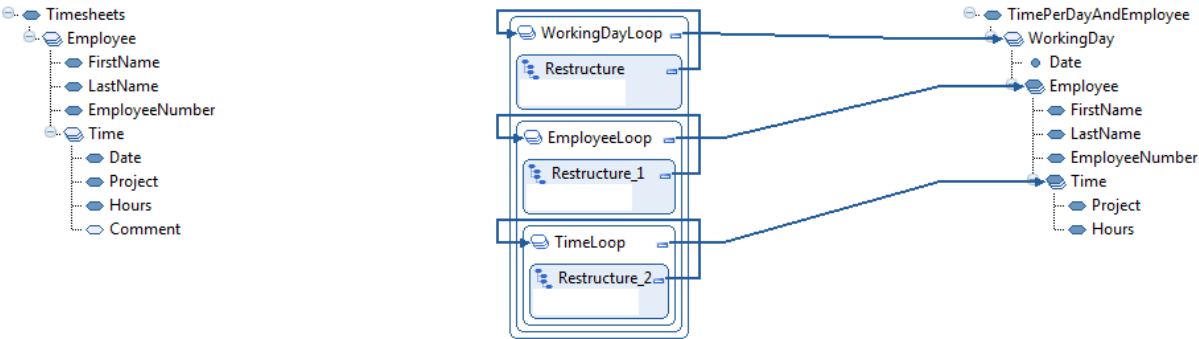
2 Select data.

Drag and drop all data elements to use:

- From the input document to the Restructure functions

- From the Restructure functions to the elements and attribute in the output document
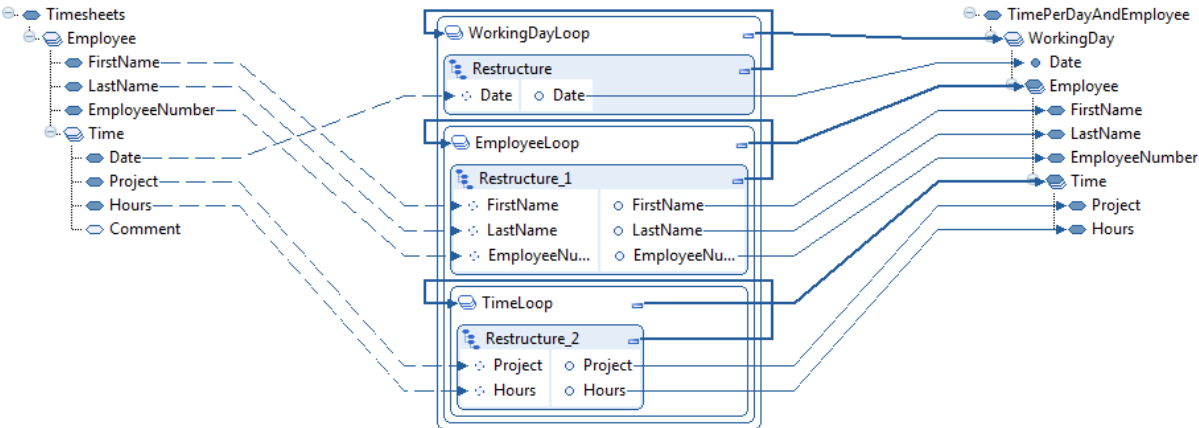
3 Select input loop levels.

This table shows the movements of the elements:

Element	Movement
Time	To get all the dates, drag and drop the Time element to loop on this element from the input document to the first (outmost) Restructure function's header.

Element	Movement
Employee	To get all the dates, drag and drop the Employee element to loop on this element from the input document to the second (middle) Restructure function's header.
Time	To get all the dates, drag and drop the Time element to loop on this element from the input document to the last (inner) Restructure function's header.

This diagram shows the completed input loop levels:
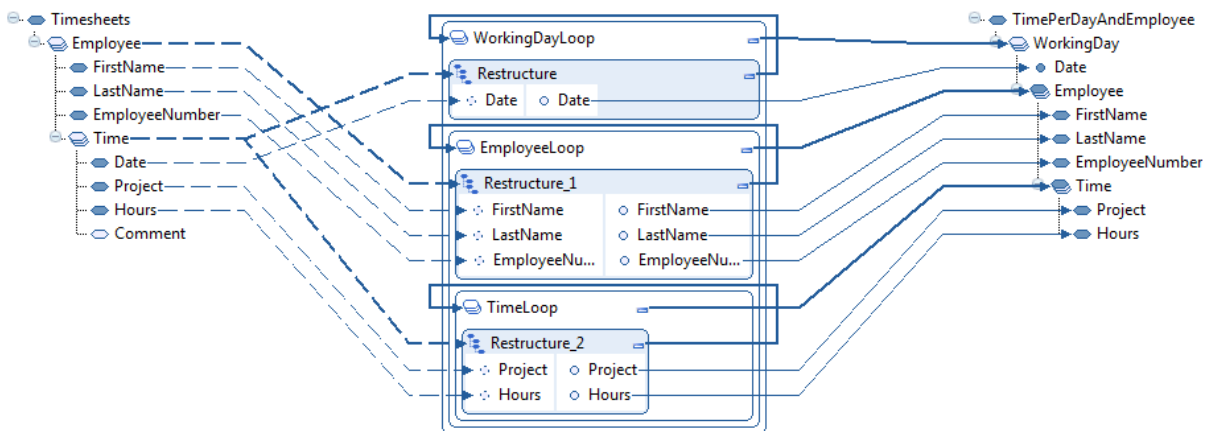
4   Sort and group the data.

Set the properties for the input parameters to get the correct sorting and grouping.

- In the first restructure function, to make the outmost restructure loop, create one *WorkingDay* output element per unique date for all time transactions.

  Follow these property settings:

Property	Setting
Sorting Sequence	1 for Date
Is Grouped	true for Date

- In the second Restructure function:

  To create one Employee element per unique combination of first name and last name, follow these property settings:

Property	Setting
Sorting Sequence	1 for FirstName 2 for LastName
Is Grouped	true for LastName

- In the last Restructure function:

  To sort the time transactions on project per employee and date, follow these property settings:

Property	Setting
Sorting Sequence	1 for Project

To avoid duplicate transactions for a project per employee and date, apply these property settings in the last (inner) Restructure function:

Property	Setting
Is Grouped	true for Project
Is Summable	true

Example 3

In this example, we want to create a time summary per project with the option to show the transactions per date.
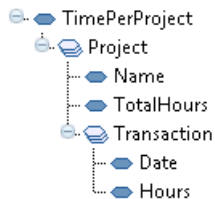
Creating a time summary per project

1 Create a loop controlling the repetitive Project element in the output document.

The loop is controlled by a Restructure function.

2 Select data and input loop levels

Follow these element movements:

Element	Movement
Project	**a** Drag and drop in the input document to the Restructure function **b** Link the corresponding output parameter to the `Name` element in the output document.
Hours	**a** Drag and drop in the input document to the Restructure function **b** Link the corresponding output parameter to the `TotalHours` element in the output document.

Element	Movement
Both the Time and Employee repeating elements	Drag and drop from the input document to the Restructure function's header. This will get all time transactions for all employees.

Note: If we forget the link from the Employee element, we only get time transactions for the first employee.
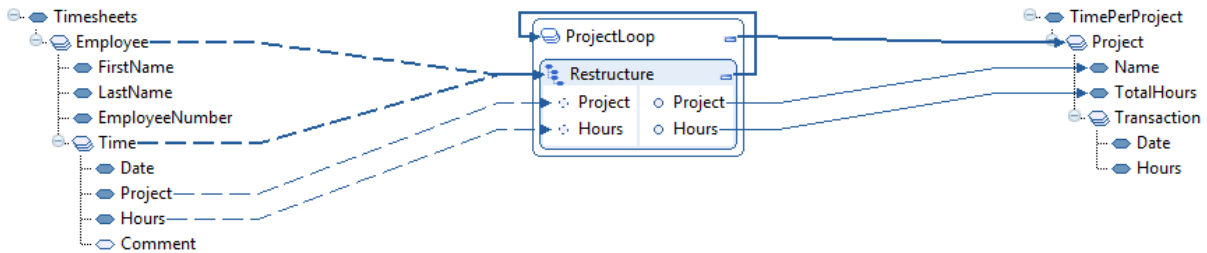
3 Set and group the data.

Follow these property settings:

Property	Setting
Sorting Sequence	1 for project
Is Grouped	true for project
Is Summable	true for Hours

Now, you have created a mapping for a simple time summary per project.

Adding the time transaction details

Use these steps to add the time transaction details.

1 Create a nested loop controlling the repetitive Transaction element in the output document.

The loop is controlled by a Restructure function.

2 Select transaction data and input loop.

Follow these element movements:

Element	Movement
Date	**a** Drag and drop from the input document to the second inner Restructure function **b** Link the corresponding output parameter to the Date element in the output document

Element	Movement
Hours	**a** Drag and drop from the input document to the second inner Restructure function **b** Link the corresponding output parameter to the Hours element in the output document
Time	Drag and drop the **Time** repetitive element from the input document to the second (inner) Restructure function's header. This will get all time transactions per project, as handled by the outer Restructure function.

Note: We do not need to link from the **Employee** repetitive element again. It is already linked to the first (outer) Restructure function. So the restructure logic already knows that we want to loop on **Employee** in the input document, but it will not do any harm to link the **Employee** element once again.
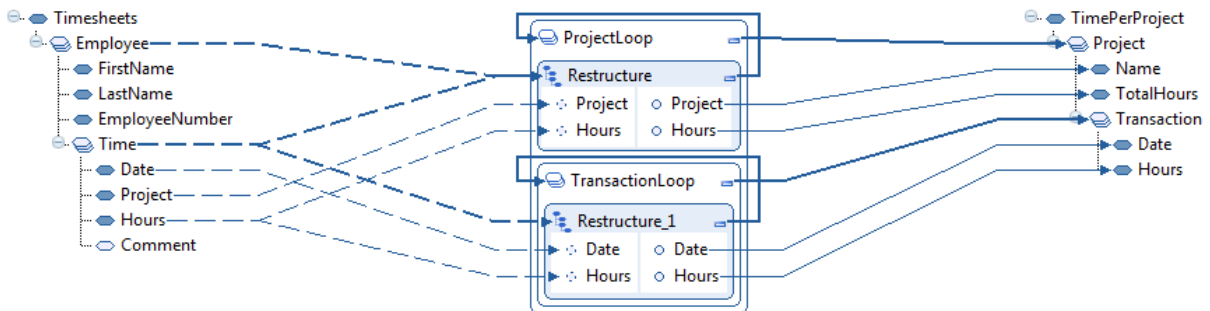
3 Sort the data.

Follow these property settings:

Property	Setting
Sorting Sequence	1 for Date This will sort the transactions per date and per project.

Adding a Restructure function in a Mapping

A Restructure function behaves as a UBJ function.

Consider these function behaviors:

- It controls the loop iterations.
- It can only be created in a loop.
- There can only be one Restructure function within a loop and it must be the first in the loop's sequence.

Use this procedure to add a Restructure function in a mapping. Repeat this procedure for nested restructure loops, if necessary.

1   Double-click a mapping to open it in the Mapper Editor.

2   In the Core Functions pane, select **Restructure**.

Drag and drop it in the space inside a loop where the Restructure function must be added.

A control link from the Restructure function to the loop is automatically created.

Note: When connecting input elements, you can drop elements only where it is acceptable.

3   Add input parameters to receive input data.

You can use data directly from the input document or from Collector functions.

To create an input parameter, follow these instructions:

- To use data directly from the input document, click and drag a data element or an attribute from the input document to the parameter area.

- To use data from a Collector function, click and drag an output parameter for a Collector function to the parameter area.

A corresponding output parameter is automatically created.

The input links to Restructure functions are dashed to indicate that data is not read when executing the restructure function. Instead the data has already been read.

4   Add input control links to the function header.

This is only valid when you are using data directly from the input document. Control links to function headers when using data from collector functions, are not allowed.

Add this to indicate repeating elements in the input document that must control loops when reading the data. The icon for Restructure input parameters, similar with Collector output parameters, is dashed to indicate reference parameters.

Click and drag a repeating element from the input document to the function header. A bold dotted input link is created.

You can create several control links to the Restructure function, if needed.

5   Link the output parameters that contain the restructured data.

Click and drag an output parameter to a data element or an attribute in the output document or vice versa.

You can also assign an output parameter to a global variable or click and drag the output parameter to an input parameter for succeeding function.

6   Define the restructure processing.

Use these property settings for the input parameters:

Property	Details
Sorting Sequence	Settings for the sorting sequence: • `0` for no sorting • A positive integer to sort the data A given sorting sequence `>0` must be unique within the function. The default value for this property is `0`.
Sort Order	If sorting sequence is `>` `0,` set the sort order: • Ascending • Descending The default value for this property is **Ascending**.
Is Grouped	If sorting sequence `>0`, use these grouping settings: • **false** no grouping • **true** grouping of the data The default value for this property is **false**. You can only set this property to **true** for the input parameter with the highest sorting sequence. If a sorting sequence is set for two or more input parameters, data will be grouped on the unique combination of these parameter values.
Is Summable	• If another input parameter is grouped and the data for the current input parameter is numerical, you can set this property to **true** to automatically sum the data. • If another input parameter is grouped and this property is set to the default **false**, data is used from the first record in the current group according to the total sorting. • If no input parameter is grouped, this property has no effect. **Note:** • If you set an invalid value for a property, the value will automatically revert back to its previous value. • If you click **Tab** after setting an invalid property value, an error message is displayed in the status bar.

7   Save the function in the Mapping Editor.

For a nested Restructure loop structure for all Restructure functions within the Restructure loop structure, you must directly use data from the input document or from Collector functions. You cannot, for example, use data from the input document in the outer Restructure loop and use data

from Collector functions in the inner Restructure loop. Because data would then come from different contexts, there is no structural relationship between the input document and the Collector functions. To restructure the data, ensure that all input data belongs to the same structure.

Restructure concepts

Restructure function discusses the details behind running functions. References to these input document and collector loops will be used.

Input document
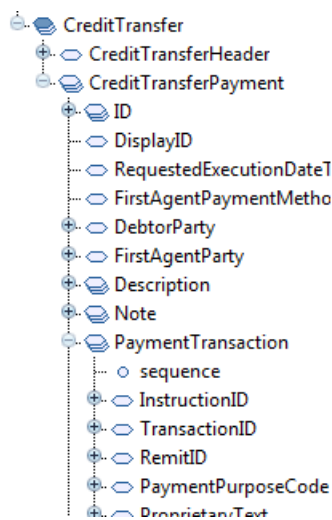
Collector loops
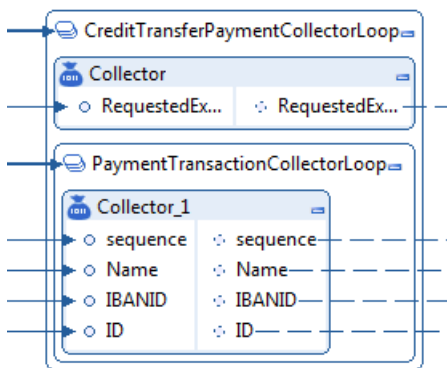
Restructure function

Normally, a function (Java, Boolean, Database, API, and others) is executed as defined in the sequence. If the function has input parameters that are linked to elements and attributes in the input document, here is what happens:

- The corresponding data is read from the input document, from preceding functions, or from variables.
- The function is run, for example, the manually created Java code.
- If the function has output parameters, data created when executing the function is written to elements and attributes in the output document or assigned to variables.

These steps are run in real time per function in the sequence: read data, run function, and write data.

The Restructure function is different. The last two parts are the same as described above; the function is executed and the output parameters fetch data that is linked to the output document or assigned to variables. This data is not fetched from the input document, even though the input parameters have to be linked to elements and attributes in the input document. If the input parameters are linked to Collector functions, data is not fetched from these. Instead, the data is fetched from memory where it has been restructured. So, when is the data read from the input document or from the Collector functions?

You can compare the Restructure function to a Boolean function. It can only exist first inside a loop and it has to control the loop. This type of loop is now called a **restructure loop**. You can have nested restructure loops, each controlled by a separate Restructure function.

Before the outmost restructure loop begins to execute, the data is prepared. If data is used directly from the input document, it is read from the input document through an invisible "automatic mapping" that, if visible, would look more or less like the mapping you would manually create to read data from the input document. If the input document contains two nested repeating elements, for example *CreditTransferPayment* and *PaymentTransaction*, the "automatic mapping" will have two nested loops controlled by these two input elements. Within each loop, data is read from input data elements and attributes. Only the references to the data are stored in memory, the input data is never duplicated. This will keep the memory consumption low for the restructuring process.

If data from Collector functions is used, you have to manually create a mapping to collect the restructure data. Collector functions are used to store references to the data in memory. If you use data directly from the input document, the collector mapping will look similar to the "automatic mapping" described here. In this case, it is easier to use data directly from the input document, because you do not need to create the collector mapping. Although, if you need to alter the data before it is restructured, you must create the collector mapping, for example, removing time to sort and group data on date, converting amounts, or maybe excluding records.

See the [Collector function](#) on page 264.

Whether the data is read from the input document or from Collector functions, it is structured data. The input schema defines the structure for the input document. XPaths for elements and attributes are used as internal identifiers. For Collector functions, the function paths and the parameter names are used as internal identifiers.

Follow this example:

- The collector path "`/CreditTransferPaymentCollectorLoop/PaymentTransactionCollectorLoop/Collector_1/Name`"

  corresponds to:

- The input document XPath "`/SyncCreditTransfer/DataArea/CreditTransfer/CreditTransferPayment/PaymentTransaction/Name`".

These two different structure contexts cannot be mixed. For example, you cannot combine the element **RequestedExecutionDateTime** from the input document with Name from Collector_1 because they have no structural relationship. You do not know if the Collector parameter Name is linked from the input document, or if it is created in some other way. This means that for a nested Restructure loop structure you must use data, either directly from the input document or from Collector functions, for all Restructure functions within the Restructure loop structure. All input data must belong to the same structure to be able to restructure the data.

When you have the data in memory, in its original structure, the data or data references are restructured. Then, the outmost restructure loop executed is controlled by the Restructure function. Restructured data is fetched from memory and assigned to the output parameters. If there are nested restructure loops, these are also executed like any other nested loop. The nested Restructure functions automatically keep track of the restructure context. For example, only the payment transactions for supplier A are given if there is an outer restructure loop grouping the suppliers, and supplier A is the current supplier.

Defining output data to write

The structure for the output document dictates the structure of the loops and functions in the sequence because these functions create the output document. If you want to loop through all suppliers, follow these guidelines:

*   Create a loop that controls the repeating element for supplier in the output document.
*   Create a Restructure function in the loop.
    The Restructure function automatically controls the loop.

If the output document has a nested structure, for example payment transactions per supplier, follow these guidelines:

*   Create the corresponding nested loop controlled by another Restructure function.
*   Define the input data to use.

Defining input data to read

The automatic mapping, which is the data to read from the input document, is defined by the links from the input document to the input parameters and function headers of the Restructure function. Alternatively, if you use Collector functions, you create the corresponding mapping manually.

You can only create input parameters for a Restructure function in any of these methods:

*   Linking elements, or attributes, in the input document to the Restructure function
*   Linking Collector functions' output parameters to the Restructure function, where input parameters are automatically created, as for Java and Boolean function

After you create input parameters, these links are dashed and not solid like the other links. This is to indicate that data reading does not happen at that exact place in the sequence, but earlier. Think of

these links as use this data, or references. The icon for Restructure input parameters and for Collector output parameters is also dashed to indicate reference parameters.

When an input parameter is created, an output parameter with the same name is automatically created.

Follow these rules that apply to the parameters:

*   If you rename the input parameter, the corresponding output parameter is also renamed.
*   If you delete the input parameter, the corresponding output parameter is also deleted.
*   If you move the output parameter, the corresponding input parameter is also moved.

The restructure process does not alter the data in any way, except when numerical data is summed, so we get this parameter pairing. The input and output parameters always have the `String` data type.

When an input parameter is highlighted, the corresponding output parameter is also highlighted and the other way around. This is highlighted because the data is not altered in the function. Select the source for an input parameter to see where it is linked to after the Restructure function. This highlighting mechanism is also used for Collector functions. The data flow is highlighted from the selected data source through the Collector function and the Restructure function or functions to its destination or destinations.

Loop in the input document

This section is valid only if you use data directly from the input document.

For the Restructure function, you must create links by linking repeating elements in the input document to the function's header. Drag and drop (or use the Link tool) a repeating input element to the blue area of the Restructure function, for example, to the icon. This is a control link, so the line is bold. The link is also dashed like the data links.

When the automatic mapping is executed, it must be known which input elements should control the loops. That is, on which input elements you want to loop (the loop level). For the credit transfer example, you loop on the repeating elements `CreditTransferPayment` and `PaymentTransaction`. Think of these links as "I want to loop on these elements".

Follow these guidelines:

*   Every Restructure function must have at least one loop link and one data link (input/output parameter).
*   You can link the same repeating input element to several Restructure functions.
*   You can link an input data element to several Restructure functions.

The input loop or control links do not need to be nested in the same way as the restructure loop. Otherwise, this would not be restructuring.

If you have a loop level (a repeating element) in the input document that only contain an inner loop (repeating child element), and you do not need any data from that outer loop level, you still need to link that repeating element to a Restructure function. Otherwise, the automatic mapping will not loop on that level and will only read the first instance.

So, why link input elements to Restructure functions in this way? To create the automatic mapping that reads data from the input document, you need all XPaths for the repeating elements controlling the

loops and all XPaths for the data elements to read. The order in which the XPaths are defined does not matter because the actual XPaths can be sorted to create correct metadata for the automatic mapping.

In this way, the input document's data structure is decoupled from the desired (output document) data structure.

Loop for Collector functions

This section is valid only if you use data from Collector functions.

When you use Collector functions, you manually create the loops, so you control the looping. The Restructure functions do not need to know about the loops in the original data structure, the data structure from the collectors is decoupled from the desired (output document) data structure.

Note: You can only use data from one nested Collector loop structure in a nested Restructure loop structure. Within a mapping, you can use several pairs of Collector and Restructure loop structures, or Restructure loop structures separately by using data directly from the input document.

Defining data to be sorted

Before you can restructure the data, you must sort the data.

These are the input parameter properties for sorting:

Sorting Sequence

By default, this property is set to `0`, which means no sorting. When Sorting Sequence is set to a positive integer value, the data is sorted on that element. You can set Sorting Sequence for several input parameters for a Restructure function.

Follow these example sorting sequence:

* Sort on supplier name (*Sorting Sequence = 1*)
* Sort on IBANID ()

Note: A Sorting Sequence value that is `Sorting Sequence = >0` must be unique within a Restructure function.

Sort Order

If Sorting Sequence is >0, you can set the property **Sort Order** to **Ascending** or **Descending** sort order.

Note: All data are handled as strings and sorted lexicographically according to the Unicode Collation Algorithm (UCA). For more information on UCA, see `http://unicode.org/reports/tr10/`.

For example, `11` is less than `2` due to the different lengths for ascending sort order.

The null value for a non-existing input element is less than the lowest value for an existing input element. This means, null is less than an empty value and any other value for ascending sort order.

These input parameter properties decide the sorting for the data at the current loop level. The total sorting is constructed from the sorting properties for Restructure functions for all nested restructure loops. See this example:

If data for the outmost restructure loop is sorted on:

*   1 *supplier name*
*   2 *IBAN ID*

And, if data for the inner restructure loop is sorted on:

*   *date*

This becomes the total sorting:

1 `supplier name`,

2 `IBAN ID`, and

3 `date`

Defining data to be grouped

In this example, you have an outer restructure loop for supplier, so you may also want to group the data on supplier.

If the input parameter property `Sorting Sequence` is >0, you can set the property `Is Grouped` to `true`.

If you set `Sorting Sequence` for *supplier name* to `1`, and `Is Grouped` to `true`, then you will get one restructure loop iteration per unique supplier name. You can also group several elements.

To get one iteration per unique combination of *supplier name* and *IBAN ID*, follow these property settings:

Property	Setting
`Sorting Sequence`	1
`Is Grouped`	`false` for *supplier name*
`Sorting Sequence`	2
`Is Grouped`	`true` for *IBAN ID*

Note: You can only set the property `Is Grouped` to `true`   for the input parameter with the highest sorting sequence number.

Defining data to be summed

If you sort payment transaction on *date*, you will get one iteration per payment transaction in date order. If setting the property `Is Grouped` to `true` for the *date* parameter, you will instead get one iteration per unique date. If you also have an *amount* parameter in that Restructure function, you will get the amount for one of the payment transactions per unique date as default.

By setting the property `Is Summable` to true for the *amount* parameter, you will get a summed amount for all payment transactions per unique date. In order to make this work, you must ensure that the linked input amount element always exists, that is not null, and contains a valid integer or decimal number. Otherwise, the mapping will terminate abnormally at runtime. There is no decimal rounding, or other number processing performed.

Note: The summing is made using the Java class `BigDecimal`.

Restructure limitations

If you use the data directly from the input document, data is read from the input document to memory by the automatic mapping. This means, you cannot modify the input data before it is restructured, neither apply logic on it in any other way. After the restructuring, you can modify the data in the mapping. But if you need to do modify the input data before it is restructured, use Collector functions.

These examples illustrate where you can use the Collector functions:

- To substring a value so that all values have the same length to enable correct sorting and grouping, or if you need to group on only a part of the values.

  One example is to sort and group on the date part of an element that contains date plus time.

- To use the instance with a specific attribute value for a repeating data element that is not looped in the restructuring process.

  Such as, to always use the main ID, or the English description.

- To sort the value for either element `A` or element `B`, depending on the value of another element.

  Then, you can create a mapping logic that assigns either element A, or element B, to an input parameter for a Collector function.

- To sort the values from two input elements that are included within a choice as one value. That is, only one of the elements can exist at a time.

  Then, you can create a mapping logic that assigns the element that is not null to an input parameter for a Collector function.

- To convert a negative amount to a positive amount for correct summing.

- To filter out data.

  One example is to skip all transactions with a negative amount.

The current version does not support restructuring of parallel repeating elements or loops in the input document or in the Collector structure. It is not possible to restructure, for example, a repeating element for notes plus a repeating element for payment transactions, both per credit transfer payment, even if

you only want to move the notes loop to the output document as it is. Support for this will be added in a future version.
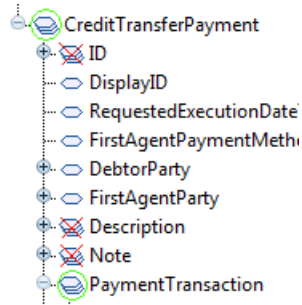
Follow these guidelines:

- If you use data directly from the input document, then all loop levels must be given in the input document instance.

  For example, if the complex element *PaymentTransaction* is not given for a *CreditTransferPayment* instance, the *CreditTransferPayment* instance will not be included in the restructured data.

- If you use data from a nested Collector loop structure, then the inner loop must always iterate at least once.

Collector function

The Collector function is used to store your altered data in memory so that it can be restructured. It is assumed that you already know how the Restructure function works. In this example scenario, we will describe how to use the Collector function.

- [Example 4](#)

Example 4

This scenario builds on

The prerequisites are exactly the same as before, we want to group the time transactions per day (date) and employee. The problem is that the input element *Date* now contains a time stamp, for example `2013-06-17T16:43:23`. The time transactions may be created at different times per day. This setting makes it impossible to group the time transactions per date using the input data as it is. We need to remove the time part of the time stamps before restructuring the data. To achieve this, we must use Collector functions.

Example 4 steps

1   Create collector loops.

To collect the input data, we need two nested loops, one for the repetitive element **Employee** and one for the repetitive element **Time**

The loops are controlled by these elements. Insert one Collector function into each loop.

2   Collect data.

a   Drag and drop the input elements **FirstName, LastName**, and **EmployeeNumber** to the first Collector function.

b   Because we want to modify the date, we create a **UVJ** function.

Insert a UVJ function into the inner loop before the second Collector function.

Drag and drop the input element Date to the Java function and create an output parameter oDate.

c   Create a logic in the **UVJ** function to remove time, so that the input date and time `2013-06-17T16:43:23` will be converted to `20130617`. Use the EDIFACT format `"102"`, that is the format **CCYYMMDD**. This format is equal to `D8` for ANSI X12.

For example, use this Java code:

```
oDate = new DateTimeConverter().setXMLDateTime(iDate).getEDIDate
Time("102");
```

d   Create a new input parameter **Date** in the second Collector function.

Drag and drop the Java function's output parameter **oDate** to the Collector function's input parameter **Date**.

e   Drag and drop the input elements **Project** and **Hours** to the second Collector function.
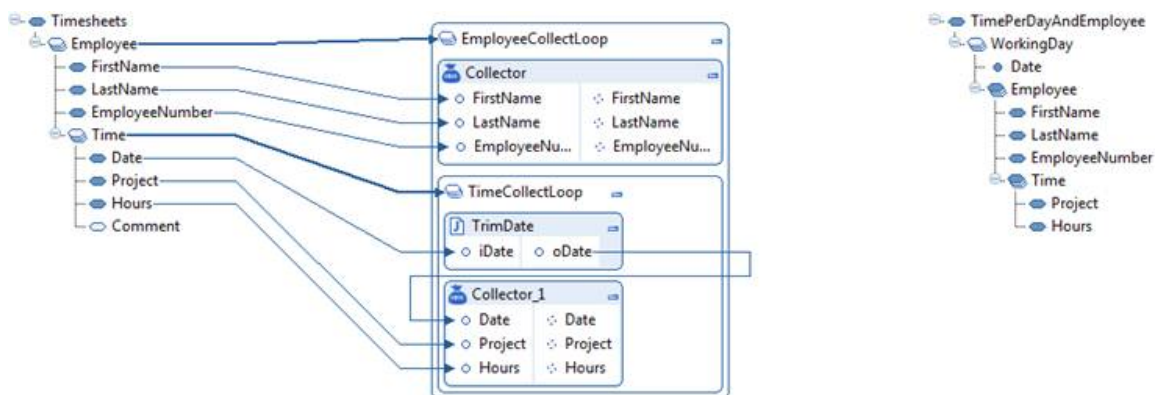
3   Create restructure loops.

This step is the same as **step 1** in Example 2. Create the three nested restructure loops after the collector loops.

4   Select data.

Instead of using the input document's elements, we use the Collector functions' output parameters.

a   Drag and drop these parameters:

- **Date** from the second Collector function to the first Restructure function

- **FirstName** from the first Collector function to the second Restructure function, and so on

b   Because we are restructuring the data, you do not need to match the loop levels for the Collector functions and the Restructure functions. Additionally, you have already created the collector loops for this purpose, so you do not need to select input loop levels for the Restructure functions as you do in Example 2.

c   Drag and drop from the Restructure functions' output parameters to the elements and attribute in the output
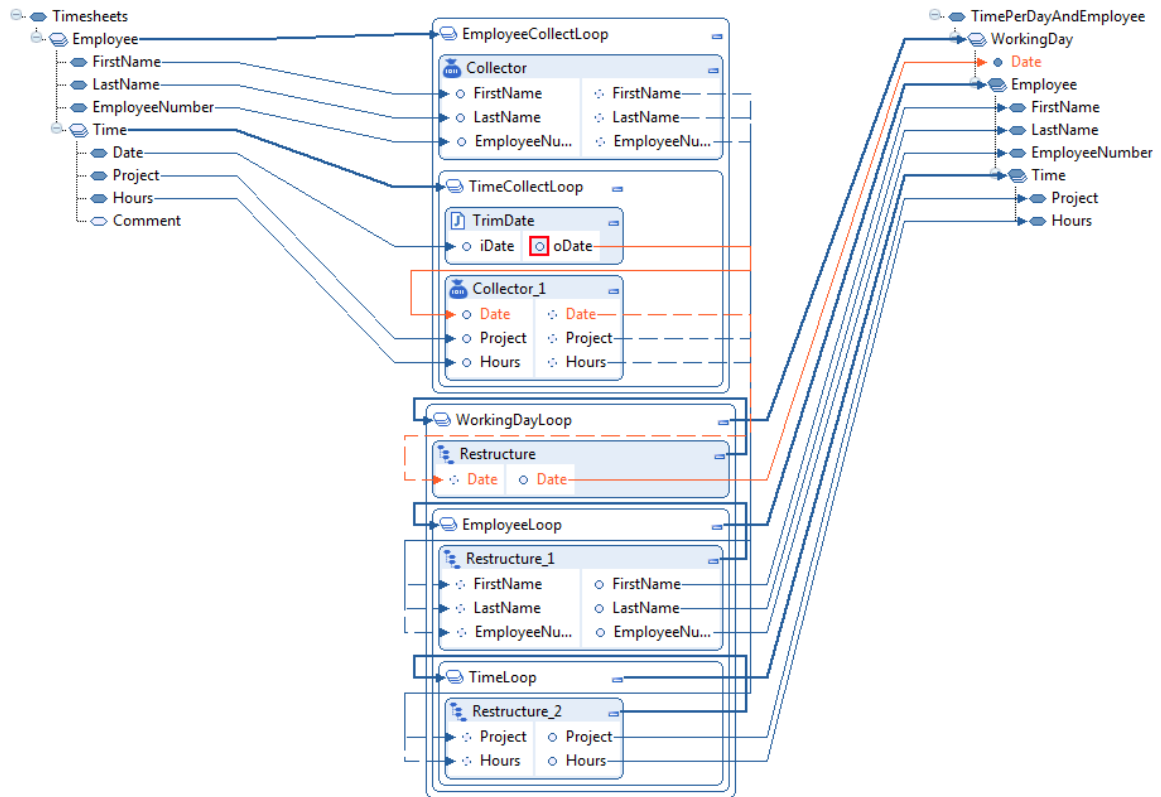
5   Sort and group the data.

This step is the same as **step 4** in Example 2. This example only differs on how you define which data to use. Sorting and grouping are done exactly in the same way as in Example 2.

Adding a Collector function in a Mapping

A Collector function behaves like a UVJ function with regard to input parameters. Input document elements and attributes, preceding functions' output parameters, and global variables and constants can be linked to collector function. Collector function stores your altered data into memory, so it can be sorted before the outmost restructure loop starts.

This list outlines the differences of a Collector function from a UVJ function:

- A collector function can only be created in a loop.
- There must be only one Collector function within a loop, usually placed last in the loop's sequence.
- Output parameters can only be linked to a following Restructure widget input parameters.

Note: When an output parameter is dragged and dropped to a Restructure function, an input parameter with the same name is automatically created for the Restructure function. The reference link is dashed.

Use this procedure to add a Collector function in a mapping. Repeat this procedure for nested collector loops, if necessary.

1 Double-click a mapping to open it in the Mapper Editor.

2 In the Core Functions pane, select **Collector**.

   Drag and drop Collector in the space inside a loop to add a Collector function.

   Note: When connecting input elements, you can drop elements only where it is acceptable.

3 Add input parameters to receive input data.

   You can directly use data from the input document, data from preceding functions, or data from global variables or constants. Input parameters always have the data type `String`.

   A corresponding output parameter is automatically created. The icon for Collector output parameters is dashed to indicate *reference parameters*.

4 Link the output parameters to Restructure functions.

   Click and drag an output parameter to the succeeding Restructure function. The created link is dashed to indicate *reference link*. No data is actually read in runtime.

5 Save the function in the Mapping Editor.

Collector concepts

When you retrieve data from the input document into the restructuring process as it is, you cannot alter the data or apply any kind of logic on it before it is restructured. After the restructure process, you can alter the data.

Sometimes, you need to alter the data before it goes into the restructure process. The only way to do this without any limitations is to use ordinary mapping logic to collect the data to be restructured. You can use these examples to restructure the data to create a new virtual element that you can use for tasks such as sorting and grouping, converting amounts, and filtering out data.

For example:

- Use substring values.
- Use correct IDs from BODs depending on attribute values.
- Use several different input elements, such as elements within a choice.

When you have all data in the format that you want, link the data to the Collector function just as you would link the data to a **UVJ** function. The purpose of the Collector function is to store your altered data in memory so that it can be restructured.

The Collector function can only exist in a loop, because you need to have a set of records to be able to sort them. If you have a nested data structure, for example, a set of employees each having a set of time transactions, you must use nested collector loops. For the Restructure function, the Collector function is also aware of its context through the nested loops. For example, the Collector function for the time transactions knows to which employee the time transactions belongs, through the Collector function for the employee in the outer loop.

When linking elements and attributes from the input document directly to the Restructure functions, an invisible automatic mapping is executed just before the outmost restructure loop begins to execute. The collector loops replace this invisible automatic mapping. In both cases, the purpose is to collect data to restructure.

When using Collector functions, you must link output parameters for the Collector functions to the Restructure functions. These links are dashed because these are not ordinary data links. If you want to say, "*I want to use this data*". For example, it is preferable to say, "*I want to use /EmployeeCollectLoop/TimeCollectLoop/Collector_1/Date from the collected data*". The rest of the restructure logic is the same as when using data directly from the input document.

You cannot mix data from the input document and data from Collector functions. You must choose one of these two ways of retrieving data for all Restructure functions within a nested restructure loop structure. All data to be restructured must belong to the same data structure, otherwise there is no common context. You can link data from the input document to the Collector functions, and you can also use Collector functions with other data, for example, data retrieved through M3 API calls or database operations. In summary, there is no common context between Collector functions and the input document.

When using Collector functions, you cannot add input control links to Restructure function headers. Input control links are only used for the automatic mapping from the input document. You have already defined the collector loops that fulfill the same purpose.

After data is collected through the Collector functions, the data is sorted before the (outmost) restructure loop starts. Then, the restructure logic is the same as described in [Restructure function](#) on page 247. The restructure logic only differs on how you get the data to restructure.

Collector limitations

The same limitations regarding parallel repeating input elements, as with the Restructure widget, apply for Collector loops. In the *Timesheets* example, you cannot, for example, add a repeating element for employee address and link the element to a loop in parallel with the *TimeCollectLoop*.

For example, if you use two "if" statements in the `TimeCollectLoop` loop to populate the parameters *Date, Project*, and *Hours* differently, you cannot put these collectors in these if statements:

- *Collector_1* in the first if statement
- *Collector_2* in the second if statement.

  Note: *Collector_2* has the same parameters as *Collector_1*.

This example will give you two Collector functions containing the same parameters, and you cannot link from both Collector functions to the Restructure functions.

If statements must be before the Collector function, preparing data for *Date*, *Project*, and *Hours* in variables. Then, after the if statements, and on the same structural level, place the Collector function using the variables as input.

The same restriction regarding loop levels, as with the Restructure widget, applies for Collector loops. In the given example, you cannot have an *Employee* without *Time* elements. If this happens, the innermost Collector widget, *Collector_1*, will not be called for that *Employee*, and the *Employee* will not be saved.

Flat File Definition tool user tasks

The Flat File Definition (FFD) tool is a set of design time tools within Business Document Mapper, that create flat file definition according to the specifications in a flat file message. FFD is a component within the Infor Enterprise Collaborator (IEC) product.

FFD tool supports flat file messaging in IEC. IEC uses FFD tool to process a message and transform a flat file to XML or an XML to a flat file.

Initially, flat file messaging supports conversion of bank messages using banking plug-ins. Later, this feature was extended to become the flat file parser that manages a larger range of flat files in runtime. Starting from IEC release version 2.0, flat file messaging included design time tooling and simplified administration, and configuration of flat file messaging.

Users of Flat File Definition tool

These topics provide you with a general understanding about all the aspects of the creation, management, and how Flat File Definition tool works in order to process flat file messages. This includes an overview of flat files, flat file messaging support in Infor Enterprise Collaborator (IEC), flat file definitions and Flat File Parser.

Appendix A describes the details of a flat file definition created manually using an XML editor of the user's choice.

Appendix B contains examples and exercises about the available features of the Flat File Parser.

These topics does not provide details on how to create and maintain partner agreements to enable IEC Server to detect and process flat messages in run time. Also, these topics do not provide information about particular flat file formats, such as comma separated values (CSV).

The users of the Infor Enterprise Collaborator (IEC) Flat File Definition Tool are application engineers and business consultants who create, edit or validate flat file definitions for their flat file messages.

The users of Partner Administration Tool can also refer to these topics for flat file definitions and similar topics. These users manage the partner agreement information models and the IEC server to send and receive messages between users and their partners. Also, these users can have multiple messaging partners or multiple installations of IEC on a single server.

Prerequisite Knowledge

Users of Flat File Definition Tool must be knowledgeable in these topics:

- XML and XML schemas (XSD)
- Flat files and flat file messaging

Flat File Definition tool overview

The Flat File Definition (FFD) tool is a set of design time tools that creates flat file definition according to the specifications in a flat file message. FFD is a component within the Infor Enterprise Collaborator (IEC) product.

FFD supports flat file messaging in IEC. IEC uses FFD to process a message and transform a flat file to XML or an XML to a flat file.

Initially, flat file messaging supports conversion of bank messages using banking plug-ins. Later, this feature was extended to become the flat file parser that manages a larger range of flat files in runtime. Starting from IEC release version 2.0, flat file messaging included design time tooling and simplified administration, and configuration of flat file messaging.

Flat file processing in runtime

Here is an overview of processing flat file in runtime:

Step	Process	Details
Step 1	Detect	A message received is identified and detected. You define the detection rules in Partner Administration tool.
Step 2	Process	Detected messages are processed based on process steps. You define the process steps in Partner Administration tool.
		For inbound flat file messages, here are the typical process steps:
		• **Convert the flat file to XML**. IEC uses the flat file administration to convert flat file messages to XML.
		• **Apply a mapping**. You define in a mapping how to retrieve or store data in M3. You define a mapping in the Business Document Mapper tool.

Flat file definition in runtime:
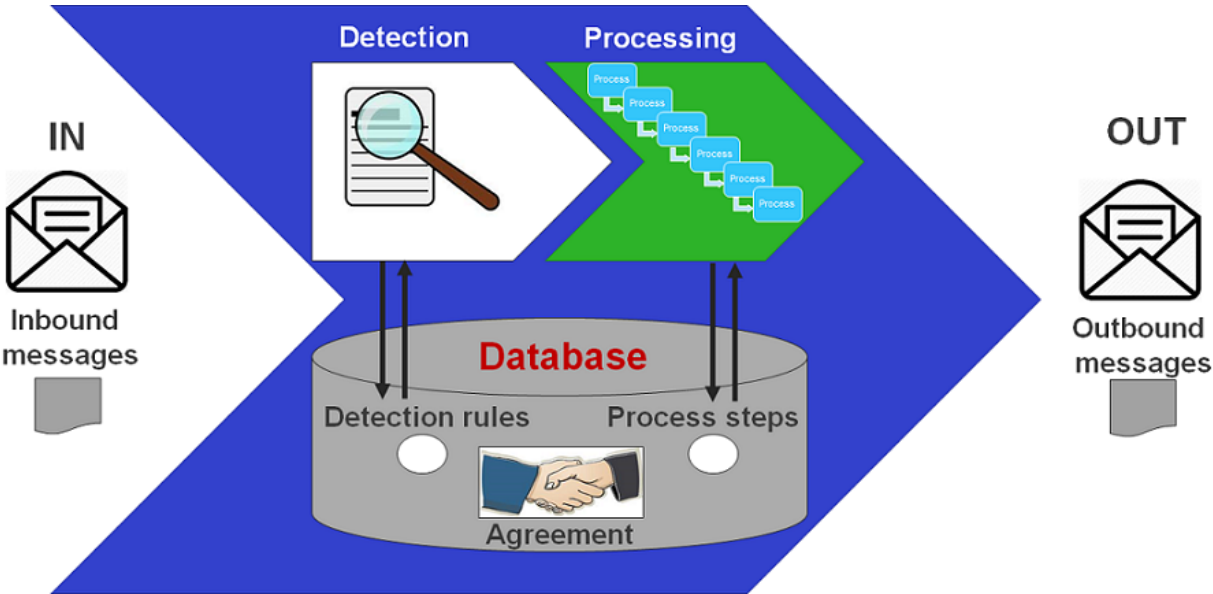
Flat file parser
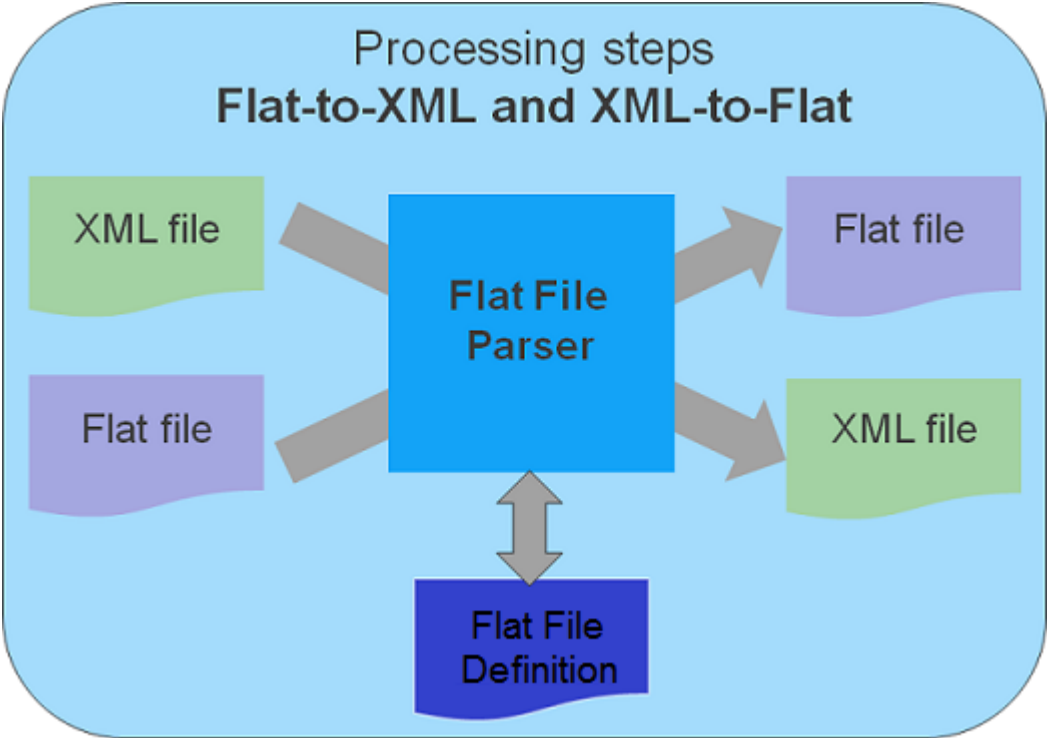
Note:

In IEC versions release earlier than 2.0, only XML files are detected by IEC and the conversion of Flat-to-XML is a required process step.

Starting from IEC version release 2.0 to the latest, the Flat and XML files are detected and managed in the same way. Although, if you will apply a mapping, the transformation to XML is still necessary.

Flat file processing in design time

Flat files are detected and processed in design time. The format of an inbound or outbound flat file message is defined ahead of time. You cannot modify a formatted flat file message.

Here is an overview of processing flat file in design time:

Task	Details
1. Analyze the requirements	Ensure that you have all required documents for the flat file message and all sample flat files.
2. Create a flat file definition	You create a flat file definition based on the flat file message specification. A flat file definition is the basis in transforming a flat file to XML or XML to flat file.
3. Apply Mapping	If you need to apply a Mapping on the resulting transformation, you must generate the XML schema for the corresponding XML message from the Flat File Describer tool. Then, create a mapping to use.
4. Create a Mapping	You create a mapping and then publish the mapping in Business Document Mapper tool.
5. Publish the flat file definition	You publish the flat file definition in the Flat File Repository Manager. Do this to be able to set up the transformation processing step in the Partner Administration tool.
6. Publish the mapping	You use the published mapping when you set up the process steps in the Partner Administration Tool. You publish the mapping in the Business Document Mapper tool.
7. Define an agreement	The agreement includes rules for detecting and processing flat file messages. You define agreements in the Partner Administration tool.
8. Generate sample data	You use sample data as raw data input to create test data. You generate sample data in the Flat File Describer component of the Flat File Definition tool.

Managing a flat file message in design time
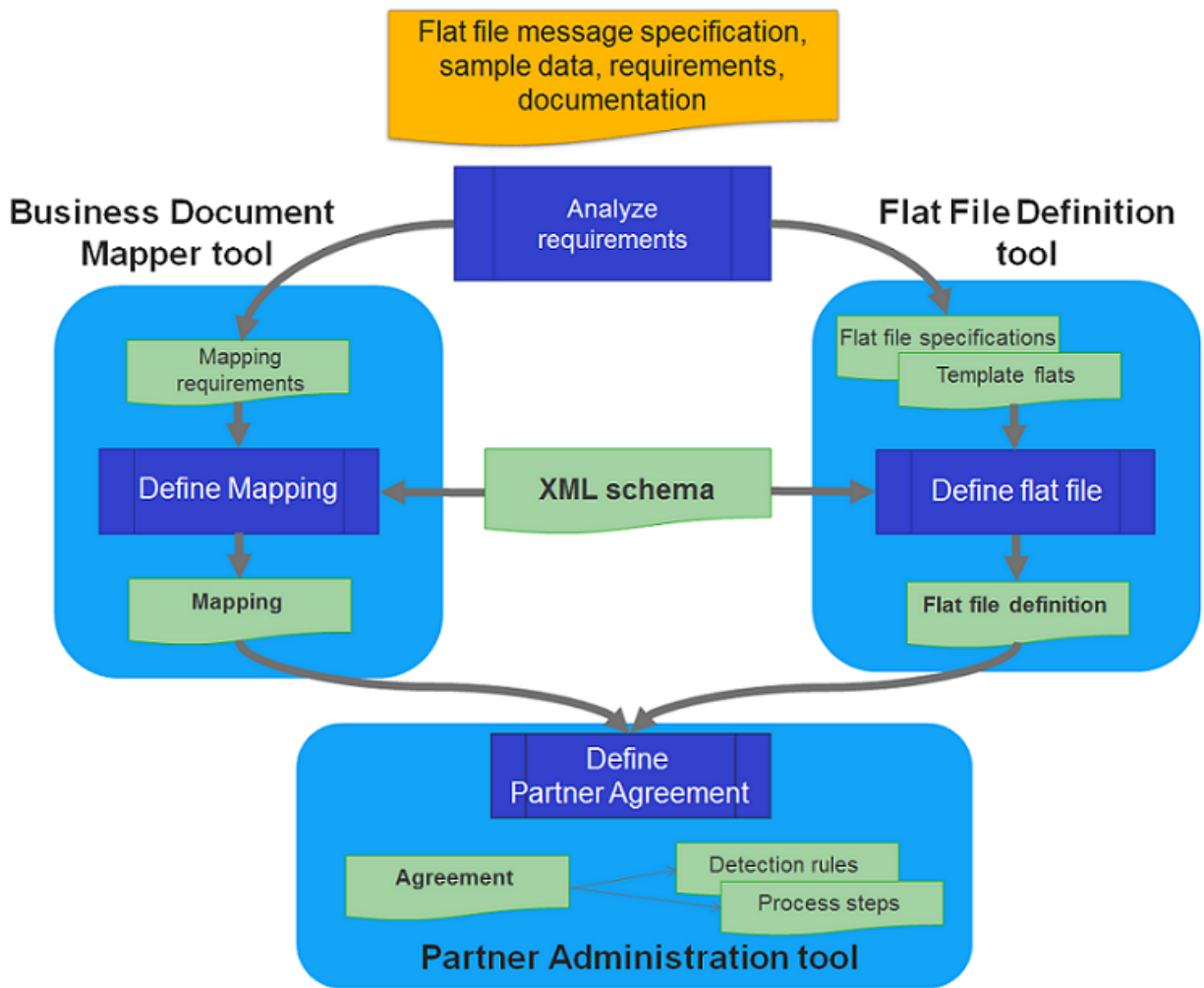
Detection rules are not dependent on uploaded flat file definition. Detection rules are set up separately from flat file definition. Flat file definition is used to transform the flat file definition from or to XML.

Note: The Flat File Definition Tool is now incorporated in the Business Document Mapper tool.

Flat file overview

A flat file is a data file that contains plain text format. It is typically a standalone listing of data.

A flat file is structured to enable a machine to parse it. A flat file does not contain details about how the information is structured, nor does it relate to or contain any links to other files. Flat files can be related, but only if the applications sending and receiving the files are programmed to be aware of such relationships. Relations to other files are described in other tools that are managed in a relation database or in a file manager. An example data relationship is describing the customers for an order or the vendors of the purchases.

Flat files can be used to store and transport all types of application data in the form of flat messages. Often, flat files are used for exchanging messages between two nodes within a network, for example, to exchange purchase orders, invoices, and payments between a supplier and a vendor.

Data in flat files remain in their original form until the data are transmitted in a form of flat file messages. Flat file messages are electronic business messages in flat file format. When transmission is completed, the data is modified and saved in a different format, for example from a flat file to an XML file.

Flat and XML file formats

Flat and XML files are two of the frequently used data file formats. These file formats are both formally structured. However, XML files contain constructs that file files do not have.

- A **flat file** can be constructed in different ways. Each flat file describes a message that has a corresponding XML schema to match its contents. But there is no corresponding flat file structure for every XML schema.

  Each flat file can be stored in XML format.

- An **XML file** is created using custom-defined tags according to XML Document Type Definition (DTD) standards. XML files contain constructs that flat files do not have. An XML is a standard while a flat file can be constructed in different ways.

  An XML file cannot be stored in flat file format.

Another important distinction between XML and flat files is that flat files do not contain metadata or information about the data itself. A receiver of a flat file must use a flat file definition to determine how data is organized in a flat file. The definition can include properties such as length of each item, data type such as integer, float, string, or Boolean, and relationships to other data items within the file.

Flat file message structures

Records or group of records contained in flat files define the structure of a message. Every record is specified in a single line and consists of a special sequence of fields. The sequence of fields in a record is defined as having a number of occurrences.

The first row in a flat file refers to the field name. Fields are separated by comma, records are separated with a line feed character (UNIX style), or a combination of line feed and carriage return character (Windows style). Flat files that include comma as delimiters help ensure fixed-width data formatting, also called comma separated value (CSV) file.

Example 1, six records separated by a line break

The fields in each record are separated by a comma. The third and fourth records are two occurrences of the same type of record which are structured in exactly the same way.

```
DHDR,JSA EENMI567,Blueberryhills Technologies;222-555-1234EEN
MAM98329A0176643002200011796V4000230,DAVID JONES CO;C011000209485456-145RM
```

```
 IV23456,41796RM IV2345,-30000CM,CREDIT MEMO ADJUSTMEN
TA042557710000110316V40000004,SOFTBANK
```

Example 2, a record that uses extra formatting to avoid separator confusion

If you want to use a separator without its special meaning, use an escape sequence consisting of a string of characters. When it precedes a separator, an escape sequence has the effect of negating the separator's special meaning. The escape sequence forward slash "**/**" is used to parse fields.

```
This field has a comma/,that needs to be escaped,this is another field,
```

Example 3, a record that uses an identifier

The first field in the record is defined as an identifier which allows the enclosing records' signature to separate from other records in the message.

```
Order,39827483927,03982409238,02384902
```

Flat file definition structure

- The flat file message structure has a top-level Message component called "DTAZVBankTransfer".
- The fields in the flat file are structured as separate **Fields** in the structure under a **Record** component.
- XML tags in the corresponding XML file are also related to the records and fields of the flat file structure and the XML file.
- You can use and add a **Group** component to define a nested structure of records and fields in the flat file structure and the XML file.

Flat file definition metadata

Metadata in a flat file is captured in a flat file definition. Some metadata are mandatory and some are optional. You define mandatory metadata by using a message default or by setting the metadata on the group, record, or field level. For example, the mandatory metadata for a field are: Name, Tag Name, Data Type, and if applicable, Start Position.

For more information, see the topics "Flat File Definition format" and "Changing the Message Component Property Values" in this user guide.

These tables show a list of metadata and description for each component in a flat file definition.

File

This table shows File component metadata.

Metadata	Details
ID	Identifier for a message transformation definition
Messages	Allows only one message per file

Message Definition

This table shows Message Definition component metadata.

Metadata	Details
Name	Specified for documentation purposes
Tag name	Specify a tag identified in XML or a tag written to XML
Escape character	Escape sequence for field and record separators
Record separators	Common record separators are: Carriage return: `\r` or `newline` and Line feed: `\n` or `lf`
Fixed length separator	If set to true, it defines fixed lengths of records
Field separator fields	Specify a value. There can only be one separator in a message. A record separator is considered as a valid separator. This value overrides any field length definition.
Ignore preceding field separator	Select to ignore preceding field separators
Default alignment	Specify the alignment to use when a field does not have a defined alignment. You can use a numeric and alpha field value.
Default padding	Specify the padding character to use when a field does not have a defined padding character. You can use numeric and alpha field value.
Default trim	Specify a trim behavior for XML to record if no trim is defined for a field. Use this to remove a preceding and trailing white space or keep the data unchanged. You can set this for a numeric and alpha field values.
Default empty field value	Specify a value to use if the data is empty for a field or tag. If left blanks, empty fields will contain empty tags.
Add first field separator	Specify an extra field separator which is defined on the message level in front of the first field of every record in the message
Add last field separator	Specify to add extra file separators after the last field of every record in the message

Group

This table shows Group component metadata.

Metadata	Details
Name	Used for documentation purposes
Tag name	Tag to identify or write
Minimum occurrences	Minimum required number of times this group must occur
Maximum occurrences	Maximum number of times this group is allowed to occur. Specify a 0 (null) value for unlimited occurrences.

Record

This table shows Record component metadata.

Metadata	Details
Maximum occurrences	Specify the allowed maximum number of times for this group to occur. Specify a 0 (null) value for unlimited occurrences.
Name	Used for documentation purposes
Tag name	Select a tag to identify or a tag to write
Minimum occurrences	Specify the minimum required number of times this group must occur
Maximum occurrences	Specify the maximum number of times this group is allowed to occur. Specify a 0 (null) value for unlimited occurrences.

Field

This table shows Field component metadata.

Metadata	Details
Name	Used for documentation purposes
Tag name	Tag to identify or write
Data type	Specify a data type value to differentiate alignment, padding, trimming, and default empty value handling. You can use a numeric and alpha field values. The default value type is alpha.
Padding character	Character to pad this field
Alignment	Alignment to use for this field
Empty field value	Value to use if this field is empty
Trim	Indicates whether to trim the field
Identifier	Defines whether this field is an identifier
Default value	Identifier value that will be tested against, or written to.

Metadata	Details
Start position	Field starting position if length-defined fields (no field separator) are defined
Length	If the field length is shorter than this length, the field will be padded.
	If the length shall be used as parsing fields without a field separator, then the last field may be defined as "-1" , which will allow the field to have a variable length.

Flat file parser concepts

Flat file parser is a module used to break up a document into separate components. Parsed documents are attached to tags that define each component, including the component's function and format. Flat file parser is the component in EC that performs the actual transformation between flat and XML files using a defined record structure for the message. The detailed information about how the data is organized within the file and any relationships to other data items within the file is contained in a flat file definition.

In addition to the message structure, fields are managed in flat file parser based on your specified values in these metadata.

Metadata	Description
Padding	A field length is fixed; otherwise, you need a padding to have the same length for the input and the output field. For example, in a field with fixed seven digits where only three digits are used, you need a padding to fill up the four digits gap.
Alignment	Location of the padding characters' placement relative to the padded information.
White Space Trim	Removal of all formatting directives surrounding a string of characters, such as tabs, line feeds, and blanks.
Empty Value	Define a value to use on a field with empty value. This is signaled using a special string of information that is inserted by the parser.
Record Separators	Commonly used characters as standard end of line separators.

This table shoes a list of commonly used characters as end of line record separators:

Platform	Description	In Java	In EC	Hex values
PC	[return][linefeed]	[\r][\n]	[\r][\n]	[0D][0A]
UNIX	[line feed]	[\n]	[\n]	[0A]
Mac	[return]	[\r]	[\r]	[0D]

You specify the flat file message encoding in the Partner Administration tool. The encoding is not part of the flat file definition. You can use the same flat file definition for flat files that have the same structure but are encoded differently.

Note:

When you import to, or export from, the Flat File Repository, the flat file definitions are always encoded in Java Encoding (UTF-16).

Managing Flat File Definition tool

This section describes how you can manage the Flat File Definition tool using Business Document Mapper.

Creating a flat file definition

You can create file file definitions using the Flat File Definition tool in Business Document Mapper. Follow these steps to create flat file definitions.

1   Right-click a mapping project in the Mapping Explorer view and select New Flat File Definition.
    The Flat File Definition Editor is displayed.

2   Create the flat file definition.
    For a list of example flat file definitions, see the appendices section in this user guide.

3   Click the Save icon or press Ctrl + S to save the flat file definition.

4   Right-click in the Flat File Definition Editor's definition tree and select **Save to Server** to save the flat file definition to the appropriate server.

Adding a new record

A message can contain any number of groups and records. You can add a new record in an existing flat file definition by following these steps.

Note: Top level and sibling records, can not be made into child records of a group. Child records can not be promoted to a higher level record. You can move a child record from one group to another group.

1   Right-click on the Message or Group component in the Flat File Definition pane and click **New Child > Record**.

2   Verify that the new record is added correctly.

Adding a new group

A message can contain any number of groups and records. You can add new groups in an existing flat file definition by following these steps.

Note: Groups on the top level and sibling groups, can not be made into child groups. Child groups cannot be promoted to a higher level group.

1   Right-click on the Message component in the Flat File Definition pane and click **New Child > Group**.
2   Verify that the new group is added correctly.

Adding a field to an existing record

A message can contain any number of fields. You can a add new field in an existing flat file definition by selecting any of these options:

1   Right-click on Record component and select **New Child > Field**. The child field is positioned after an existing field.
2   Right-click on Field component and select **New Sibling > Field**. The sibling field is positioned after an existing field.

Importing a flat file definition

Before you can use a flat file definition, you must import the flat file definition from the server.

Note: Importing a flat file definition from the server or a classic file, saves the XML file in your workspace. Renaming the flat file definition also renames the XML file in your workspace.

1   In the Mapping Explorer, right click a mapping project and click **Import > Flat File Definition from Server**.

    The Import Flat File Definition wizard is displayed.

2   Select the appropriate server then click **Next**.
3   Select the flat file definition to import.
4   Click **Finish**.
    Flat File Definition node is displayed in the Mapping Explorer. Double-click to open the flat file definition in the Flat File Definition Editor. Right-click the Flat File Definition node and click Open Flat File Definition can also open the flat file definition.

Exporting a flat file definition

Use this procedure to export a copy of a flat file definition from the server to the local disk. Exported copies can be imported into another server or use as a draft to modify the existing flat file definition.

1   In the Mapping Explorer view pane, right-click the flat file definition you want to export and select Export.
2   Click **Browse** to select the local file system to store the exported flat file definition.
3   Click **Finish**.

Deleting a flat file definition

Use this procedure to delete a flat file definition from the repository or from a local disk.

Note:

Delete has no undo function. Verify the name of a flat file definition before you invoke delete. You cannot delete a flat file definition that is in use. When you invoke delete, the flat file definition is physically removed from the database or from the Mapping Explorer view.

1 Open the flat file definition in the Flat File Definition Editor, to delete from repository.
2 Right-click on the definition tree and select **Remove from Server**. Closed the Flat File Definition Editor.
3 In the Mapping Explorer view pane, right-click the flat file definition you want to delete, select **Delete**.
4 Verify that the deleted definition is removed from the Mapping Explorer view pane.

Editing a flat file definition

You can update the imported flat file definition both in your local workspace and server. Do this by opening the flat file definition in the Flat File Definition Editor.

1 Right-click the flat file definition in the Mapping Explorer view pane.
   You can also double-click the flat file definition in the Mapping Explorer view pane.
2 Select Open Flat File Definition.
   The Flat File Definition Editor is displayed.
3 Edit the flat file definition metadata.

Editing a definition in the repository

Follow these steps to edit definitions using Flat File Definition tool.

1 Right-click the flat file definition in the Mapping Explorer view pane.
   You can also double-click the flat file definition in the Mapping Explorer view pane
2 Select **Open Flat File Definition**.
   The Flat File Definition Editor is displayed.
3 Make any changes to the flat file definition metadata.
4 Click the Save button or press Ctrl + s to save the flat file definition modified in the editor.
5 Right-click in the definition tree area, and click **Save to Server**.
6 Select the appropriate server.
7 Click **Finish**.

Activating a flat file definition

You can activate a flat file definition by selecting any of the following options:

1   Right-click a flat file definition in the Mapping Explorer, and select **Activate**.

2   Open a flat file definition in the Flat File Definition Editor, right-click on the definition tree area, and select **Activate**.

Deactivating a flat file definition

You can deactivate a flat file definition by selecting any of the following options:

1   Right-click a flat file definition in the Mapping Explorer, and select **Deactivate**.

2   Open a flat file definition in the Flat File Definition Editor, right-click on the definition tree area, and select **Deactivate**.

Activating multiple flat file definitions

You can activate multiple flat file definitions simultaneously by following these steps.

1   Select flat file definitions in the Mapping Explorer view by holding Shift and left-clicking the flat file definitions.

2   Right-click of the selected flat file definitions.

3   Click **Activate...**.

Deactivating multiple flat file definitions

You can deactivate multiple flat file definitions simultaneously by following these steps.

1   Select flat file definitions in the Mapping Explorer view by holding Shift and left-clicking the flat file definitions.

2   Right-click of the selected flat file definitions.

3   Click **Deactivate...**.

Flat File Definition Editor overview

The Flat File Definition Editor is an editor packaged in Business Document Mapper. It is an editor that helps you create, verify, and validate flat file definitions. It can also be used to define flat files or as a complement tool to an XML editor of your choice. You can also generate sample files in XML, XSD, and flat file formats using this tool.

Flat File Definition Editor user interface

The user interface of the Flat File Definition Editor is composed of several tabs and views.

In addition to these views tabs and complementary views, Properties and Flat File Definition Console are also available.

Here is a list of Flat File Definition editor user interface components.

Definition Tab

This tab contains Message Definition Tree and Flat File Template Text Editor.

Message Definition Tree View

This tree view is part of the Definition tab and shows the message structure of a flat file. You can define the structure of a flat file using components such as Message, Group, and Record and Field.

Flat File Template editor

This text editor is part of Definition tab shows the structure of a template flat file from which you can create a flat file definition. You can load a template flat file from local file system using the Load button located at the upper right of the Flat File Definition Editor.

Sample Flat File + XML Tab

This tab contains the Sample Flat File view and Sample XML view.

Sample Flat File view

This view is part of Sample Flat + XML tab and shows a preview of the output from an XML-to-flat transformation. This is auto-generated when you click on the tab. Changes made in the Message Definition tree view is reflected in this view.

Sample XML view

This view is part of Sample Flat + XML tab and shows a preview of the output from a flat-to-XML transformation. This is auto-generated as changes in the Message Definition tree view is reflected in this view. You can also save the sample XML file to disk.

XML Schema view

This view shows a preview the XML schema that defines the format of the sample XML. You can generate a schema from the Tools menu. You can save the sample XSD to disk and use it when you define a mapping.

Properties view

This view shows the properties of components selected in the tree view. You can set a value for each property but some properties can have a default value defined on the message level. Some properties are mandatory and some are optional.

Flat File Definition Console view

This view shows import, save, delete, update, activate and deactivate flat file definition server operation information.

Uses of Flat File Definition Editor

The Flat File Definition Editor enables you to create Flat File Definitions. These are the tasks you can perform using the Flat File Definition Editor.

Defining flat file structures in message definition view

This section describes how you can define flat file structures in the Message Definition View.

These topics describe the tasks that you can perform.

Changing the message component property values

Use this procedure to change the message component property values.

1   In the Message Definition view, select the component to edit.
    The Property view shows the component properties.

2   In the Properties view, select the value to edit and click the Value field link.
    Current value is highlighted.

3   In the value field, specify a new value.
    For properties with predefined values, select a new value from a drop-down list.

4   Press Enter to confirm the change.

Adding a group to a message

A message can contain any number of groups and records. Use this procedure to add a group to a message.

1   In the Message Definition view, right-click a Message and select:
    New Child > Group to add a group. The group is positioned after an existing group or record.

2   Verify that the new group is added correctly.

Adding a record to a message

A message can contain any number of groups and a record. Use this procedure to add a record to a message.

1   In the Message Definition view, right-click a Message and select **New Child > Record** to add a record. The record is positioned after an existing group or record.

2   Verify that the new record is added correctly.

Adding a record to a group

A group can contain any number of groups and records. Use this procedure to add a record to a group.

1   In the Message Definition view, right-click a Group component and select:

   New Child > Record to add a new record. The record is positioned after an existing record.

2   Verify that the new record is added correctly.

Adding a field to a record

Use this procedure to add a field to a record.

1   In the Message Definition view, expand a Record component and right-click the Fields component and select New Child > Field to add a new field to a record or a second field. The child field is positioned after an existing field.

2   Verify that the new field is added correctly.

Defining template flat files

This section describes how you can define template flat files in the Template Flat View.

These are the tasks you can perform:

Opening and preparing a template flat file

A template flat file helps you create a flat file definition. Use this procedure to open and prepare a template flat file.

1   In the Flat File Definition Editor, click the **Load** button beside Flat File Template text editor.

2   Select an existing template flat file and click **Open**.

   The template file displays in the Template Flat view.

3   If the template flat file is based on a sample flat file, use these guidelines to prepare the template file:

   Remove all duplicate records.

   Add optional records to enable the records to be defined.

4   The loaded template file is automatically saved in your workspace.

   Caution:

> Editing parts of the template flat file that have already been defined might cause the flat file definition to become inconsistent with the template flat file.

Validating a flat file definition

Validating a flat file definition ensures that your definition is syntactically and semantically correct. The results should be considered as an indication of the validity of a definition. Validation in Flat File Definition covers these tasks:

Check the structure of the flat file definition. This is useful when the flat file definition is created manually using an XML editor and then imported into the tool.

Note:

You must validate that a flat file definition is correct according to the specification of the Flat File Message.

Use this procedure to validate the syntax and semantics of a flat file definition.

1   Open an existing template flat file or a flat file definition.

2   In the Flat File Definition Editor, right-click in the Message Definition tree view, select **Validate**.

The Validation Information dialog displays the result of the validation process.

Sample data generation

The Flat File Definition Editor can be used to generate sample data.

Here are the topics included:

Generating sample data

Use this procedure to generate sample data in XML, XSD, and flat format. These sample data are based on the message description that you have created.

1   Open an existing flat file definition.

2   In the Flat File Definition Editor, here are the available options:

- Select Sample Flat File + XML tab to view the generated sample Flat File data in the Flat File view.
- Select Sample Flat File + XML tab to view the generated sample XML data in the XML view.
- Select XML Schema tab to view the generated schema data.

Sample flat file

A flat file is a text document that contain records without a structured relationship. In **Sample Flat File + XML** tab the view shows the contents of your selected sample flat file.

Note: Some properties in your flat file definition structure will affect the generated sample flat file.

Use these guidelines:

- If a field has a specified length, it will be filled with data in the sample.
- If a field does not have a length, it will get a data and a separator.
- A record separator is always written after a record.
- Each occurrence of a sample record will be written three times or, with the minimum number of occurrences you specified.

Sample XML

XML contains user-defined tags to describe data. In **Sample XML + Flat** tab the view shows the contents of your selected sample XML.

Note: In Flat File Definition Editor, some properties in your flat file definition structure will affect the generated sample XML file.

Use these guidelines:

- Every unlimited occurrence will be written three times, or you can specify a minimum number of occurrences.
- You can use alpha or numeric field data values.

Sample XML schema

XSD is a file written in XML that contains the descriptions or metadata file of a particular xml file. XSD file defines which elements and attributes are allowed and in what order of execution.

The XML Schema tab view shows the contents of your selected sample XSD file.

Note: In Flat File Definition Editor, some properties in your flat file definition structure will affect the generated sample schema.

Appendix A: Flat File Definition format

This section details example flat file administration formats that you can refer to when manually creating flat file definitions using an XML editor. The remainder of this section provides the XML schema for flat file definitions.

Example of a Flat File Definition in XML

This example shows a classic flat file definition created manually using an XML editor.

Note: Use Import > Flat File Definition from Classic File (.xml) to import this file to Business Document Mapper.

```
<BankRepository>
  <Files>
    <File>
      <ID>flat_test_1</ID>
      <Messages>
       <Message>
        <Name>TopNode</Name>
         <TagName>TopNode</TagName>
         <DefaultAlignment>
            <Numeric>RIGHT</Numeric>
            <alpha>LEFT</alpha>
         </DefaultAlignment>
         <DefaultPadding>
            <Numeric>0</Numeric>
            <alpha></alpha>
         </DefaultPadding>
         <DefaultTrim>
            <Numeric>1<Numeric>
            <alpha/>
         <DefaultEmptyFieldValue>
            <Numeric>0</Numeric>
            <alpha/>
         </DefaultEmptyFieldValue>
        <EscapeChar/>
        <RecordSeparator>newline</RecordSeparator>
        <FieldSeparator>;</FieldSeparator>
        <IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSeparator>
        <Records>
         <Group>
         <Name>Person</Name>
         <TagName>Person</TagName>
         <MinOccur<1</MinOccur>
         <MaxOccur>0</MaxOccur>
         <Records>
          <Record>
           <Name>PersonDetails</Name>
           <TagName>PersonDetails</TagName>
           <MinOccur>1</MinOccur>
           <MaxOccur>1</MaxOccur>
           <Fields>
            <Field>
              <Name>FirstName</Name>
              <TagName>FirstName</TagName>
              <DataType>Alpha</DataType>
              <PaddingChar/>
              <DecimalLength/>
              <Alignment>LEFT</Alignment>
```

```
            <EmptyFieldValue></EmptyFieldValue>
            <Trim>0</Trim>
            <StartPos>0</StartPos>
            <Length>10</Length>
            <DefaultValue>/DefaultValue>
            <Identifier>0</Identifier>
          </Field>
        </Fields>
        </Record>
        </Records>
      </Group>
      </Records>
    </Message>
  </Messages>
  </File>
  </Files>
</BankRepository>
```

Elements and types

This section describes in detail the elements and complex types used in example flat file definition in XML.

Elements in Flat File Definition

This table lists the elements used in the example flat file definition in XML including the details such as Type, Properties, Children, Usage, and Source.

Element	Type	Properties	Children	Used by	Source

Alpha	xs:string	content = simple		DefaultEmp-tyFieldValue-TypeDefault-Padding-Type De-faultTrim-Type	`<xs:element name="Alpha"`
					
					`type="xs:string"/>`

BankReposi-tory		content =complex	Files		`<xs:element`
					`name="BankRepository">`
					`  <xs:complexType>`
					`   <xs:sequence>`
					`   <xs:element`
					`name="Files"`
					
					`type="FilesType"/>`
					`   </xs:sequence>`
					`  </xs:complexType>`
					`</xs:element>`

BankReposi-to-ry/Files	FilesType	content =complexis-Ref = 0	File	FieldType	`<xs:element name="Files"`
					
					`type="FilesType"/>`

DataType	restric-tionofxs:string	content =simple			`<xs:element`
					`name="DataType">`
					`  <xs:simpleType>`
					`    <xs:restriction`
					`base="xs:string">`
					`     <xs:enumeration`
					`value="Alpha"/>`
					`     <xs:enumeration`
					`value="Numeric"/>`
					`    </xs:restriction>`
					`  </xs:simpleType>`
					`</xs:element>`

ID	xs:string	content =simple		FileType	`<xs:element name="ID"` `  type="xs:string"/>`

MaxOc-cur	xs:string	content =simple
		Note: 0,U, unbounded signifies unlimited oc-curences
				GroupType-RecordType	`<xs:element name="MaxOc`
					`cur"`
					`  type="xs:string"/>`

Element	Type	Properties	Children	Used by	Source
MinOccur	xs:integer	content =simple		GroupType-RecordType	`<xs:element name="MinOccur" type="xs:integer"/>`
Name	xs:string	content =simple		FieldType GroupType-MessageType RecordType	`<xs:element name="Name" type="xs:string"/>`
Numeric	xs:integer	content =simple		DefaultEmptyFieldValue-TypeDefault-Padding-Type DefaultTrim-Type	`<xs:element name="Numeric" type="xs:integer"/>`
Tag-Name	xs:string	content =simple		FieldType GroupType-MessageType RecordType	`<xs:element name="TagName" type="xs:string"/>`
DefaultAlignment-Type/Numeric	restriction of xs:string	content =simpleisRef = 0		DefaultEmptyFieldValue-TypeDefault-Padding-TypeDefault-TrimType	`<xs:element name="Numeric" minOccurs="0"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="RIGHT"/> <xs:enumeration value="LEFT"/> </xs:restriction> </xs:simpleType> </xs:element>`
DefaultAlignment-Type/Alpha	restriction of xs:string	content =simpleisRef = 0		DefaultEmptyFieldValue-TypeDefault-Padding-TypeDefault-TrimType	`<xs:element name="Alpha" minOccurs="0"> <xs:simpleType> <xs:restriction base="xs:string"> <xs:enumeration value="RIGHT"/> <xs:enumeration value="LEFT"/> </xs:restriction> </xs:simpleType> </xs:element>`

Element	Type	Properties	Children	Used by	Source
FieldsType/Field	FieldType	content =complexisRef = 0	Name, TagName, DataType, PaddingChar, Length, StartPos, Alignment, DefaultValue, Trim, EmptyFieldValue, Identifier		`<xs:element name="Field" type="FieldType" maxOccurs="unbounded"/>`
FieldType/PaddingChar	xs:string	content =simpleisRef = 0			`<xs:element name="PaddingChar" type="xs:string" minOccurs="0"/>`
FieldType/Length	xs:integer	content =simpleisRef = 0			`<xs:element name="Length" type="xs:integer"/`
FieldType/StartPos	xs:integer	content = simpleisRef = 0			`<xs:element name="Start Pos" type="xs:integer"/>`
FieldType/Alignment	restriction ofxs:string	content =simpleisRef = 0			`<xs:element name="Align ment"     minOccurs="0">   <xs:simpleType>     <xs:restriction base="xs:string">       <xs:enumeration value="RIGHT"/>       <xs:enumeration value="LEFT"/>     </xs:restriction>   </xs:simpleType> </xs:element>`
FieldType/DefaultValue	xs:string	content =simpleisRef = 0			`<xs:element name="Default Value" type="xs:string" minOccurs="0"/>`

Element	Type	Properties	Children	Used by	Source
Field-Type/Trim	xs:integer	content =simpleis-Ref = 0			`<xs:element name="Trim" type="xs:integer" minOccurs="0"/>`
Field-Type/Emp-tyField-Value	xs:string	content =simpleis-Ref = 0			`<xs:element name="Empty FieldValue" type="xs:string" minOccurs="0"/>`
Field-Type/Iden-tifier	xs:string	content =simpleis-Ref = 0			`<xs:element name="Identifier" type="xs:string" minOccurs="0"/>`
FilesType/File	FileType	content =complex isRef = 0	ID Messages		`<xs:element name="File" type="FileType"/>`
File-Type/Mes-sages	Mes-sagesType	content =complex isRef = 0	Message		`<xs:element name="Messages" type="MessagesType"/>`
Group-Type/Group	Group-Type	content =complex isRef = 0	Name, Tag-Name, MinOc-cur, Max-Occur, Group, Records		`<xs:element name="Group" type="GroupType" minOccurs="0" maxOccurs="unbounded"/>`
Group-TypeRecords	Record-sType	content =complex isRef = 0	Record		`<xs:element name="Records" type="RecordsType" minOccurs="0"/>`
Mes-sagesTypeMes-sage	exten-sion of Mes-sageType	content =complex isRef = 0			`<xs:element name="Message"> <xs:complexType> <xs:complexContent> <xs:extension base="MessageType"/> </xs:complexContent> </xs:complexType> </xs:element>`

Element	Type	Properties	Children	Used by	Source
			Name, Tag-Name, DefaultAlignment, Default-Padding, Default-Trim, DefaultEmptyField-Value, Es-capeChar, Record-Separator, Field-Separator, RecordsGroup		
Mes-sageTypeDe-faultAlign-ment	DefaultAl-ignment-Type	content =complex isRef = 0	Numer-ic,Alpha		`<xs:element name="Default Alignment"` `  type="DefaultAlignment Type"/>`
Mes-sageTypeDe-fault-Padding	Default-Padding-Type	content =complex isRef = 0	Numer-ic,Alpha		`<xs:element name="Default Padding"` `  type="DefaultPadding Type"/>`
Mes-sageTypeDe-faultTrim	Default-TrimType	content =complex isRef = 0	Numer-ic,Alpha		`<xs:element name="Default Trim"` `  type="DefaultTrimType"/>`
Mes-sageTypeDe-faultEmp-tyField-Value	Default-Empty-FieldVal-ueType	content =complex isRef = 0	Numer-ic,Alpha		`<xs:element name="DefaultEmptyField Value"` `  type="DefaultEmptyField ValueType"/>`
Mes-sageTypeEs-capeChar	xs:string	content =simpleis-Ref = 0			`<xs:element name="Es capeChar"` `  type="xs:string" minOc curs="0"/>`
	xs:string				

Element	Type	Properties	Children	Used by	Source
Message Type Record Separator		content =simpleisRef = 0			`<xs:element name="RecordSeparator" type="xs:string"/>`
Message Type Field Separator	xs:string	content =simpleisRef = 0			`<xs:element name="FieldSeparator" type="xs:string" minOccurs="0"/>`
Message Type Records	RecordsType	content =complexisRef = 0	Record		`<xs:element name="Records" type="RecordsType" minOccurs="0"/>`
Message Type Group	GroupType	content =complexisRef = 0	Name, TagName, MinOccur, MaxOccur, Group, Records		`<xs:element name="Group" type="GroupType" minOccurs="0" maxOccurs="unbounded"/>`
Records Type Record	RecordType	content =complexisRef = 0	Name, TagName, MinOccur, MaxOccur, Fields		`<xs:element name="Record" type="RecordType" maxOccurs="unbounded"/>`
Record Type/Fields	FieldsType	content =complexisRef = 0	Field		`<xs:element name="Fields" type="FieldsType" minOccurs="0"/>`

ComplexTypes in Flat File Definition

This table lists the ComplexType components in the example flat file definition in XML including the details such as ComplexType, Children, Usage, and Source.

Details	Source
ComplexType - `DefaultAlignmentType` Children - `Numeric, Alpha` Used by - `MessageType/DefaultAlignment`	```<xs:complexType name="DefaultAlignmentType">```
ComplexType - `DefaultEmptyFieldValueType` Children - `Numeric, Alpha` Used by - `MessageType/DefaultEmptyFieldValue`	```<xs:complexType name="DefaultEmptyFieldValueType">```
ComplexType - `DefaultPaddingType` Children - `Numeric, Alpha` Used by - `MessageType/DefaultPadding`	```<xs:complexType name="DefaultPaddingType"><xs:sequence><xs:element ref="Numeric" minOccurs="0"/><xs:element ref="Alpha" minOccurs="0"/></xs:sequence></xs:complexType>```
ComplexType - `DefaultTrimType` Children - `Numeric, Alpha` Used by - `MessageType/DefaultTrim`	```<xs:complexType name="DefaultTrimType"><xs:sequence><xs:element ref="Numeric" minOccurs="0"/><xs:element ref="Alpha" minOccurs="0"/></xs:sequence></xs:complexType>```
ComplexType - `FieldsType` Children - `Field` Used by - `RecordType/Fields`	```<xs:complexType name="FieldsType"><xs:sequence><xs:element name="Field" type="FieldType" maxOccurs="unbounded"/></xs:sequence></xs:complexType>```
ComplexType - `FilesType` Children - `File` Used by - `BankRepository/Files`	```<xs:complexType name="FilesType"><xs:sequence><xs:element name="File" type="FileType"/></xs:sequence></xs:complexType>```

Details	Source
ComplexType - `FileType` Children - `ID Messages` Used by - `FilesType/File`	``` <xs:complexType name="FileType"> <xs:sequence> <xs:element ref="ID"/> <xs:element name="Messages" type="MessagesType"/> </xs:sequence> </xs:complexType> ```

ComplexType - `GroupType`
 	
Children - `Name, TagName, MinOccur, MaxOccur, Group, Records`
 	
Used by - `MessageType/Group,GroupType/ Group`
	``
	<xs:complexType name="GroupType">
	  <xs:sequence>
	    <xs:element ref="Name"/>
	    <xs:element ref="TagName"/>
	    <xs:element ref="MinOccur"/>
	    <xs:element ref="MaxOccur"/>
	    <xs:element name="Group"
	type="GroupType"
	        minOccurs="0" maxOc
	curs="unbounded"/>
	    <xs:element name="Records"
	type="RecordsType"
	        minOccurs="0"/>
	  </xs:sequence>
	</xs:complexType>
	``

ComplexType - `MessagesType` Children - `Message` Used by - `FileType/Messages`	``` <xs:complexType name="Mes sagesType"> <xs:sequence> <xs:element name="Message"> <xs:complexType> <xs:complexContent> <xs:extension base="Mes sageType"/> </xs:complexContent> </xs:complexType> </xs:element> </xs:sequence> </xs:complexType> ```
ComplexType - `RecordsType` Children - `Record` Used by - `MessageType/Records, GroupType/Records`	``` <xs:complexType name="Record sType"> <xs:sequence> <xs:element name="Record" type="RecordType" maxOccurs="unbounded"/> </xs:sequence> </xs:complexType> ```

Details	Source
ComplexType - `RecordType` **Children** - `Name, TagName, MinOccur, MaxOccur, Fields` **Used by** - `RecordsType/Record`	```<xs:complexType name="RecordType">

  <xs:sequence>
    <xs:element ref="Name"/>
    <xs:element ref="TagName"/>
    <xs:element ref="MinOccur"/>
    <xs:element ref="MaxOccur"/>
    <xs:element name="Fields"
type="FieldsType"
        minOccurs="0"/>
  </xs:sequence>
</xs:complexType>``` |
| **ComplexType** - `RecordsType`

Children - `Record`

Used by - `MessageType/Records, GroupType/Records` | ```<xs:complexType name="RecordsType">
  <xs:sequence>
    <xs:element name="Record"
type="RecordType"
        maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>``` |
| **ComplexType** - `RecordType`

Children - `Name, TagName, MinOccur, MaxOccur, Fields`

Used by - `RecordsType/Record` | ```<xs:complexType name="RecordType">

  <xs:sequence>
    <xs:element ref="Name"/>
    <xs:element ref="TagName"/>
    <xs:element ref="MinOccur"/>
    <xs:element ref="MaxOccur"/>
    <xs:element name="Fields"
type="FieldsType"
        minOccurs="0"/>
  </xs:sequence>
</xs:complexType>``` |
| **ComplexType** - `RecordsType`

Children - `Record`

Used by - `MessageType/Records, GroupType/Records` | ```<xs:complexType name="RecordsType">
  <xs:sequence>
    <xs:element name="Record"
type="RecordType"
        maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>``` |
| **ComplexType** - `RecordType`

Children - `Name, TagName, MinOccur, MaxOccur, Fields`

Used by - `RecordsType/Record` | |

Details	Source
| | ```xml
<xs:complexType name="RecordType">

  <xs:sequence>
    <xs:element ref="Name"/>
    <xs:element ref="TagName"/>
    <xs:element ref="MinOccur"/>
    <xs:element ref="MaxOccur"/>
    <xs:element name="Fields"
type="FieldsType"
        minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
``` |
| ComplexType - `RecordsType`<br><br>Children - `Record`<br><br>Used by - `MessageType/Records, GroupType/Records` | ```xml
<xs:complexType name="Record
sType">
  <xs:sequence>
    <xs:element name="Record"
type="RecordType"
        maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
``` |
| ComplexType - `RecordType`<br><br>Children - `Name, TagName, MinOccur, MaxOccur, Fields`<br><br>Used by - `RecordsType/Record` | ```xml
<xs:complexType name="RecordType">

  <xs:sequence>
    <xs:element ref="Name"/>
    <xs:element ref="TagName"/>
    <xs:element ref="MinOccur"/>
    <xs:element ref="MaxOccur"/>
    <xs:element name="Fields"
type="FieldsType"
        minOccurs="0"/>
  </xs:sequence>
</xs:complexType>
``` |
| ComplexType - `RecordsType`<br><br>Children - `Record`<br><br>Used by - `MessageType/Records, GroupType/Records` | ```xml
<xs:complexType name="Record
sType">
  <xs:sequence>
    <xs:element name="Record"
type="RecordType"
        maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
``` |
| ComplexType - `RecordsType`

Children - `Record`

Used by - `MessageType/Records, GroupType/Records` | |

Details	Source

	``xml
	<xs:complexType name="Record
	sType">
	  <xs:sequence>
	    <xs:element name="Record"
	type="RecordType"
	          maxOccurs="unbounded"/>
	  </xs:sequence>
	</xs:complexType>
	``

ComplexType - `RecordType`
 	
Children - `Name, TagName, MinOccur, MaxOccur, Fields`
 	
Used by - `RecordsType/Record`
	``xml
	<xs:complexType name="RecordType">
	
	  <xs:sequence>
	    <xs:element ref="Name"/>
	    <xs:element ref="TagName"/>
	    <xs:element ref="MinOccur"/>
	    <xs:element ref="MaxOccur"/>
	    <xs:element name="Fields"
	type="FieldsType"
	          minOccurs="0"/>
	  </xs:sequence>
	</xs:complexType>
	``

ComplexType - `MessageType` **Children** - `Name,                TagName,DefaultAlignment, DefaultPadding, DefaultTrim, DefaultEmptyFieldValue,                EscapeChar, RecordSeparator, FieldSeparator, Records, Group` **Used by** - `MessagesType/Message`	

Details	Source
| | ```xml
<xs:complexType name="Mes
sageType">
  <xs:sequence>
    <xs:element ref="Name"/>
    <xs:element ref="TagName"/>
    <xs:element name="DefaultAlign
ment"
         type="DefaultAlignment
Type"/>
    <xs:element name="Default
Padding"
         type="DefaultPadding
Type"/>
    <xs:element name="DefaultTrim"

         type="DefaultTrim
Type"/>
    <xs:element name="DefaultEmpty
FieldValue"
         type="DefaultEmptyField
ValueType"/>
    <xs:element name="EscapeChar"

         type="xs:string" minOc
curs="0"/>
    <xs:element name="RecordSepara
tor"
         type="xs:string"/>
    <xs:element name="FieldSepara
tor"
         type="xs:string" minOc
curs="0"/>
    <xs:sequence>
      <xs:element name="Records"
type="RecordsType"
         minOccurs="0"/>
      <xs:element name="Group"
type="GroupType"
         minOccurs="0" maxOc
curs="unbounded"/>
    </xs:sequence>
  </xs:sequence>
</xs:complexType>
``` |

ComplexType - `FieldType`

Children - `Name, TagName, DataType, PaddingChar, Length, StartPos, Alig nment, DefaultValue, Trim, EmptyFie ldValue,                Identifi er`

Used by - `FieldsType/Field`

Details	Source

	``xml
	<xs:complexType name="FieldType">
	  <xs:sequence>
	    <xs:element ref="Name"/>
	    <xs:element ref="TagName"/>
	    <xs:element ref="DataType"/>
	    <xs:element name="PaddingChar"
	
	         type="xs:string" minOc
	curs="0"/>
	    <xs:element name="Length"
	         type="xs:integer"/>
	    <xs:element name="StartPos"
	         type="xs:integer"/>
	    <xs:element name="Alignment"
	minOccurs="0">
	      <xs:simpleType>
	        <xs:restriction
	base="xs:string">
	         <xs:enumeration val
	ue="RIGHT"/>
	         <xs:enumeration val
	ue="LEFT"/>
	        </xs:restriction>
	      </xs:simpleType>
	    </xs:element>
	    <xs:element name="DefaultVal
	ue"
	         type="xs:string" minOc
	curs="0"/>
	    <xs:element name="Trim"
	         type="xs:integer" minOc
	curs="0"/>
	   <xs:element name="EmptyFieldVal
	ue"
	         type="xs:string" minOc
	curs="0"/>
	    <xs:element name="Identifier"
	
	         type="xs:string" minOc
	curs="0"/>
	  </xs:sequence>
	</xs:complexType>
	``

Appendix B: Flat File examples

This section contain examples and exercises for reference to available Flat File Parser features.

Caution:

When not sure about a feature, you are recommended to create a test scenario similar to the examples here. Then, verify that you have correctly interpreted the feature. If your required feature is not available use a short example and the desired output as basis of a feature request.

Record-Separated and Field-Separated Flat File to XML

This example shows one of the most straightforward types of transformations to perform with a parser. A flat file with semicolon-separated fields and end-of-line-separated records.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleOne</ID>
   <Messages>
    <Message>
              <IgnorePrecedingFieldSeparator>0</IgnorePrecedingFieldSep
arator>
              <DefaultEmptyFieldValue>
                 <Numeric>0</Numeric>
                 <Alpha></Alpha>
               </DefaultEmptyFieldValue>
               <DefaultAlignment>
                 <Numeric>RIGHT</Numeric>
                 <Alpha>LEFT</Alpha>
               </DefaultAlignment>
               <DefaultPadding>
                 <Numeric>0</Numeric>
                 <Alpha> </Alpha>
               </DefaultPadding>
               <DefaultTrim>
                 <Numeric>0</Numeric>
                 <Alpha>0</Alpha>
               </DefaultTrim>
     <Name>Sample</Name>
     <TagName>Sample</TagName>
     <EscapeChar/>
     <RecordSeparator>\n</RecordSeparator>
     <FieldSeparator>;</FieldSeparator>
     <Records>
      <Record>
       <Name>SampleRecord</Name>
       <TagName>SampleRecord</TagName>
       <MinOccur>1</MinOccur>
       <MaxOccur>0</MaxOccur>
       <Fields>
        <Field>
         <Name>SampleFieldOne</Name>
         <TagName>SampleFieldOne</TagName>
         <DataType>Alpha</DataType>
         <PaddingChar/>
```

```
        <DecimalLength/>
        <Alignment>LEFT</Alignment>
        <StartPos>0</StartPos>
        <Length>0</Length>
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
       <Field>
        <Name>SampleFieldTwo</Name>
        <TagName>SampleFieldTwo</TagName>
        <DataType>Alpha</DataType>
        <PaddingChar/>
        <DecimalLength/>
        <Alignment>LEFT</Alignment>
        <StartPos>0</StartPos>
        <Length>0</Length>
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
      </Fields>
     </Record>
    </Records>
   </Message>
  </Messages>
 </File>
 </Files>
</BankRepository>
```

Input File:

```
Record_One_Field_Data_One;Record_One_Field_Data_Two;
Record_Two_Field_Data_One;Record_Two_Field_Data_Two;
```

The expected result without the surrounding envelope will be:

```
<SampleMessage>
    <SampleRecord>
         <SampleFieldOne>Record_One_Field_Data_One</SampleFieldOne>
         <SampleFieldTwo>Record_One_Field_Data_Two</SampleFieldTwo>
    </SampleRecord>
    <SampleRecord>
         <SampleFieldOne>Record_Two_Field_Data_One</SampleFieldOne>
         <SampleFieldTwo>Record_Two_Field_Data_Two</SampleFieldTwo>
    </SampleRecord>
<SampleMessage>
```

XML to Record-Separated and Field-Separated FlatFile

This example is another type of transformations to perform with a parser. An XML file will be transformed into a flat file with semicolon-separated fields and end-of-line-separated records.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleTwo</ID>
   <Messages>
    <Message>
                <IgnorePrecedingFieldSeparator>0</IgnorePrecedingFieldSep
arator>
                <DefaultEmptyFieldValue>
                   <Numeric>0</Numeric>
                   <Alpha></Alpha>
                </DefaultEmptyFieldValue>
                <DefaultAlignment>
                   <Numeric>RIGHT</Numeric>
                   <Alpha>LEFT</Alpha>
                </DefaultAlignment>
                <DefaultPadding>
                   <Numeric>0</Numeric>
                   <Alpha> </Alpha>
                </DefaultPadding>
                <DefaultTrim>
                   <Numeric>0</Numeric>
                   <Alpha>0</Alpha>
                </DefaultTrim>
     <Name>Sample</Name>
     <TagName>Sample</TagName>
     <EscapeChar/>
     <RecordSeparator>\n</RecordSeparator>
     <FieldSeparator>;</FieldSeparator>
     <Records>
      <Record>
       <Name>SampleRecord</Name>
       <TagName>SampleRecord</TagName>
       <MinOccur>1</MinOccur>
       <MaxOccur>0</MaxOccur>
       <Fields>
        <Field>
         <Name>SampleFieldOne</Name>
         <TagName>SampleFieldOne</TagName>
         <DataType>Alpha</DataType>
         <PaddingChar/>
         <DecimalLength/>
         <Alignment>LEFT</Alignment>
         <StartPos>0</StartPos>
         <Length>0</Length>
         <DefaultValue></DefaultValue>
         <Identifier>0</Identifier>
        </Field>
        <Field>
         <Name>SampleFieldTwo</Name>
         <TagName>SampleFieldTwo</TagName>
         <DataType>Alpha</DataType>
         <PaddingChar/>
         <DecimalLength/>
```

```
            <Alignment>LEFT</Alignment>
            <StartPos>0</StartPos>
            <Length>0</Length>
            <DefaultValue></DefaultValue>
            <Identifier>0</Identifier>
          </Field>
        </Fields>
      </Record>
    </Records>
   </Message>
  </Messages>
 </File>
</Files>
</BankRepository>
```

Input file:

```
<SampleMessage>
      <SampleRecord>
              <SampleFieldOne>Record_One_Field_Data_One</SampleFieldOne>

              <SampleFieldTwo>Record_One_Field_Data_Two</SampleFieldTwo>

      </SampleRecord>
      <SampleRecord>
              <SampleFieldOne>Record_Two_Field_Data_One</SampleFieldOne>

              <SampleFieldTwo>Record_Two_Field_Data_Two</SampleFieldTwo>

      </SampleRecord>
<SampleMessage>
```

The expected result will be:

```
Record_One_Field_Data_One;Record_One_Field_Data_Two;
Record_Two_Field_Data_One;Record_Two_Field_Data_Two;
```

Record-Separated Fixed-Length Fields Flat File to XML

This example illustrates a transformation for fields that are not separated by a sequence of characters. These fields must have defined length.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleThree</ID>
   <Messages>
    <Message>
```

```
                <IgnorePrecedingFieldSeparator>0</IgnorePrecedingFieldSep
arator>
                <DefaultEmptyFieldValue>
                    <Numeric>0</Numeric>
                    <Alpha></Alpha>
                 </DefaultEmptyFieldValue>
                 <DefaultAlignment>
                    <Numeric>RIGHT</Numeric>
                    <Alpha>LEFT</Alpha>
                 </DefaultAlignment>
                 <DefaultPadding>
                    <Numeric>0</Numeric>
                    <Alpha> </Alpha>
                 </DefaultPadding>
                 <DefaultTrim>
                    <Numeric>0</Numeric>
                    <Alpha>0</Alpha>
                 </DefaultTrim>
    <Name>Sample</Name>
    <TagName>Sample</TagName>
    <EscapeChar/>
    <RecordSeparator>\n</RecordSeparator>
    <FieldSeparator></FieldSeparator>
    <Records>
     <Record>
      <Name>SampleRecord</Name>
      <TagName>SampleRecord</TagName>
      <MinOccur>1</MinOccur>
      <MaxOccur>0</MaxOccur>
      <Fields>
       <Field>
        <Name>SampleFieldOne</Name>
        <TagName>SampleFieldOne</TagName>
        <DataType>Alpha</DataType>
        <PaddingChar/>
        <DecimalLength/>
        <Alignment>LEFT</Alignment>
        <StartPos>1</StartPos>
        <Length>3</Length>
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
       <Field>
        <Name>SampleFieldTwo</Name>
        <TagName>SampleFieldTwo</TagName>
        <DataType>Alpha</DataType>
        <PaddingChar/>
        <DecimalLength/>
        <Alignment>LEFT</Alignment>
        <StartPos>4</StartPos>
        <Length>3</Length>
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
      </Fields>
     </Record>
```

```
      </Records>
    </Message>
   </Messages>
  </File>
 </Files>
</BankRepository>
```

Input file:

```
AAABBB
CCCDDD
```

The expected result without the surrounding envelope will be:

```
<Sample>
            <SampleRecord>
                <SampleFieldOne>AAA</SampleFieldOne>
                <SampleFieldTwo>BBB</SampleFieldTwo>
            </SampleRecord>
            <SampleRecord>
                <SampleFieldOne>CCC</SampleFieldOne>
                <SampleFieldTwo>DDD</SampleFieldTwo>
            </SampleRecord>
</Sample>
```

XML to record-separated fixed-length fields Flat File

This example is another type of transformation to perform with a parser. An XML file will be transformed into a record-separated fixed-length fields flat file.

Input File:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleFour</ID>
   <Messages>
    <Message>
              <IgnorePrecedingFieldSeparator>0</IgnorePrecedingFieldSep
arator>
              <DefaultEmptyFieldValue>
                 <Numeric>0</Numeric>
                 <Alpha></Alpha>
              </DefaultEmptyFieldValue>
              <DefaultAlignment>
                 <Numeric>RIGHT</Numeric>
                 <Alpha>LEFT</Alpha>
              </DefaultAlignment>
              <DefaultPadding>
                 <Numeric>0</Numeric>
                 <Alpha> </Alpha>
```

```
                </DefaultPadding>
                <DefaultTrim>
                  <Numeric>0</Numeric>
                  <Alpha>0</Alpha>
                </DefaultTrim>
      <Name>Sample</Name>
      <TagName>Sample</TagName>
      <EscapeChar/>
      <RecordSeparator>\n</RecordSeparator>
      <FieldSeparator></FieldSeparator>
      <Records>
       <Record>
        <Name>SampleRecord</Name>
        <TagName>SampleRecord</TagName>
        <MinOccur>1</MinOccur>
        <MaxOccur>0</MaxOccur>
        <Fields>
         <Field>
          <Name>SampleFieldOne</Name>
          <TagName>SampleFieldOne</TagName>
          <DataType>Alpha</DataType>
          <PaddingChar/>
          <DecimalLength/>
          <Alignment>LEFT</Alignment>
          <StartPos>1</StartPos>
          <Length>3</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
         <Field>
          <Name>SampleFieldTwo</Name>
          <TagName>SampleFieldTwo</TagName>
          <DataType>Alpha</DataType>
          <PaddingChar/>
          <DecimalLength/>
          <Alignment>LEFT</Alignment>
          <StartPos>4</StartPos>
          <Length>3</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
        </Fields>
       </Record>
      </Records>
     </Message>
    </Messages>
   </File>
  </Files>
</BankRepository>
```

Input file:

```
<SampleMessage>
        <SampleRecord>
```

```
            <SampleFieldOne>AAA_One</SampleFieldOne>
            <SampleFieldTwo>BBB</SampleFieldTwo>
        </SampleRecord>
        <SampleRecord>
            <SampleFieldOne>CCC</SampleFieldOne>
            <SampleFieldTwo>DDD</SampleFieldTwo>
        </SampleRecord>
<SampleMessage>
```

The expected result will be:

```
AAABBB
CCCDDD
```

Fixed-length records and field flat file to XML

When no record separators are available, the records must consist of all fixed length fields. This example illustrates a transformation from fixed-length record and field flat file to XML.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleFive</ID>
   <Messages>
    <Message>
                <IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSep
arator>
                <DefaultEmptyFieldValue>
                   <Numeric>0</Numeric>
                   <Alpha></Alpha>
                </DefaultEmptyFieldValue>
                <DefaultAlignment>
                   <Numeric>RIGHT</Numeric>
                   <Alpha>LEFT</Alpha>
                </DefaultAlignment>
                <DefaultPadding>
                   <Numeric>0</Numeric>
                   <Alpha> </Alpha>
                </DefaultPadding>
                <DefaultTrim>
                   <Numeric>1</Numeric>
                   <Alpha>0</Alpha>
                </DefaultTrim>
     <Name>TopNode</Name>
     <TagName>TopNode</TagName>
     <EscapeChar/>
     <RecordSeparator></RecordSeparator>
     <FieldSeparator></FieldSeparator>
     <FixedLengthSeparator>1</FixedLengthSeparator>
     <Records>
```

```
    <Record>
     <Name>RecordOne</Name>
     <TagName>RecordOne</TagName>
     <MinOccur>1</MinOccur>
     <MaxOccur>0</MaxOccur>
     <Fields>
      <Field>
       <Name>Field1</Name>
       <TagName>Field1</TagName>
       <DataType>Alpha</DataType>
       <PaddingChar/>
       <DecimalLength/>
       <Alignment>LEFT</Alignment>
       <StartPos>1</StartPos>
       <Length>1</Length>
       <DefaultValue></DefaultValue>
       <Identifier>0</Identifier>
      </Field>
      <Field>
       <Name>Field2</Name>
       <TagName>Field2</TagName>
       <DataType>Alpha</DataType>
       <PaddingChar/>
       <DecimalLength/>
       <Alignment>LEFT</Alignment>
       <StartPos>2</StartPos>
       <Length>1</Length>
       <DefaultValue></DefaultValue>
       <Identifier>0</Identifier>
      </Field>
     </Fields>
    </Record>
   </Records>
  </Message>
 </Messages>
 </File>
 </Files>
</BankRepository>
```

Input file content:

```
abCDeFGh
```

The expected result without the surrounding envelope will be:

```
<TopNode>
    <RecordOne>
        <Field1>a</Field1>
        <Field2>b</Field2>
    </RecordOne>>
    <RecordOne>
        <Field1>C</Field1>
        <Field2>D</Field2>
```

```
    </RecordOne>>
    <RecordOne>
        <Field1>e</Field1>
        <Field2>F</Field2>
    </RecordOne>>
    <RecordOne>
        <Field1>G</Field1>
        <Field2>h</Field2>
    </RecordOne>
</TopNode>
```

XML to Fixed-Length Records and Fields

This example is another type of transformation to perform with a parser. An XML file is transformed into a flat file with fixed-length records and fields.

Input File:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleSix</ID>
   <Messages>
    <Message>
               <IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSep
arator>
               <DefaultEmptyFieldValue>
                  <Numeric>0</Numeric>
                  <Alpha></Alpha>
                </DefaultEmptyFieldValue>
                <DefaultAlignment>
                  <Numeric>RIGHT</Numeric>
                  <Alpha>LEFT</Alpha>
                </DefaultAlignment>
                <DefaultPadding>
                  <Numeric>0</Numeric>
                  <Alpha> </Alpha>
                </DefaultPadding>
                <DefaultTrim>
                  <Numeric>1</Numeric>
                  <Alpha>0</Alpha>
                </DefaultTrim>
     <Name>TopNode</Name>
     <TagName>TopNode</TagName>
     <EscapeChar/>
     <RecordSeparator></RecordSeparator>
     <FieldSeparator></FieldSeparator>
     <FixedLengthSeparator>1</FixedLengthSeparator>
     <Records>
      <Record>
        <Name>RecordOne</Name>
        <TagName>RecordOne</TagName>
```

```
        <MinOccur>1</MinOccur>
        <MaxOccur>0</MaxOccur>
        <Fields>
         <Field>
          <Name>Field1</Name>
          <TagName>Field1</TagName>
          <DataType>Alpha</DataType>
          <PaddingChar/>
          <DecimalLength/>
          <Alignment>LEFT</Alignment>
          <StartPos>1</StartPos>
          <Length>1</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
         <Field>
          <Name>Field2</Name>
          <TagName>Field2</TagName>
          <DataType>Alpha</DataType>
          <PaddingChar/>
          <DecimalLength/>
          <Alignment>LEFT</Alignment>
          <StartPos>2</StartPos>
          <Length>1</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
        </Fields>
       </Record>
      </Records>
     </Message>
    </Messages>
   </File>
  </Files>
</BankRepository>
```

Input file:

```
<TopNode>
    <RecordOne>
        <Field1>a</Field1>
        <Field2>b</Field2>
    </RecordOne>>
    <RecordOne>
        <Field1>C</Field1>
        <Field2>D</Field2>
    </RecordOne>>
    <RecordOne>
        <Field1>e</Field1>
        <Field2>F</Field2>
    </RecordOne>>
    <RecordOne>
        <Field1>G</Field1>
        <Field2>h</Field2>
```

```
      </RecordOne>
</TopNode>
```

The expected result will be:

```
abCDeFGh
```

Default empty field value

These examples show transformations that use empty field values. When a field is empty, the value to write is defined using the default empty field value.

Flat to XML with Empty Fields

This example shows an input flat file that is field-separated with some empty fields. The generated XML output will be substituted with the default empty value.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleSeven</ID>
   <Messages>
    <Message>
               <IgnorePrecedingFieldSeparator>0</IgnorePrecedingFieldSep
arator>
               <DefaultEmptyFieldValue>
                  <Numeric>0</Numeric>
                  <Alpha>X</Alpha>
               </DefaultEmptyFieldValue>
               <DefaultAlignment>
                  <Numeric>RIGHT</Numeric>
                  <Alpha>LEFT</Alpha>
               </DefaultAlignment>
               <DefaultPadding>
                  <Numeric>0</Numeric>
                  <Alpha> </Alpha>
               </DefaultPadding>
               <DefaultTrim>
                  <Numeric>1</Numeric>
                  <Alpha></Alpha>
               </DefaultTrim>
     <Name>TopNode</Name>
     <TagName>TopNode</TagName>
     <EscapeChar/>
     <RecordSeparator>\n</RecordSeparator>
     <FieldSeparator>:</FieldSeparator>
     <Records>
      <Record>
        <Name>RecordOne</Name>
```

```
        <TagName>RecordOne</TagName>
        <MinOccur>1</MinOccur>
        <MaxOccur>0</MaxOccur>
        <Fields>
         <Field>
          <Name>Field1</Name>
          <TagName>Field1</TagName>
          <DataType>Alpha</DataType>
          <PaddingChar/>
          <DecimalLength/>
          <Alignment>LEFT</Alignment>
          <StartPos>0</StartPos>
          <Length>0</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
         <Field>
          <Name>Field2</Name>
          <TagName>Field2</TagName>
          <DataType>Alpha</DataType>
          <PaddingChar/>
          <DecimalLength/>
          <Alignment>LEFT</Alignment>
          <StartPos>0</StartPos>
          <Length>0</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
        </Fields>
       </Record>
      </Records>
     </Message>
    </Messages>
   </File>
 </Files>
</BankRepository>
```

Input file:

```
a:b:
c::
::
:d:
e:f:
```

The expected result will be:

```
<TopNode>
        <RecordOne>
              <Field1>a</Field1>
               <Field2>b</Field2>
        </RecordOne>
        <RecordOne>
              <Field1>c</Field1>
```

```
            <Field2>X</Field2>
        </RecordOne>
        <RecordOne>
            <Field1>X</Field1>
            <Field2>X</Field2>
        </RecordOne>
        <RecordOne>
            <Field1>X</Field1>
            <Field2>d</Field2>
        </RecordOne>
        <RecordOne>
             <Field1>e</Field1>
             <Field2>f</Field2>
        </RecordOne>
</TopNode>
```

XML to Flat with Empty Fields

This example shows an input XML file with empty content for some fields. The generated flat file output will be substituted with the default empty value. When writing fixed-length flat files, the alignment and padding must be defined because the empty field value will be aligned and padded to fill the specified length.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleSeven</ID>
   <Messages>
    <Message>
                <IgnorePrecedingFieldSeparator>0</IgnorePrecedingFieldSep
arator>
                <DefaultEmptyFieldValue>
                   <Numeric>0</Numeric>
                   <Alpha>X</Alpha>
                </DefaultEmptyFieldValue>
                <DefaultAlignment>
                   <Numeric>RIGHT</Numeric>
                   <Alpha>LEFT</Alpha>
                </DefaultAlignment>
                <DefaultPadding>
                   <Numeric>0</Numeric>
                   <Alpha> </Alpha>
                </DefaultPadding>
                <DefaultTrim>
                   <Numeric>1</Numeric>
                   <Alpha></Alpha>
                </DefaultTrim>
    <Name>TopNode</Name>
    <TagName>TopNode</TagName>
    <EscapeChar/>
    <RecordSeparator>\n</RecordSeparator>
    <FieldSeparator>:</FieldSeparator>
    <Records>
```

```
     <Record>
      <Name>RecordOne</Name>
      <TagName>RecordOne</TagName>
      <MinOccur>1</MinOccur>
      <MaxOccur>0</MaxOccur>
      <Fields>
       <Field>
        <Name>Field1</Name>
        <TagName>Field1</TagName>
        <DataType>Alpha</DataType>
        <PaddingChar/>
        <DecimalLength/>
        <Alignment>LEFT</Alignment>
        <StartPos>0</StartPos>
        <Length>0</Length>
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
       <Field>
        <Name>Field2</Name>
        <TagName>Field2</TagName>
        <DataType>Alpha</DataType>
        <PaddingChar/>
        <DecimalLength/>
        <Alignment>LEFT</Alignment>
        <StartPos>0</StartPos>
        <Length>0</Length>
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
      </Fields>
     </Record>
    </Records>
   </Message>
  </Messages>
  </File>
 </Files>
</BankRepository>
```

Input File:

```
<Record>
     <FieldOne>a</FieldOne>
     <FieldTwo>b</FieldTwo>
</Record>
<Record>
     <FieldOne/>
     <FieldTwo/>
</Record>
<Record>
     <FieldOne>c</FieldOne>
     <FieldTwo/>
</Record>
<Record>
     <FieldOne/>
```

```
    <FieldTwo>d</FieldTwo>
</Record>
<Record>
    <FieldOne>e</FieldOne>
    <FieldTwo>f</FieldTwo>
</Record>
```

The expected result will be:

```
a:b:
X:X:
c:X:
X:d:
e:f:
```

Padding and alignment

When a field has fixed length and the data is shorter than the specified length, the data must be padded with additional characters for it to be able to reach the specified length. These examples show transformations that use padding and alignment.

Flat File with Padding to XML

This example shows a transformation in which the padding of the flat file and left alignment is removed in the generated XML output.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleEight</ID>
   <Messages>
    <Message>
               <IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSep
arator>
               <DefaultEmptyFieldValue>
                  <Numeric>0</Numeric>
                  <Alpha></Alpha>
               </DefaultEmptyFieldValue>
               <DefaultAlignment>
                  <Numeric>RIGHT</Numeric>
                  <Alpha>LEFT</Alpha>
               </DefaultAlignment>
               <DefaultPadding>
                  <Numeric>0</Numeric>
                  <Alpha>X</Alpha>
               </DefaultPadding>
               <DefaultTrim>
                  <Numeric>1</Numeric>
                  <Alpha>1</Alpha>
               </DefaultTrim>
```

```
      <Name>TopNode</Name>
      <TagName>TopNode</TagName>
      <EscapeChar/>
      <RecordSeparator>\n</RecordSeparator>
      <FieldSeparator></FieldSeparator>
      <Records>
       <Record>
        <Name>RecordOne</Name>
        <TagName>RecordOne</TagName>
        <MinOccur>1</MinOccur>
        <MaxOccur>0</MaxOccur>
        <Fields>
         <Field>
          <Name>Field1</Name>
          <TagName>Field1</TagName>
          <DataType>Alpha</DataType>
          <StartPos>1</StartPos>
          <Length>3</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
         <Field>
          <Name>Field2</Name>
          <TagName>Field2</TagName>
          <DataType>Alpha</DataType>
          <StartPos>4</StartPos>
          <Length>3</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
        </Fields>
       </Record>
      </Records>
     </Message>
    </Messages>
   </File>
  </Files>
</BankRepository>
```

Input file:

```
XXAXXB
XCXDXX
```

The expected result will be:

```
<TopNode>
 <RecordOne>
  <Field1>XXA</Field1>
  <Field2>XXB</Field2>
 </RecordOne>
 <RecordOne>
  <Field1>XC</Field1>
  <Field2>D</Field2>
```

```
    </RecordOne>
</TopNode>
```

XML to Flat File with Padding

This example shows a transformation of an XML file to a flat file output with padding. Fields are padded when they are too short.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleEight</ID>
   <Messages>
    <Message>
                <IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSep
arator>
                <DefaultEmptyFieldValue>
                   <Numeric>0</Numeric>
                   <Alpha></Alpha>
                </DefaultEmptyFieldValue>
                <DefaultAlignment>
                   <Numeric>RIGHT</Numeric>
                   <Alpha>LEFT</Alpha>
                </DefaultAlignment>
                <DefaultPadding>
                   <Numeric>0</Numeric>
                   <Alpha>X</Alpha>
                </DefaultPadding>
                <DefaultTrim>
                   <Numeric>1</Numeric>
                   <Alpha>1</Alpha>
                </DefaultTrim>
     <Name>TopNode</Name>
     <TagName>TopNode</TagName>
     <EscapeChar/>
     <RecordSeparator>\n</RecordSeparator>
     <FieldSeparator></FieldSeparator>
     <Records>
      <Record>
       <Name>RecordOne</Name>
       <TagName>RecordOne</TagName>
       <MinOccur>1</MinOccur>
       <MaxOccur>0</MaxOccur>
       <Fields>
        <Field>
         <Name>Field1</Name>
         <TagName>Field1</TagName>
         <DataType>Alpha</DataType>
         <StartPos>1</StartPos>
         <Length>3</Length>
         <DefaultValue></DefaultValue>
         <Identifier>0</Identifier>
        </Field>
```

```
          <Field>
           <Name>Field2</Name>
           <TagName>Field2</TagName>
           <DataType>Alpha</DataType>
           <StartPos>4</StartPos>
           <Length>3</Length>
           <DefaultValue></DefaultValue>
           <Identifier>0</Identifier>
          </Field>
         </Fields>
        </Record>
       </Records>
      </Message>
     </Messages>
    </File>
   </Files>
</BankRepository>
```

Input File:

```
<TopNode>
 <RecordOne>
  <Field1>XXA</Field1>
  <Field2>XXB</Field2>
 </RecordOne>
 <RecordOne>
  <Field1>XC</Field1>
  <Field2>D</Field2>
 </RecordOne>
</TopNode>
```

The expected result will be:

```
XXAXXB
XCXDXX
```

Trim

This example shows XML to flat file transformation in which the white space surrounding the data in XML is removed. Trimming is not applicable in the flat file to XML transformation.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleNine</ID>
   <Messages>
    <Message>
              <IgnorePrecedingFieldSeparator>0</IgnorePrecedingFieldSep
arator>
              <DefaultEmptyFieldValue>
```

```
                <Numeric></Numeric>
                <Alpha></Alpha>
              </DefaultEmptyFieldValue>
              <DefaultAlignment>
                <Numeric>RIGHT</Numeric>
                <Alpha>LEFT</Alpha>
              </DefaultAlignment>
              <DefaultPadding>
                <Numeric>0</Numeric>
                <Alpha> </Alpha>
              </DefaultPadding>
              <DefaultTrim>
                <Numeric>1</Numeric>
                <Alpha>1</Alpha>
              </DefaultTrim>
      <Name>TopNode</Name>
      <TagName>TopNode</TagName>
      <EscapeChar/>
      <RecordSeparator>\n</RecordSeparator>
      <FieldSeparator>:</FieldSeparator>
      <Records>
       <Record>
        <Name>Record</Name>
        <TagName>Record</TagName>
        <MinOccur>1</MinOccur>
        <MaxOccur>0</MaxOccur>
        <Fields>
         <Field>
          <Name>FieldOne</Name>
          <TagName>FieldOne</TagName>
          <DataType>Alpha</DataType>
          <PaddingChar/>
          <DecimalLength/>
          <Alignment>LEFT</Alignment>
          <StartPos>0</StartPos>
          <Length>0</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
         <Field>
          <Name>FieldTwo</Name>
          <TagName>FieldTwo</TagName>
          <DataType>Alpha</DataType>
          <PaddingChar/>
          <DecimalLength/>
          <Alignment>LEFT</Alignment>
          <StartPos>0</StartPos>
          <Length>0</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
        </Fields>
       </Record>
      </Records>
     </Message>
    </Messages>
```

```
   </File>
  </Files>
</BankRepository>
```

Input File:

```
<TopNode>
    <Record>
        <FieldOne> a </FieldOne>
        <FieldTwo> b </FieldTwo>
    </Record>
    <Record>
        <FieldOne> e</FieldOne>
        <FieldTwo>f </FieldTwo>
    </Record>
</TopNode>
```

The expected result will be:

```
a:b:
e:f:
```

Escape character

This example shows flat file to XML transformation in which the effect of a field or record separator that occurs as data in the input flat file is negated using an escape character prefix. The escape sequence will be included in the data.

These are typical issues encountered with escape sequences:

• Escapewithafixed length is not tested.

• If XML data contains a separator as data, the parser will not escape the separator and an incorrect flat file will be generated.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleTen</ID>
   <Messages>
    <Message>
                <IgnorePrecedingFieldSeparator>0</IgnorePrecedingFieldSep
arator>
                <DefaultEmptyFieldValue>
                    <Numeric></Numeric>
                    <Alpha></Alpha>
                 </DefaultEmptyFieldValue>
                 <DefaultAlignment>
```

```
              <Numeric>RIGHT</Numeric>
              <Alpha>LEFT</Alpha>
            </DefaultAlignment>
            <DefaultPadding>
              <Numeric></Numeric>
              <Alpha></Alpha>
            </DefaultPadding>
            <DefaultTrim>
              <Numeric>0</Numeric>
              <Alpha>0</Alpha>
            </DefaultTrim>
    <Name>TopNode</Name>
    <TagName>TopNode</TagName>
    <EscapeChar>++</EscapeChar>
    <RecordSeparator>\n</RecordSeparator>
    <FieldSeparator>:</FieldSeparator>
    <Records>
     <Record>
      <Name>RecordOne</Name>
      <TagName>RecordOne</TagName>
      <MinOccur>1</MinOccur>
      <MaxOccur>0</MaxOccur>
      <Fields>
       <Field>
        <Name>Field1</Name>
        <TagName>Field1</TagName>
        <DataType>Alpha</DataType>
        <PaddingChar/>
        <DecimalLength/>
        <Alignment>LEFT</Alignment>
        <StartPos>0</StartPos>
        <Length>0</Length>
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
       <Field>
        <Name>Field2</Name>
        <TagName>Field2</TagName>
        <DataType>Alpha</DataType>
        <PaddingChar/>
        <DecimalLength/>
        <Alignment>LEFT</Alignment>
        <StartPos>0</StartPos>
        <Length>0</Length>
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
      </Fields>
     </Record>
    </Records>
   </Message>
  </Messages>
 </File>
</Files>
</BankRepository>
```

Input File:

```
aa++:aa:bb++:bb:
cc++::dd++::
++:ee:++:ff:
g:h:
i:++
j:
```

The expected result will be:

```
<TopNode>
   <RecordOne>
       <Field1>aa:aa</Field1>
       <Field2>bb:bb</Field2>
   </RecordOne>
   <RecordOne>
       <Field1>cc:</Field1>
       <Field2>dd:</Field2>
   </RecordOne>
   <RecordOne>
       <Field1>:ee</Field1>
       <Field2>:ff</Field2>
   </RecordOne>
   <RecordOne>
       <Field1>g</Field1>
       <Field2>h</Field2>
  </RecordOne>
  <RecordOne>
       <Field1>i</Field1>
       <Field2>++j:</Field2>
   </RecordOne>
</TopNode>
```

Ignore preceding and trailing field separator

These examples show transformations that use ignore preceding and trailing field separator.

Ignore Preceding

If this option is used a record that starts with a field separator ignores this first field. Otherwise, it is considered an empty field.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleEleven</ID>
   <Messages>
    <Message>
```

```
                <IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSep
arator>
                <DefaultEmptyFieldValue>
                   <Numeric>0</Numeric>
                   <Alpha></Alpha>
                </DefaultEmptyFieldValue>
                <DefaultAlignment>
                   <Numeric>RIGHT</Numeric>
                   <Alpha>LEFT</Alpha>
                </DefaultAlignment>
                <DefaultPadding>
                   <Numeric>0</Numeric>
                   <Alpha> </Alpha>
                </DefaultPadding>
                <DefaultTrim>
                   <Numeric>1</Numeric>
                   <Alpha>0</Alpha>
                </DefaultTrim>
     <Name>TopNode</Name>
     <TagName>TopNode</TagName>
     <EscapeChar/>
     <RecordSeparator>\n</RecordSeparator>
     <FieldSeparator>:</FieldSeparator>
     <Records>
      <Record>
       <Name>RecordOne</Name>
       <TagName>RecordOne</TagName>
       <MinOccur>1</MinOccur>
       <MaxOccur>0</MaxOccur>
       <Fields>
        <Field>
         <Name>Field1</Name>
         <TagName>Field1</TagName>
         <DataType>Alpha</DataType>
         <PaddingChar/>
         <DecimalLength/>
         <Alignment>LEFT</Alignment>
         <StartPos>0</StartPos>
         <Length>0</Length>
         <DefaultValue></DefaultValue>
         <Identifier>0</Identifier>
        </Field>
        <Field>
         <Name>Field2</Name>
         <TagName>Field2</TagName>
         <DataType>Alpha</DataType>
         <PaddingChar/>
         <DecimalLength/>
         <Alignment>LEFT</Alignment>
         <StartPos>0</StartPos>
         <Length>0</Length>
         <DefaultValue></DefaultValue>
         <Identifier>0</Identifier>
        </Field>
       </Fields>
      </Record>
```

```
        </Records>
      </Message>
     </Messages>
    </File>
  </Files>
</BankRepository>
```

Input File:

Input without a preceding field separator. Note that this input is not used, it is only shown as a reference.

```
a:b:
c:d
```

Input with preceding field separator will be:

```
a:b:
c:d
```

Ignore Trailing

If a field separator is the last field in a record, it is not mandatory. And, this is not a configurable behavior.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleEleven</ID>
   <Messages>
    <Message>
               <IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSep
arator>
               <DefaultEmptyFieldValue>
                  <Numeric>0</Numeric>
                  <Alpha></Alpha>
                </DefaultEmptyFieldValue>
                <DefaultAlignment>
                  <Numeric>RIGHT</Numeric>
                  <Alpha>LEFT</Alpha>
                </DefaultAlignment>
                <DefaultPadding>
                  <Numeric>0</Numeric>
                  <Alpha> </Alpha>
                </DefaultPadding>
                <DefaultTrim>
                  <Numeric>1</Numeric>
                  <Alpha>0</Alpha>
                </DefaultTrim>
      <Name>TopNode</Name>
      <TagName>TopNode</TagName>
      <EscapeChar/>
      <RecordSeparator>\n</RecordSeparator>
```

```
      <FieldSeparator>:</FieldSeparator>
      <Records>
       <Record>
        <Name>RecordOne</Name>
        <TagName>RecordOne</TagName>
        <MinOccur>1</MinOccur>
        <MaxOccur>0</MaxOccur>
        <Fields>
         <Field>
          <Name>Field1</Name>
          <TagName>Field1</TagName>
          <DataType>Alpha</DataType>
          <PaddingChar/>
          <DecimalLength/>
          <Alignment>LEFT</Alignment>
          <StartPos>0</StartPos>
          <Length>0</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
         <Field>
          <Name>Field2</Name>
          <TagName>Field2</TagName>
          <DataType>Alpha</DataType>
          <PaddingChar/>
          <DecimalLength/>
          <Alignment>LEFT</Alignment>
          <StartPos>0</StartPos>
          <Length>0</Length>
          <DefaultValue></DefaultValue>
          <Identifier>0</Identifier>
         </Field>
        </Fields>
       </Record>
      </Records>
     </Message>
    </Messages>
   </File>
  </Files>
</BankRepository>
```

Input File:

Input without trailing field separator and with preceding separator:

```
:a:b
:c:d
```

Record-Separated Semi-Fixed-Length Fields To and From XML

Using "-1" as the length value of the last field in a record means that the field is of undefined length and that the end of the record will be used to signal the end of the field.

Here are two transformation examples.

Record-Separated Semi-Fixed Length Fields to XML

Here is an example on how a record-separated semi-fixed length fields in a flat file is transformed to XML.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleTwelve</ID>
   <Messages>
    <Message>
               <IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSep
arator>
               <DefaultEmptyFieldValue>
                  <Numeric>0</Numeric>
                  <Alpha></Alpha>
                </DefaultEmptyFieldValue>
                <DefaultAlignment>
                  <Numeric>RIGHT</Numeric>
                  <Alpha>LEFT</Alpha>
                </DefaultAlignment>
                <DefaultPadding>
                  <Numeric>0</Numeric>
                  <Alpha> </Alpha>
                </DefaultPadding>
                <DefaultTrim>
                  <Numeric>1</Numeric>
                  <Alpha>0</Alpha>
                </DefaultTrim>
     <Name>TopNode</Name>
     <TagName>TopNode</TagName>
     <EscapeChar/>
     <RecordSeparator>CRLF</RecordSeparator>
     <FieldSeparator></FieldSeparator>
     <!--<FixedLengthSeparator>1</FixedLengthSeparator>-->
     <Records>
      <Record>
       <Name>RecordOne</Name>
       <TagName>RecordOne</TagName>
       <MinOccur>1</MinOccur>
       <MaxOccur>0</MaxOccur>
       <Fields>
        <Field>
         <Name>Field1</Name>
         <TagName>Field1</TagName>
         <DataType>Alpha</DataType>
         <PaddingChar/>
         <DecimalLength/>
         <Alignment>LEFT</Alignment>
         <StartPos>1</StartPos>
         <Length>3</Length>
```

```xml
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
       <Field>
        <Name>Field2</Name>
        <TagName>Field2</TagName>
        <DataType>Alpha</DataType>
        <PaddingChar/>
        <DecimalLength/>
        <Alignment>LEFT</Alignment>
        <StartPos>4</StartPos>
        <Length>3</Length>
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
       <Field>
        <Name>Field3</Name>
        <TagName>Field3</TagName>
        <DataType>Alpha</DataType>
        <PaddingChar/>
        <DecimalLength/>
        <Alignment>LEFT</Alignment>
        <StartPos>7</StartPos>
        <Length>-1</Length>
        <DefaultValue></DefaultValue>
        <Identifier>0</Identifier>
       </Field>
      </Fields>
     </Record>
    </Records>
   </Message>
  </Messages>
 </File>
 </Files>
</BankRepository>
```

The input with variable length last field is:

```
A B CDEFG
1 4 789
```

The expected result will be:

```xml
<TopNode>
 <RecordOne>
  <Field1>A</Field1>
  <Field2>B</Field2>
  <Field3>CDEFG</Field3>
 </RecordOne>
 <RecordOne>
  <Field1>1</Field1>
  <Field2>4</Field2>
  <Field3>789</Field3>
```

```
    </RecordOne>
</TopNode>
```

XML to Record-Separated Semi-Fixed-Length Fields

This is the opposite of the previous example. The output and repositories in the previous example are used here to generate new sample output.

Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleTwelve</ID>
   <Messages>
    <Message>
               <IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSep
arator>
               <DefaultEmptyFieldValue>
                  <Numeric>0</Numeric>
                  <Alpha></Alpha>
                </DefaultEmptyFieldValue>
                <DefaultAlignment>
                  <Numeric>RIGHT</Numeric>
                  <Alpha>LEFT</Alpha>
                </DefaultAlignment>
                <DefaultPadding>
                  <Numeric>0</Numeric>
                  <Alpha> </Alpha>
                </DefaultPadding>
                <DefaultTrim>
                  <Numeric>1</Numeric>
                  <Alpha>0</Alpha>
                </DefaultTrim>
      <Name>TopNode</Name>
      <TagName>TopNode</TagName>
      <EscapeChar/>
      <RecordSeparator>CRLF</RecordSeparator>
      <FieldSeparator></FieldSeparator>
      <!--<FixedLengthSeparator>1</FixedLengthSeparator>-->
      <Records>
       <Record>
         <Name>RecordOne</Name>
         <TagName>RecordOne</TagName>
         <MinOccur>1</MinOccur>
         <MaxOccur>0</MaxOccur>
         <Fields>
          <Field>
            <Name>Field1</Name>
            <TagName>Field1</TagName>
            <DataType>Alpha</DataType>
            <PaddingChar/>
            <DecimalLength/>
            <Alignment>LEFT</Alignment>
            <StartPos>1</StartPos>
```

```
         <Length>3</Length>
         <DefaultValue></DefaultValue>
         <Identifier>0</Identifier>
        </Field>
        <Field>
         <Name>Field2</Name>
         <TagName>Field2</TagName>
         <DataType>Alpha</DataType>
         <PaddingChar/>
         <DecimalLength/>
         <Alignment>LEFT</Alignment>
         <StartPos>4</StartPos>
         <Length>3</Length>
         <DefaultValue></DefaultValue>
         <Identifier>0</Identifier>
        </Field>
        <Field>
         <Name>Field3</Name>
         <TagName>Field3</TagName>
         <DataType>Alpha</DataType>
         <PaddingChar/>
         <DecimalLength/>
         <Alignment>LEFT</Alignment>
         <StartPos>7</StartPos>
         <Length>-1</Length>
         <DefaultValue></DefaultValue>
         <Identifier>0</Identifier>
        </Field>
       </Fields>
      </Record>
     </Records>
    </Message>
   </Messages>
  </File>
 </Files>
</BankRepository>
```

Input file:

```
<TopNode>
 <RecordOne>
  <Field1>A</Field1>
  <Field2>B</Field2>
  <Field3>CDEFG</Field3>
 </RecordOne>
 <RecordOne>
  <Field1>1</Field1>
  <Field2>4</Field2>
  <Field3>789</Field3>
 </RecordOne>
</TopNode>
```

The expected result will be:

```
A B CDEFG
1 4 789
```

Variable number of occurrences, groups and identifiers

Here are two transformation examples that use variable number of occurrences, groups, and identifiers. Groups are containers used to manage a group of records and other groups, often with a variable number of occurrences. The number of occurrences of a record can be controlled. To identify a record an identifier can be used.

Example 13 A

This example shows how a field and record-separated flat file with identifiers is transformed to XML.

Flat File Definition:

```
<BankRepository>
 <Files>
      <File>
        <ID>SampleThirteenA</ID>
   <Messages>
    <Message>
     <IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSeparator>
     <DefaultEmptyFieldValue><Numeric>0</Numeric><Alpha></Alpha></Default
EmptyFieldValue>
     <DefaultAlignment><Numeric>RIGHT</Numeric><Alpha>LEFT</Alpha></De
faultAlignment>
     <DefaultPadding><Numeric>0</Numeric><Alpha> </Alpha></DefaultPadding>

     <DefaultTrim><Numeric>1</Numeric><Alpha>0</Alpha></DefaultTrim>
     <Name>Purchases</Name>
     <TagName>PurchasedItems</TagName>
     <EscapeChar/>
     <RecordSeparator>\n</RecordSeparator>
     <FieldSeparator>:</FieldSeparator>
               <Records>
                  <Group>
                     <Name>Purchase</Name>
                     <TagName>Purchase</TagName>
                     <MinOccur>0</MinOccur>
                     <MaxOccur>U</MaxOccur>
                     <Records>
                        <Record>
                           <Name>CustomerDetails</Name>
                           <TagName>CustomerDetails</TagName>
                           <MinOccur>1</MinOccur>
                           <MaxOccur>1</MaxOccur>
                           <Fields>
                              <Field>
                                 <Name>ID</Name>
```

```
                    <TagName>Identifier</TagName>
                    <DataType>Alpha</DataType>
                    <PaddingChar/>
                    <DecimalLength/>
                    <Alignment>LEFT</Alignment>
                    <StartPos>0</StartPos>
                    <Length>0</Length>
                    <DefaultValue>CUSTOMER</DefaultValue>
                    <Identifier>1</Identifier>
                </Field>
                <Field>
                    <Name>Name</Name>
                    <TagName>Name</TagName>
                    <DataType>Alpha</DataType>
                    <PaddingChar/>
                    <DecimalLength/>
                    <Alignment>LEFT</Alignment>
                    <StartPos>0</StartPos>
                    <Length>0</Length>
                    <DefaultValue></DefaultValue>
                    <Identifier>0</Identifier>
                </Field>
                <Field>
                    <Name>Number</Name>
                    <TagName>Number</TagName>
                    <DataType>Alpha</DataType>
                    <PaddingChar/>
                    <DecimalLength/>
                    <Alignment>LEFT</Alignment>
                    <StartPos>0</StartPos>
                    <Length>0</Length>
                    <DefaultValue></DefaultValue>
                    <Identifier>0</Identifier>
                </Field>
            </Fields>
        </Record>
        <Group>
            <Name>Payment</Name>
            <TagName>Payment</TagName>
            <MinOccur>1</MinOccur>
            <MaxOccur>1</MaxOccur>
            <Records>
                <Record>
                    <Name>PaymentDetails</Name>
                    <TagName>PaymentDetails</TagName>
                    <MinOccur>0</MinOccur>
                    <MaxOccur>1</MaxOccur>
                    <Fields>
                        <Field>
                            <Name>ID</Name>
                            <TagName>Identifier</TagName>
                            <DataType>Alpha</DataType>
                            <PaddingChar/>
                            <DecimalLength/>
                            <Alignment>LEFT</Alignment>
```

```
                                         <StartPos>0</StartPos>
                                         <Length>0</Length>
                              <DefaultValue>PAYMENT</DefaultValue>

                                         <Identifier>1</Identifier>
                                    </Field>
                                    <Field>
                                         <Name>ID</Name>
                                         <TagName>Identifier</TagName>
                                         <DataType>Alpha</DataType>
                                         <PaddingChar/>
                                         <DecimalLength/>
                                         <Alignment>LEFT</Alignment>
                                         <StartPos>0</StartPos>
                                         <Length>0</Length>
                                         <DefaultValue>TYPE</DefaultValue>
                                         <Identifier>1</Identifier>
                                    </Field>
                                    <Field>
                                         <Name>ID</Name>
                                         <TagName>Identifier</TagName>
                                         <DataType>Alpha</DataType>
                                         <PaddingChar/>
                                         <DecimalLength/>
                                         <Alignment>LEFT</Alignment>
                                         <StartPos>0</StartPos>
                                         <Length>0</Length>
                                         <DefaultValue>CARD</DefaultValue>
                                         <Identifier>1</Identifier>
                                    </Field>
                                    <Field>
                                         <Name>CardType</Name>
                                         <TagName>CardType</TagName>
                                         <DataType>Alpha</DataType>
                                         <PaddingChar/>
                                         <DecimalLength/>
                                         <Alignment>LEFT</Alignment>
                                         <StartPos>0</StartPos>
                                         <Length>0</Length>
                                         <DefaultValue></DefaultValue>
                                         <Identifier>0</Identifier>
                                    </Field>
                                    <Field>
                                         <Name>CardNumber</Name>
                                         <TagName>CardNumber</TagName>
                                         <DataType>Alpha</DataType>
                                         <PaddingChar/>
                                         <DecimalLength/>
                                         <Alignment>LEFT</Alignment>
                                         <StartPos>0</StartPos>
                                         <Length>0</Length>
                                         <DefaultValue></DefaultValue>
                                         <Identifier>0</Identifier>
                                    </Field>
                                </Fields>
```

```
            </Record>
            <Record>
                <Name>PaymentDetails</Name>
                <TagName>PaymentDetails</TagName>
                <MinOccur>0</MinOccur>
                <MaxOccur>1</MaxOccur>
                <Fields>
                    <Field>
                        <Name>ID</Name>
                        <TagName>Identifier</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>0</Length>
                     <DefaultValue>PAYMENT</DefaultValue>

                        <Identifier>1</Identifier>
                    </Field>
                    <Field>
                        <Name>ID</Name>
                        <TagName>Identifier</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>0</Length>
                        <DefaultValue>TYPE</DefaultValue>
                        <Identifier>1</Identifier>
                    </Field>
                    <Field>
                        <Name>ID</Name>
                        <TagName>Identifier</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>0</Length>
                        <DefaultValue>CASH</DefaultValue>
                        <Identifier>1</Identifier>
                    </Field>
                </Fields>
            </Record>
        </Records>
    </Group>
    <Record>
        <Name>Date</Name>
        <TagName>Date</TagName>
        <MinOccur>1</MinOccur>
        <MaxOccur>1</MaxOccur>
        <Fields>
            <Field>
```

```
                        <Name>ID</Name>
                        <TagName>Identifier</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>0</Length>
                        <DefaultValue>DATE</DefaultValue>
                        <Identifier>1</Identifier>
                    </Field>
                    <Field>
                        <Name>Day</Name>
                        <TagName>Day</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>0</Length>
                        <DefaultValue></DefaultValue>
                        <Identifier>0</Identifier>
                    </Field>
                    <Field>
                        <Name>Month</Name>
                        <TagName>Month</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>0</Length>
                        <DefaultValue></DefaultValue>
                        <Identifier>0</Identifier>
                    </Field>
                    <Field>
                        <Name>Year</Name>
                        <TagName>Year</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>0</Length>
                        <DefaultValue></DefaultValue>
                        <Identifier>0</Identifier>
                    </Field>
                </Fields>
            </Record>
        </Records>
        <Group>
            <Name>Items</Name>
            <TagName>Items</TagName>
            <MinOccur>1</MinOccur>
            <MaxOccur>1</MaxOccur>
```

```
<Records>
    <Group>
        <Name>Item</Name>
        <TagName>Item</TagName>
        <MinOccur>1</MinOccur>
        <MaxOccur>U</MaxOccur>
        <Records>
            <Record>
                <Name>ItemDetails</Name>
                <TagName>ItemDetails</TagName>
                <MinOccur>1</MinOccur>
                <MaxOccur>1</MaxOccur>
                <Fields>
                    <Field>
                        <Name>ID</Name>
                        <TagName>Identifier</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>0</Length>
                       <DefaultValue>ITEM</DefaultValue>

                        <Identifier>1</Identifier>
                    </Field>
                    <Field>
                        <Name>Name</Name>
                        <TagName>Name</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>0</Length>
                        <DefaultValue></DefaultValue>
                        <Identifier>0</Identifier>
                    </Field>
                    <Field>
                        <Name>Quantity</Name>
                        <TagName>Quantity</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>0</Length>
                        <DefaultValue></DefaultValue>
                        <Identifier>0</Identifier>
                    </Field>
                    <Field>
                        <Name>Price</Name>
                        <TagName>Price</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
```

```
                                               <DecimalLength/>
                                               <Alignment>LEFT</Alignment>
                                               <StartPos>0</StartPos>
                                               <Length>0</Length>
                                               <DefaultValue></DefaultValue>
                                               <Identifier>0</Identifier>
                                          </Field>
                                     </Fields>
                                </Record>
                           </Records>
                      </Group>
                  </Records>
               </Group>
           </Group>
        </Records>
   </Message>
  </Messages>
 </File>
</Files>
</BankRepository>
```

Input file:

```
CUSTOMER:Jonas:0123456:
PAYMENT:TYPE:CARD:VISA:98765432:
DATE:09:10:2002:
ITEM:Hub cap:4:199.00:
ITEM:Tyre:4:679.00:
ITEM:Fuzzy dice:1:37.50:
CUSTOMER:Magnus:1234567:
PAYMENT:TYPE:CASH:
DATE:10:10:2002:
ITEM:SAAB Oil pump:1:1229.00:
ITEM:SAAB Oil filter:1:56.00:
```

The expected result will be:

```
<PurchasedItems>
 <Purchase>
  <CustomerDetails>
   <Identifier>CUSTOMER</Identifier>
   <Name>Jonas</Name>
   <Number>0123456</Number>
  </CustomerDetails>
  <Payment>
   <PaymentDetails>
    <Identifier>PAYMENT</Identifier>
    <Identifier>TYPE</Identifier>
    <Identifier>CARD</Identifier>
    <CardType>VISA</CardType>
    <CardNumber>98765432</CardNumber>
```

```
   </PaymentDetails>
  </Payment>
  <Date>
   <Identifier>DATE</Identifier>
   <Day>09</Day>
   <Month>10</Month>
   <Year>2002</Year>
  </Date>
  <Items>
   <Item>
    <ItemDetails>
     <Identifier>ITEM</Identifier>
     <Name>Hub cap</Name>
     <Quantity>4</Quantity>
     <Price>199.00</Price>
    </ItemDetails>
   </Item>
   <Item>
    <ItemDetails>
     <Identifier>ITEM</Identifier>
     <Name>Tyre</Name>
     <Quantity>4</Quantity>
     <Price>679.00</Price>
    </ItemDetails>
   </Item>
   <Item>
    <ItemDetails>
     <Identifier>ITEM</Identifier>
     <Name>Fuzzy dice</Name>
     <Quantity>1</Quantity>
     <Price>37.50</Price>
    </ItemDetails>
   </Item>
  </Items>
 </Purchase>
 <Purchase>
  <CustomerDetails>
   <Identifier>CUSTOMER</Identifier>
   <Name>Magnus</Name>
   <Number>1234567</Number>
  </CustomerDetails>
  <Payment>
   <PaymentDetails>
    <Identifier>PAYMENT</Identifier>
    <Identifier>TYPE</Identifier>
    <Identifier>CASH</Identifier>
   </PaymentDetails>
  </Payment>
  <Date>
   <Identifier>DATE</Identifier>
   <Day>10</Day>
   <Month>10</Month>
   <Year>2002</Year>
  </Date>
  <Items>
```

```
   <Item>
    <ItemDetails>
     <Identifier>ITEM</Identifier>
     <Name>SAAB Oil pump</Name>
     <Quantity>1</Quantity>
     <Price>1229.00</Price>
    </ItemDetails>
   </Item>
   <Item>
    <ItemDetails>
     <Identifier>ITEM</Identifier>
     <Name>SAAB Oil filter</Name>
     <Quantity>1</Quantity>
     <Price>56.00</Price></ItemDetails>
   </Item>
  </Items>
 </Purchase>
</PurchasedItems>
```

Example 13 B

This example shows how a field and record-separated flat file without identifiers is transformed to XML.

Link Flat File Definition:

```
<BankRepository>
 <Files>
  <File>
   <ID>SampleThirteenB</ID>
   <Messages>
    <Message><IgnorePrecedingFieldSeparator>1</IgnorePrecedingFieldSepara
tor>
            <DefaultEmptyFieldValue><Numeric>0</Numeric><Alpha></Al
pha></DefaultEmptyFieldValue>
            <DefaultAlignment><Numeric>RIGHT</Numeric><Alpha>LEFT</Al
pha></DefaultAlignment>
          <DefaultPadding><Numeric>0</Numeric><Alpha> </Alpha></Default
Padding>
            <DefaultTrim><Numeric>1</Numeric><Alpha>0</Alpha></Default
Trim>
     <Name>TopNode</Name>
     <TagName>TopNode</TagName>


     <EscapeChar/>
     <RecordSeparator>crlf</RecordSeparator>
     <FieldSeparator></FieldSeparator>
     <Records>
      <Group>
       <Name>People</Name>
       <TagName>People</TagName>
       <MinOccur>1</MinOccur>
```

```
<MaxOccur>1</MaxOccur>
<Records>
 <Group>
  <Name>Person</Name>
  <TagName>Person</TagName>
  <MinOccur>1</MinOccur>
  <MaxOccur>0</MaxOccur>
  <Records>
   <Record>
    <Name>PersonDetails</Name>
    <TagName>PersonDetails</TagName>
    <MinOccur>1</MinOccur>
    <MaxOccur>1</MaxOccur>
    <Fields>
     <Field>
      <Name>FirstName</Name>
      <TagName>FirstName</TagName>
      <DataType>Alpha</DataType>
      <PaddingChar/>
      <DecimalLength/>
      <Alignment>LEFT</Alignment>
      <StartPos>1</StartPos>
      <Length>10</Length>
      <DefaultValue></DefaultValue>
      <Identifier>0</Identifier>
     </Field>
     <Field>
      <Name>LastName</Name>
      <TagName>LastName</TagName>
      <DataType>Alpha</DataType>
      <PaddingChar/>
      <DecimalLength/>
      <Alignment>LEFT</Alignment>
      <StartPos>11</StartPos>
      <Length>10</Length>
      <DefaultValue></DefaultValue>
      <Identifier>0</Identifier>
     </Field>
     <Field>
      <Name>Organization</Name>
      <TagName>Organization</TagName>
      <DataType>Alpha</DataType>
      <PaddingChar/>
      <DecimalLength/>
      <Alignment>LEFT</Alignment>
      <StartPos>21</StartPos>
      <Length>10</Length>
      <DefaultValue></DefaultValue>
      <Identifier>0</Identifier>
     </Field>
     <Field>
      <Name>E-Mail</Name>
      <TagName>EMail</TagName>
      <DataType>Alpha</DataType>
      <PaddingChar/>
```

```
                    <DecimalLength/>
                    <Alignment>LEFT</Alignment>
                    <StartPos>31</StartPos>
                    <Length>40</Length>
                    <DefaultValue></DefaultValue>
                    <Identifier>0</Identifier>
                  </Field>
                </Fields>
              </Record>
              <Group>
                <Name>Address</Name>
                <TagName>Address</TagName>
                <MinOccur>0</MinOccur>
                <MaxOccur>1</MaxOccur>
                <Records>
                  <Record>
                    <Name>AddressLine</Name>
                    <TagName>AddressLine</TagName>
                    <MinOccur>1</MinOccur>
                    <MaxOccur>0</MaxOccur>
                    <Fields>
                      <Field>
                        <Name>Text</Name>
                        <TagName>Text</TagName>
                        <DataType>Alpha</DataType>
                        <PaddingChar/>
                        <DecimalLength/>
                        <Alignment>LEFT</Alignment>
                        <StartPos>0</StartPos>
                        <Length>40</Length>
                        <DefaultValue></DefaultValue>
                        <Identifier>0</Identifier>
                      </Field>
                    </Fields>
                  </Record>
                </Records>
              </Group>
            </Records>
          </Group>
        </Records>
      </Group>
    </Records>
  </Message>
  </Messages>
  </File>
 </Files>
</BankRepository>
```

The input is a field and record-separated file without identifiers:

```
12345678901234567890123456789012345678901234567890123456789012345678901234567890
Jonas Fügedi IRD jonas.fugedi@infor.se
Vendevägen 89
```

```
181 82 Danderyd
Joakim Hemligt HEM joakim.hemligt@infor.se
Vendevägen 89
Magnus Persson ICS magnus.persson@infor.se
Vendevägen 89
```

The expected result will be:

```
<TopNode>
 <People>
  <Person>
   <PersonDetails>
    <FirstName>1234567890</FirstName>
    <LastName>1234567890</LastName>
    <Organization>1234567890</Organization>
    <EMail>12345678901234567890123456789012345 67890</EMail>
   </PersonDetails>
  </Person>
  <Person>
   <PersonDetails>
    <FirstName>Jonas</FirstName>
    <LastName>Fügedi</LastName>
    <Organization>IRD</Organization>
    <EMail>jonas.fugedi@intentia.se</EMail>
   </PersonDetails>
   <Address>
    <AddressLine>
     <Text>Vendevägen 89</Text>
    </AddressLine>
    <AddressLine>
     <Text>181 82 Danderyd</Text>
    </AddressLine>
   </Address>
  </Person>
  <Person>
   <PersonDetails>
    <FirstName>Joakim</FirstName>
    <LastName>Hemligt</LastName>
    <Organization>HEM</Organization>
    <EMail>joakim.hemligt@intentia.se</EMail>
   </PersonDetails>
   <Address>
    <AddressLine>
     <Text>Vendevägen 89</Text>
    </AddressLine>
   </Address>
  </Person>
  <Person>
   <PersonDetails>
    <FirstName>Magnus</FirstName>
    <LastName>Persson</LastName>
    <Organization>ICS</Organization>
    <EMail>magnus.persson@intentia.se</EMail>
```

```
    </PersonDetails>
    <Address>
     <AddressLine>
      <Text>Vendevägen 89</Text>
     </AddressLine>
    </Address>
   </Person>
  </People>
</TopNode>
```

Chapter 16: Infor Enterprise Collaborator Agreements

Enterprise Collaborator (EC) Agreements is a component of Infor M3 Administration Tools. EC Agreements is used to manage Partner Agreements, which contains the agreement information between you and your partners. Infor Enterprise Collaborator uses the agreement information to send and receive business messages between you and your partners.

To access **Enterprise Collaborator Agreements**, select **Administration Tools > Enterprise Collaborator Agreements** from the Infor M3 Menu in Infor Ming.le.

This table shows the IFS Security roles required for accessing Infor Enterprise Collaborator and Agreements:

IFS role	Definition
M3EC-Administrator	The M3EC-Administrator IFS role grants administrator privileges in Infor Enterprise Collaborator.
M3EC-User	The M3EC-User role is limited and works with M3 data translations from an integration perspective.

Partner Admin tool information model

The information defined in PA tool is called a partner agreement. IEC use this information to send and receive business messages between you and your partners. The agreements are organized according to type and IEC identifies which message is received, from what sending party, how to process the message, which transport media to use, and what routing method to reach the proper partner.

Use the agreements to send (outbound) messages to, and receive (inbound) message from, one or multiple partners. An agreement must be created for every message that is sent (outbound) and received (inbound). For example, you can have several installations of business servers connected to a single IEC and send one message to several different partners. You can also have more than one installation of IEC on a server.

This diagram shows an overview of Infor Enterprise Collaborator in runtime.

- In **Detect** phase, a received message is identified and detected. The detection rules are defined in the PA tool.

- In **Process** phase, a message is processed and executed. IEC will apply the process steps created in the PA tool to execute the agreement.
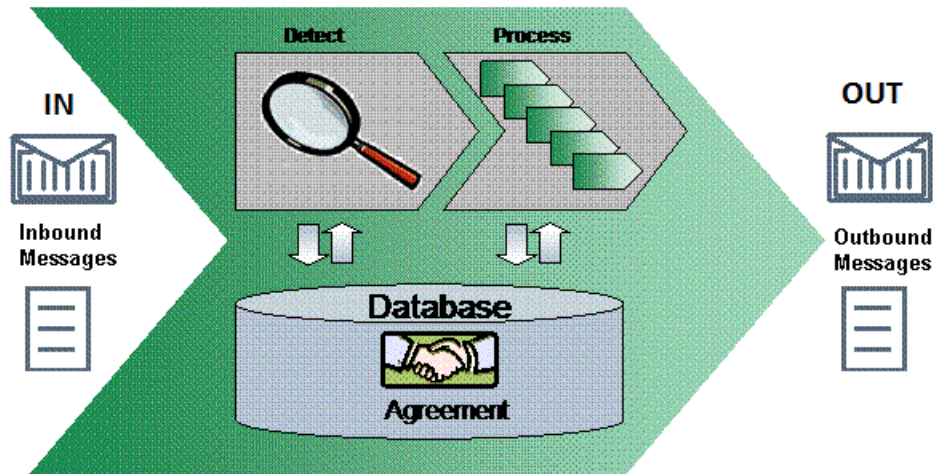
Note:

In IEC versions earlier than 2.0, only XML files are detected by IEC and the conversion of flat file format to XML format is a required process step.

Starting from IEC version 2.0 to the latest, the Flat and XML files are detected and managed in the same way. But, if you will apply a mapping, file transformation to XML is still necessary.

Uses of Partner Admin Tool

- Detections, use this to manage IEC flexible message detection settings.
- XML Mappings, use this to manage published mappings.
- Agreement, use this to set up and manage partners and agreements.
- Error Suppression, use this to suppress errors using regular expression (regex).

Enterprise Collaborator Agreements page

The agreements page has several features. These functions allow you to manage Enterprise Collaborator Agreements.

Use these procedures to create a process flow and manage your partners and agreements in the PA tool.

You can have one folder each for inbound and outbound agreements, or add new folders to set the structure of your partners and agreements. The inbound Test Agreements `Test_100` and `Test_20` are delivered with the package.

Using the Agreements tree folder

Navigate to the Enterprise Collaborator Agreements page to use the agreement tree pane. Use these functions to search, filter, add agreement groups and refresh the sidebar. You can view all of the agreement tree groups and subgroups, in this pane.

Note: You cannot add or remove agreement tree groups.

Button	Function
Search	Use this function to search for agreements. Specify the agreement that you want to search and the agreement tree pane shows the results.
Filter	Use this function to show all agreements or a combination of standard agreements, custom agreements, active agreements or inactive agreements.
Add Group	Use this function to add agreement groups to organize various agreements in the pane.
Refresh	Use this function to refresh the groups and agreements shown in the pane.

Control properties configuration

Set the control properties configuration in the **Agreement folder > Configuration**.

When Company/Division Override or CONO/DIVI is defined in multiple places, it is evaluated in increasing order of priority.

Here is an example sequence:

- User associated CONO/DIVI
- Folder control properties
- Agreement control properties
- Mapping

In this example, the top priority is "Mapping", and the least is "user associated CONO/DIVI". That is, Mapping will override all the other location settings.

Viewing name and description of Agreement tree group

View and edit the name and description of an Agreement tree group by following these steps.

1 In the agreements tree folder click **Agreement > Configuration**.
2 View or specify this information:

   Name
   The name of the agreement tree group.

Description
A brief description of the agreement tree group.

3   Click **Save** if changes were made.

Adding control properties

Add control properties to an agreement group by following these steps.

1   In the agreements tree folder click **Agreement > Configuration**.
2   Click **Configuration**.
3   Click **+** in Control Properties.
4   In the New_Property, click the Name and Value fields to edit.

Name
The name of the control property.

Value
The value contained by the control property.

5   Click the **Save** icon.

Editing control properties

Edit the control properties of an agreement group by following these steps.

1   In the agreements tree folder, navigate to the **Agreement > Configuration** you want to edit.
2   Click the field of the control property you want to edit.

Name
The name of the control property.

Value
The value contained by the control property.

3   Click **Save**.

Deleting control properties

Delete the control properties of an agreement group by following these steps.

Note:

Delete has no undo function.

1   In the agreements tree folder, navigate to the **Agreement > Configuration** you want to edit.
2   Click the check box of the control property you want to delete.
3   Click the **Delete** icon.

4   Click **Save**.

Adding an agreement group

You must add an agreement group before you can add subgroups. Create a group by following these steps in the Configuration root tree.

1   Navigate to the agreement where you want to add the group by clicking **Agreement > Configuration**.

2   Click **Configuration**.

3   Click **Add Group**. The **...** button shows if the bar is not fully displayed.

4   Specify this information:

Name
Specify the name of the agreement group.

Description
Specify the description of the agreement group.

5   Optionally, add any control properties.

6   Click **Save**.

Using and configuring subgroups

Subgroups are agreement groups inside an agreement group. Follow these topics to create and edit subgroups.

Viewing name and description of subgroups

View and edit the name and description of subgroup by following these steps.

1   In the agreements root folder, navigate to the agreement group containing the subgroup.

2   Click the subgroup.

3   Click Configuration.

4   View or specify this information:

Name
The name of the agreement folder.

Description
A brief description of the agreement.

5   Click **Save** if changes were made.

Adding an agreement to a subgroup

Add an agreement in the subgroup's Configuration by following these steps.

Note: Duplicate names are not allowed. Any agreements or groups with the same exact names will not be saved.

1   Click <u>Configuration</u>.

2   Click **+** on the details panel toolbar.

3   In the New_Agreement, specify these fields:

Name
The name of the agreement.

Version
The version of the agreement.

Description
Optionally, add a short description of the new agreement.

Control Properties
Optionally, add any new control properties.

4   Click the **Save** icon.

Moving an agreement

Move an agreement from one parent group to another, by following these steps:

1   Click the agreement you want to move, in the agreement tree.

2   Click the **Move** icon.

3   Select the new parent group to move into.

4   Click **Move**.

Editing a subgroup

Edit an agreement subgroup's properties by following these steps.

Note: Only agreements are editable. Custom BODs and default BODs are not editable.

1   Click the subgroup you want to edit, in the agreement tree.

2   Click **Configuration**.

3   Click the field of the subgroup you want to edit.

Name
The name of the control property.

Description
A short description of the subgroup.

Control Properties
Optionally, add or edit control properties in this table.

4   Click **Save**.

Deleting a subgroup

Delete an existing agreement subgroup by following these steps.

Note: You cannot delete an agreement subgroup if it contains any subgroups. You must first delete the contents of the subgroup before deleting the subgroup.

1   Click the subgroup you want to delete, in the agreement tree.

2   Click **Delete**.

3   Click **Yes**.

Using the Refresh Tree

Click the **Refresh** button on the agreement tree pane, to refresh the tree. The refresh button on the adjacent pane will refresh the agreement information. Use these functions to update the information after saving or editing.

Creating a Partner Agreement

Use this procedure to create new partner agreements.

1   Access the Agreement tree panel.

2   Select the parent folder to contain your new agreement, select **Insert Agreement**.

The parent folder for your agreement is your partner. This folder name is shown as your partner agreement name in EC Administration interface.

3   Access the details panel.

4   For your new agreement, specify this information:

When creating message agreements ensure that all field definitions are completed. If only the "Name" field was provided, an exception error may occur resulting to the message staying in "Detect" state. When this happens, review and complete the agreement field definitions.

Name

Required name for this agreement.

Description

Optionally, type a brief description for this agreement.

Creator

Optionally, specify a name for the agreement author.

Version

Set the version for this agreement.

Priority

Set the numeric priority level for this agreement.

Note: Agreements are prioritized depending on certain parameters.

Setting up a process flow

Use this procedure to set up a process flow to run in IEC.

1   Access the Agreement tree panel and select an agreement to use.

2   Access the Processes tab. In Selected processes pane, select a process step from the list.

3   At the prompts, define the process details and confirm to add.

Viewing Agreements

Use this procedure to view agreements, by Detection and Target Group.

1   In Enterprise Collaborator Agreements, click on an agreement group, in the agreement tree pane.

2   Click on the agreement to view the details of the agreement.

General Information

Open an agreement to view general information, using the tools provided in Enterprise Collaborator Agreements.

The agreement viewing pane contains several buttons for managing agreements:

Button	Function
Refresh	Refreshes the agreement information.
Save	Saves the changes you made to the agreement.
Delete	Deletes the current agreement.
Duplicate	Creates a copy of the current agreement.
Move	Moves the current agreement to another location or agreement group.
Activate	Activates the current agreement.
Stop	Stops the current agreement.

General Information tab shows several fields of agreement information that you can view and edit.

Field	Description
Name	Shows the name of the agreement.
Creator	Shows the name of the user who created the agreement.
Version	Shows the version of the agreement. This is manually specified.

Field	Description
Priority	Shows the agreement priority level.
Description	Shows a brief description of the agreement.
Control Properties	Shows the control properties of the agreement.
Notifications	Shows any agreement notifications.

Detection

Detections allow you to set the flexible message detection of IEC. There are several important objects that make up the detection part. One of which is the Message Detection. It defines the kind of data that IEC can detect. By default, only XML messages can be detected.

Detections contain one or more target groups. Each target group contains one or more targets. A target points out a unique path to the data in the receiving message.

The Detection page allows you to set detection targets.

Field	Description
Detection Type	Shows the type of detection used.
Target Group	Shows the target group of the detection.
Targets	Shows the target agreements. This table shows the name, path and value of the targets.

Uses of Detections

When IEC receives a message, it starts to detect the message according to the detections and target groups in the detection order list. The detection order list is processed from top to bottom. For example, IEC will start to look at the received message and check if it fits the MBM target group in the XML detection. If it does not fit, IEC will check if the message fits the MVX_S0_R0 target group, and so on, until it finds a matching target group for the message. If there is no matching target group, the message fails and the message is set to "detection failed" status.

In the area for Unused Targets, all the detections and the defined target groups for that detection are shown.

In the area for Targets for Selected Groups, all the detections and target groups that you selected from the Unused Target area are shown. IEC use this detection order in the Processes tab and in the Detection step.

Note: When you create target groups and put them in the detection order, the problem of shadowing can arise.

When you create message agreements, you must complete all field definitions. If only the "Name" field was specified, an exception error may occur. This error will result to the message to remain in "Detect" state. If this error occurs, go back to the agreement and complete the field definitions.

Adding a new detection

Add a new agreement detection by using the Detection tab in IEC Agreements. There are several types of detections to choose from.

1 Select the agreement in the agreement tree pane.

2 Click **Detections**.

3 Specify this information:

   Detection Type
   Select a detection type from the drop down menu.

   Target Group
   Select a target group from the menu. The target group determines the targets available for the agreement.

   Target Value
   Specify the target value.

4 Click **Save**.

Executing XML Detection using the MBM Target Group

Use this procedure to execute XML detection using the MBM target group.

1 In the Agreement view pane, select an agreement.

2 In the right pane, click the **Detection** tab.

3 Select Target Group MBM in the Detection tree view.
   The view shows the information for other detection fields.

4 Specify this information:

   Detection/env:from
   Specify your source partner detail.

   Detection/env:to
   Specify your recipient partner detail.

   Detection/env:URI
   Specify the URI for your message and the version. For example, Test_100_in + 1.1 entail #Test_100_in@1.1

5 Save the XML detection.

Executing XML Detection Using the MVX Target Group

Use this procedure to execute XML detection using the MVX target group.

1 On Partner Admin Tool, select an agreement.

2 Click the **Detection** tab.

3   Select the Target Group *MVX_X_Y*.

4   Add the target values, specify this information:

Detection/mvx:env

Add the M3 environment, for example MVXCENV.

Detection/mvx:host

Add the address to the M3 server, for example SEIRD400 or an IP address.

Detection/mvx:pgm

Specify the name of the M3 program that generated the initiator, for example (OIS606PF).

Detection/sndRefData1

Specify the sender reference field 1, for example `CONO`.

Detection/sndRefData2

Specify the sender reference field 2, for example `DIVI`.

Detection/sndRefData3

Specify the sender reference field 3.

Detection/rcpRefData1

Specify the receiver reference field, for example `CUNO`.

Detection/rcpRefData2

Specify the receiver reference field 2.

Detection/rcpRefData3

Specify the receiver reference field 3.

Important

If you leave an empty target value, the agreement is detected.

5   Save the Detection.

Processes

The Processes page shows the all the processes related to the specified agreement.

Process	Description
Move Up	Moves the selected process up.
Move Down	Moves the selected process down.
Edit	Opens the editing page for the selected process.
Delete	Deletes the selected process.

Process	Description
Add (+)	Adds a process. Select a mapping for the process after this step and click **Add**.
Send	Shows Send process groups.
XML Transform	Shows the XML Transform process groups.

Process Groups and Properties	Description
Indicator	Shows an icon for the type of process.
Key	Shows the key type of the process property.
Value	Shows the value of the process.

Selecting a process group will list all of its properties in the Group Details table.

Group Details	Description
Key	Shows the key type of the agreement group.
Value	Shows the value of the agreement group keys.

This table shows a grouped listing of available process steps:

Process steps	Available to error handling	Available to regular process
Apply Envelope	Yes	Yes
FLAT To XML	No	Yes
Remove Envelope	No	Yes
Send	Yes	Yes
XML To Flat	No	Yes
XML Transform	Yes	Yes
XSL Transform	Yes	Yes

Creating an Apply Envelope process

Create an Apply Envelope process to add an envelope to the process.

See Envelopes on page 371 for more information on using envelopes and envelope templates.

1 Click **Processes** tab.
2 Click **+** and select **Apply Envelope**.
3 Select an envelope template.
4 Select an envelope encoding.
5 Click **Add**.

Creating an Apply MBM Envelope process

Creating an Apply MBM Envelope process requires extra fields to be specified.

See Envelopes on page 371 for more information on using envelopes and envelope templates.

1 Click **Processes** tab.

2 Click **+** and select **Apply Envelope**.

3 Select MBM Envelope template.

4 Specify the following information:

Envelope encoding
Specify the type of encoding used by the envelope.

Schema
Specify the schema used by the envelope.

To Address
Specify the destination of the envelope.

From Address
Specify the origin address of the envelope.

URI
Specify the Uniform Resource Identifier used by the envelope.

5 Click **Add**.

Creating a Remove Envelope process

Create a Remove Envelope process to remove an envelope from the process.

1 Click **Processes** tab.

2 Click **+** and select **Remove Envelope**.

3 Specify the Body Element XPath.
Click the **MBM Body XPath** or the **Mvx Body XPath** to add a prefix to the Body Element XPath.

4 Click **Add**.

Creating a Send process

Send is used for sending messages to ION.

1 Click **Processes** tab.

2 Click **+** and select **Send**.

3 Select the Document that was previously defined in the ION Documents UI. Only Documents that are enabled for 'Send' appear in the list.

4 Click **Add**.

Creating an XML Transform process

Use XML Transform to select mappings and set other options for XML Transform.

1   Click **Processes** tab.

2   Click **+** and select XML Transform.

3   Select a mapping for the XML Transform process.

4   Click **Add**.

Creating a FLAT to XML process

Use FLAT to XML, to transform select mappings from a FLAT format, to XML.

Note: You can select either a standard or non-standard definition.

1   Click **Processes** tab.

2   Click **+** and select FLAT to XML.

3   Select an Enabled Flat Definition for the FLAT to XML process.

4   Click **Add**.

Creating an XML to Flat process

Use XML to FLAT, to transform select mappings from an XML format, to FLAT.

Note: You can select either a standard or non-standard definition.

1   Click **Processes** tab.

2   Click **+** and select XML to FLAT.

3   Select an Enabled Flat Definition for the XML to FLAT process.

4   Click **Add**.

Creating an XSL Transform process

XSL Transform is used to translate XML documents into other formats such as HTML, plain text files and XSL formatting objects. These subsequent formats can then be converted into other formats, such as PNG images and PDF files. Use XSL Transform to select mappings and set other options for XSL Transform.

1   Click **Processes** tab.

2   Click **+** and select XSL Transform.

3   Select an XSLT Definition for the XSL Transform process.

4   Specify these XSLT Definition Details:

   Doctype Public
   The visible document type.

Doctype System

The XSL Transform document system.

Encoding

The encoding type used by the XSL Transform.

Method

The XSL Transform method.

Version

The version used by the XSL Transform.

Indent

Select a yes or no, to add indentation to the definition.

Omit XML Declaration

Select a yes or no, to not include XSL declaration.

Standalone

Select a yes or no, to make the XSL Transform independent.

5 Click **Add**.

Error Processes

When an error occurs in a process flow, an error-mail is sent and the message log displays the error details. Use error handling to define a custom process flow on the source data to support basic communication channel or agreement-specific error conditions.

Error handling supports the XML Transform and Send processes. The input to error flow is the IEC error message document.

Creating an Apply Envelope error process

Create an Apply Envelope error process to add an envelope to the error process.

1 Click **Error Processes** tab.
2 Click **+** and select **Apply Envelope**.
3 Select an envelope template.
4 Select an envelope encoding.
5 Click **Add**.

Creating an Apply Envelope MBM error process

Creating an Apply MBM Envelope error process requires extra fields to be specified.

1 Click **Error Processes** tab.
2 Click **+** and select **Apply Envelope**.

3   Select MBM Envelope template.

4   Specify the following information:

> **Envelope encoding**
> Specify the type of encoding used by the envelope.
>
> **Schema**
> Specify the schema used by the envelope.
>
> **To Address**
> Specify the destination of the envelope.
>
> **From Address**
> Specify the origin address of the envelope.
>
> **URI**
> Specify the Uniform Resource Identifier used by the envelope.

5   Click **Add**.

Creating a Send error process

Send error process is used for sending error message documents, when an error takes place during a Send process.

1   Click **Error Processes** tab.

2   Click **+** and select **Send**.

3   Select the Document that was previously defined in the ION Documents UI. Only Documents that are enabled for 'Send' appear in the list.

4   Click **Add**.

Creating an XML Transform error process

XML Transform error process is used for sending error message documents, when an error takes place during an XML Transform error process.

1   Click **Error Processes** tab.

2   Click **+** and select **XML Transform**.

3   Select a mapping for the XML Transform process.

4   Click **Add**.

Creating an XSL Transform error process

XSL Transform error process is used for sending error message documents, when an error takes place during an XSL Transform error process.

1   Click **Error Processes** tab.

2 Click **+** and select XSL Transform.

3 Select an XSLT Definition for the XSL Transform process.

4 Specify these XSLT Definition Details:

Doctype Public
The visible document type.

Doctype System
The XSL Transform error document system.

Encoding
The encoding type used by the XSL Transform error.

Method
The XSL Transform error method.

Version
The version used by the XSL Transform error process.

Indent
Select a yes or no, to add indentation to the definition.

Omit XML Declaration
Select a yes or no, to not include XSL declaration.

Standalone
Select a yes or no, to make the XSL Transform error, independent.

5 Click **Add**.

Deleting agreements or agreement groups

Multiple deletions are not supported in PAT UI. Agreement subgroup or agreements should deleted one at a time.

Note: Delete has no undo and backup functions.

Deleting an agreement group

Follow these steps to delete an agreement group.

1 Click the **Agreement View Tab** and select the folder containing the agreement groups to delete.

2 Select **Delete group**.

3 Click **Yes** at the delete confirmation prompts.

Process steps

You add process steps to create a complete process flow for messages in existing agreements. IEC executes the process flow based on the order of included process steps.

Here are the message-handling process flow types:

- Regular process steps handle regular message process flow.

- Error-handling process steps handle errors encountered in regular process flow.

- Process steps applicable to both regular and error-handling errors in both process steps.

Regular process steps

Use regular process steps in an already existing agreement to complete the process flow of messages. You add the process steps from the context menu on the Processes tab located in the right pane of the Agreement tree. Save the agreement after you add process steps for changes to take effect.

Error handling process steps

Define specific error handling process steps in an already existing agreement to handle particular errors encountered in regular process flow. You add the process steps from the context menu on the Error Handling tab located in the right pane of the Agreement tree. Save the agreement after you add error handling process steps.

For Agreements without specific error process tied to a particular step, IEC server runs the default agreement-level handling process step.

IEC server error handling has priority on specific error processes tied to a particular process step over agreement-level error handling process step.

Error Handling

When an error occurs in a process flow, an error-mail is sent and the message log displays the error details. You can use error handling to define a custom process flow on the source data to support basic communication channel or agreement-specific error conditions.

Error handling supports the XML Transform and Send processes. The input to error flow is the IEC error message document.

Process Levels

You can set up error handling in any of the following process levels.

Pre-detection level

On pre-detection level, the error flow is defined from the point when a message is received until it is sent for detection. Detection failures are also covered on this level.

Use XML Transform and Send processes.

Agreement level

On Agreement Level, the error flow is defined when an agreement is identified from detection point onwards, until the process is completed. If there are remaining errors, Reprocess and Retry will rerun the error flow. An additional value in this level is the SyncIn channel where the actual message state is sent to the message sender. Then, you can send a custom error handling that will be sent back to the message sender.

Handling Agreement errors

Use these procedures for advanced error handling on detection, agreement, and process levels.

Handling errors in agreement level

Follow these steps to resolve agreement level errors.

1   Access **Agreement Tree** and select an agreement.
2   Access Error Processes to define the custom error flow.
3   Add from a list of available error processes.
4   On **Selected Processes** dialog box, edit or delete processes for this agreement.

Handling errors in process level

You must select an agreement to handle errors in process level.

1   Select an agreement in the agreement tree.
2   Select an error process.
3   Edit or delete the error processes.
4   Click **Save**.
5   Optionally, click **Refresh** to update the data.

Using IMS parameters

IEC sends messages through ION in two parts, the first is the payload, which is the XML file and the second are the parameters. IMS parameters are information elements that give instruction to ION on what to do with these messages. Other parameters allow for specifying preferred values.

When you send messages through ION, they have default mandatory IMS parameters. For example, one of these default parameters is the document name. You can edit these IMS parameters in **ION Desk > Connection Points or Mappings**.

See IMS parameters and manifest items on page 396 for the list of manifest items you can use as IMS parameters.

Chapter 17: Infor Enterprise Collaborator Configuration

Infor Enterprise Collaborator (IEC) Configuration is a component of Infor M3 Administration Tools. IEC Configuration is used to manage subscriptions.

To access **Enterprise Collaborator Configuration**, select **Administration Tools > Enterprise Collaborator Configuration** from the Infor M3 Menu in Infor Ming.le.

This table shows the IFS Security roles required for accessing Infor Enterprise Collaborator and Configuration:

IFS role	Definition
M3EC-Administrator	The M3EC-Administrator IFS role grants administrator privileges in Infor Enterprise Collaborator.
M3EC-User	The M3EC-User role is limited and works with M3 data translations from an integration perspective.

Event Hub Subscriptions

The Event Hub is a generic Grid extension for sending events between Infor applications. It is a publisher-subscriber application framework that allows publisher-applications to expose historical data to subscriber-applications that are interested in receiving this data.

EC can act as a subscriber application through the MECEventHubSubscriber channel.

Note: When you subscribe to MECEventHubSubscriber channels, ensure that Event Hub and EC are running on the same grid and that each channel has one subscription, at the least.

Definition of terms:

- **Event**- A discreet unit of historical data shared by an application that may be relevant to other applications or "Subscriber".
- **Subscription**- A predicate indicating that a subscriber is to receive a particular event.

Adding a subscription

Use this procedure to add a subscription for the MECEventHubSubcriber channel.

Note: After you create a subscription, edit and then assign the subscription to a MECEventHubSubcriberchannel.

1   Click **+** in **Event Hub Subscriptions**.
2   Specify this information:

Subscription

Specify a predicate indicating that a subscriber is to receive a particular event.

Use this subscription format:

<Publisher>`":"`*<DocumentName>*`[":"[`*<operation char>*`][":"[":"[]]]]`

For example: `M3:MITMAS:C::P1`

Note: One or more characters can be given for operation characters. For example, "M3:ITMAS:CU".

Variable	Definition and example
`<operation      char>`	`:C|U|D|S|X|F|Q|R`

Description
Specify a brief description.

3   Click **Save**.

Channel Assignments panel

The Channel Assignments panel shows a summary of subscription items assigned to an event hub channel. You can assign and unassign subscription items to an event hub channel. You can also click a table header to sort the data in the panel.

Use these procedures to manage the Channel Assignments panel.

Editing a subscription

Use this procedure to modify the details of a MECEventHubSubscriber channel.

1   Select the check box of the subscription you want to edit.
2   Click the edit icon to edit the subscription information.
3   Edit the subscription information.
4   Click **Save**.

Deleting a subscription

Use this procedure to delete a MECEventHubSubscriber channel.

Note: You can not delete subscriptions that are in Assigned status.

1   Select the check box of the subscription you want to delete.

2   Click the delete icon to delete the subscription information.

3   Click **Yes**.

Documents

Use the Enterprise Collaborator Configuration Documents page to define which documents are sent to ION from IEC, or received from ION.

Note: The changes to the Documents page are not finalized until you Refresh the ION Connection Point.

Accessing Documents page

You can access the ION **Documents** page or tab by clicking the **Documents** tab in Enterprise Collaborator Configuration. Access it from Ming.le by following these steps:

1   Login Infor Ming.le.

2   Click **Infor M3** from the Ming.le menu.

3   Click **Infor M3**.

4   Click Administration Tools.

5   Click Enterprise Collaborator Configuration.

6   Click **Documents**.

Using the Documents UI and features

Use the basic features to add, select, edit and delete documents.

Button	Function
Refresh	Use this function to refresh the documents list shown in the page.
Add	Use this function to add or create a new document.

Button	Function
Edit	Use this function to edit a selected document. Click the tick box of the document to select it and click edit.
Delete	Use this function to delete the selected document. Click the tick box of the document to select it and click delete.

The documents page shows the full list of documents in a table format. Each column is defined as:

Column	Function
Select (-)	Use this column to select the specific document. Tick the box to select the document.
Document	Shows the name of the document. You can search for a document by specifying the text in the search field.
Description	Shows a brief description of the document. You can search for a document by specifying the text in the search field.
Type	Shows the type or category of the document. You can search for a document by specifying the text in the search field.
Receive	If the Receive tick box is checked, IEC will receive these documents from ION.
Send	If the Send tick box is checked, IEC will send these documents to ION.

Each document has a receive and send tick box. If receive is checked for a document, the document can be sent by ION to IEC through the IMSIn Channel. If send is checked, a document can be sent by IEC going to ION. Both tick boxes can be checked simultaneously.

Creating a document

You can create new ION documents using Documents in IEC Configuration.

1   Click **+**.
2   Specify this information in the Create Documents window:

Name
The name of the new document. Use this to search for the document.

Description
Shows a brief description of the document.

Type
Select the type of document in the drop down options.

Receive
If Receive is selected, IEC will receive these documents from ION.

Send
If Send is selected, IEC will send these documents to ION.

3   Click <u>Save</u> to create the document.

Updating a document

You can edit existing documents in ION Documents.

1   Check the selection box of the document for editing.
2   Click the **Edit** button.
3   Edit any of this information in the Update document window:

Description
Shows a brief description of the document.

Type
Select the type of document in the drop down options.

Receive
If Receive is selected, IEC will receive these documents from ION.

Send
If Send is selected, IEC will send these documents to ION.

4   Click <u>Save</u> to make the changes.

Deleting a document

You can delete existing documents in ION Documents. You cannot delete documents if the Receive and Send boxes are checked. Uncheck these boxes using edit documents, before deleting.

Note: If the Send box is checked during deletion, IEC will check if an Agreement is using it. If yes, IEC will not proceed with the deletion, otherwise, deletion will proceed.

1   Check the selection box of the document for deleting.
2   Click the **Delete** button.
3   Click <u>Yes</u> to delete.

Envelopes

Use envelopes to create a text file where system and editable values are defined by start and end separators. An example of a created text file is XML. You can create any message envelope using the text file as a template. User-defined IEC envelopes are managed here.

Adding a standard envelope

Follow these steps in IEC Configuration to add standard envelopes. You can click **Refresh** to update the list of standard envelopes.

Note: You can not make changes to standard envelopes. You must use custom envelopes to make changes to the definitions.

1   Go to **Infor Enterprise Collaborator Configuration > Envelopes**.

2   Select a standard envelope.

Adding a custom envelope

Use IEC Configuration to add custom envelopes. You can click **Refresh** to update the list of custom envelopes.

Note: You can only edit custom envelopes. Standard envelopes can not be edited.

1   Go to **Infor Enterprise Collaborator Configuration > Envelopes**.

2   Click Add (**+**).

3   Specify these Definition Details:

Name
The name of the new custom envelope.

Description
Shows a brief description of the custom envelope.

Encoding
Specify your encoding for the new custom envelope.

Envelope File
Add an envelope file for your custom envelope.

4   Click **Save**.

Viewing an envelope

View envelope templates used by IEC Configuration by following these steps:

1   Go to **Infor Enterprise Collaborator Configuration > Envelopes**.

2   Select an envelope.

3   Click **View Envelope**.
    The envelope will be viewed as a template.

Exporting an envelope

Export envelope files and download them, by following these steps:

1   Go to **Infor Enterprise Collaborator Configuration > Envelopes**.

2   Select an envelope.

3   Click **Export Envelope**.
    The envelope will be downloaded in your local machine.

Adding an XML Declaration

Add XML Declarations using Envelopes, by following these steps:

1   Go to **Infor Enterprise Collaborator Configuration > Envelopes**.

2   Click XML Declaration.

3   Click Add (**+**).

4   Specify these XML Declaration details:

    Name
    The name of the new XML Declaration.

    Description
    Shows a brief description of the XML Declaration.

    Encoding
    Specify your encoding for the new XML Declaration.

    Envelope File
    Add an envelope file for your XML Declaration.

5   Click **Save**.

XSLT Definitions

XSLT or Extensible Stylesheet Language Transformations, is a computer language that transforms XML documents into other XML documents. It is also used to translate XML documents into other

formats such as HTML, plain text files and XSL formatting objects. These subsequent formats can then be converted into other formats, such as PNG images and PDF files.

Use **XSLT Definitions** in Infor Enterprise Collaborator Configuration, to manage and translate XML documents being sent into Partner Administration Tool, to other formats. To access XSLT Definitions, go to **Infor Enterprise Collaborator Configuration > XSLT Definitions**.

Adding a standard XSLT definition

Follow these steps in IEC Configuration to add standard XSLT definitions. You can click **Refresh** to update the list of standard definitions.

Note: You can not make changes to standard XSLT definitions. You must use custom XSLT definitions to make changes to the definitions.

1   Go to **Infor Enterprise Collaborator Configuration > XSLT Definitions**.

2   Select a standard XSLT definition.

3   Click Add (**+**).

Adding a custom XSLT Definition

Use IEC Configuration to add standard and custom XSLT definitions. You can click **Refresh** to update the list of custom definitions.

Note: You can only edit custom definitions. Standard definitions can not be edited.

1   Go to **Infor Enterprise Collaborator Configuration > XSLT Definitions**.

2   Click Add (**+**).

3   Specify these Definition Details:

Name
The name of the new custom definition.

Description
Shows a brief description of the custom XSLT definition.

Definition File
Specify your definition file for the new custom XSLT definition.

4   Click **Save**.

Viewing an XSLT definition

View definition files used by XSLT definitions by following these steps:

1   Go to **Infor Enterprise Collaborator Configuration > XSLT Definitions**.

2   Select an XSLT definition.

3   Click **View Definition**.
The definition will be viewed as a template.

Exporting an XSLT definition

Export definition files and download them as XML files, by following these steps:

1   Go to **Infor Enterprise Collaborator Configuration > XSLT Definitions**.

2   Select an XSLT definition.

3   Click **Export Definition**.
The definition will be downloaded in your local machine as an XML file.

Deleting an XSLT definition

Delete existing XSLT definitions by following these steps:

1   Go to **Infor Enterprise Collaborator Configuration > XSLT Definitions**.

2   Select an XSLT definition.

3   Click **Delete Definition**.
The definition will be deleted and the definitions list refreshed.

String Replacer

A string replacer defines the strings to be replaced in mapped agreements via it's map name. When a map name used by an agreement matches the map name in the string replacer, the characters defined will be replaced for output.

Adding a standard string replacer

Follow these steps in IEC Configuration to add a standard string replacer. You can click **Refresh** to update the list of standard string replacers.

Note: You can not add or make changes to standard string replacers. Standard string replacers can only be added in the content package. You must use custom string replacers to make changes to the definitions.

1   Go to **Infor Enterprise Collaborator Configuration > String Replacer**.

2   Select a standard string replacer.

Adding a custom string replacer

Use IEC Configuration to add custom string replacers. These string replacers automatically reload. You can click **Refresh** to update the list of custom string replacers.

Note: You can only edit custom string replacers. Standard string replacers can not be edited.

1 Go to **Infor Enterprise Collaborator Configuration > String Replacer**.
2 Click Add (**+**).
3 Specify these Definition Details:

   Name
   The name of the new custom string replacer.

   Description
   Shows a brief description of the custom string replacer.

   Definition file
   Add a definition file for your custom string replacer.

4 Click **Save**.

Viewing a string replacer

View existing string replacer templates used by IEC Configuration, by following these steps:

1 Go to **Infor Enterprise Collaborator Configuration > String Replacer**.
2 Select a string replacer.
3 Click **View Definition**.
   The string replacer will be viewed as a template.

Exporting a string replacer

Export string replacer files and download them, by following these steps:

1 Go to **Infor Enterprise Collaborator Configuration > String Replacer**.
2 Select a string replacer.
3 Click **Export Definition**.
   The string replacer will be downloaded in your local machine.

Chapter 18: Troubleshooting

Use troubleshooting tools that are available in M3 CE Core.

Managing Business Engine jobs

Logging and profiling for a job or viewing active jobs can be useful for troubleshooting. The option to delete a job can be used if, for some reason, there is a need to stop a job that has a problem. The user must be connected to the role *M3BE-FndUser* or *M3BE-FndAdmin* to access **Administration Tools > Business Engine Jobs** from the Infor M3 menu in Infor Ming.le.

Viewing and killing jobs

The role *M3BE-FndAdmin* is required for viewing and deleting jobs for other users. Users with the role *M3BE-FndUser* can view and kill their own jobs.

1   Click **Administration Tools > Business Engine Jobs**.
2   Filter by Name, Type, Configuration used, Owner, Status, Change, CPU %, Uptime or State to search for the job to view.
3   Click the **Refresh** icon to refresh the view.
4   Right-click or mark a job from the list and select **Kill job** to delete the job.

Viewing and configuring logging

The role *M3BE-FndAdmin* is required for configure logging and log levels for jobs for other users. Users with the role *M3BE-FndUser* can view their own jobs and job logs but require the role *M3BE-LogEnabler* to also configure logging and log levels.

1   Click **Administration Tools > Business Engine Jobs**.
2   Right-click or mark a job from the list and select **Logging > Configure Logging** to edit, add, or delete log configuration for the job.
    a   Under Log levels, select the levels to enable logging for.

    b   Under Components, select the components to enable logging for.

    c   Under Concepts, select Enable Concepts to enable concept logging and use the drop-down to select the concept names to enable.

    d   Under Other Settings, use **Backlog size** to select the number of log entries to keep in the backlog, or select **Log to file** to save all log entries to a file on the host where the job is running. The file will contain all log entries, not only the backlog, and will be prefixed by "JOBLOG-". It can be downloaded on the **Business Engine Logs** page.

    e   Click **Save** to save the log configuration or click **Delete** to delete an existing log configuration.
        Note: A log configuration will be active for 4 hours and will then be automatically removed from the job.

3   Right-click or mark a job that has logging configured and select **Logging > View log** to view the backlog of the job.

    a   Optionally, you can disable the **Refresh automatically** check box to stop refreshing the backlog while viewing it.

    b   Filter by Time, Level, or Message to display specific log entries.

    c   Click **Export** to download the displayed log entries as text file.

Configure future logging

The role *M3BE-FndAdmin* is required for configuring future logging for jobs for other users. Users with the role *M3BE-FndUser* can view their own jobs but require the role *M3BE-LogEnabler* to configure future logging.

Future logging can be set up to automatically start logging on new jobs that are started. Any future log configuration will be active for at most 12 hours before it is automatically removed.

1   Click **Administration Tools > Business Engine Jobs**.

2   To configure future logging, select **Logging > Configure future logging**.

3   Right-click or mark a future log configuration and select **Remove** to remove it.

4   Select **Add** to create, right-click or mark a future log configuration and select **Edit** to update an existing future log configuration.

    a   Specify the owner of the job.
        Note: This field is only editable if the user has the role *M3BE-FndAdmin.*

    b   Specify the name of the job.

    c   Specify how many jobs the log configuration should be activated for. Maximum is 5.

    d   Under Log levels, select the levels to enable logging for.

    e   Under Components, select the components to enable logging for.

    f   Under Concepts, select Enable Concepts to enable concept logging and use the drop-down to select the concept names to enable.

    g   Under Other Settings, use **Backlog size** to select the number of log entries to keep in the backlog, or select **Log to file** to save all log entries to a file on the host where the job is running. The file will contain all log entries, not only the backlog, and will be prefixed by "JOBLOG-". It can be downloaded on the Business Engine Logs page.

5   Click **Save** to save the future log configuration or **Close** to discard changes.

Enabling and disabling profiling

The role *M3BE-FndAdmin* is required for enabling and disabling profiling for jobs for other users. Users with the role *M3BE-FndUser* can view their own jobs and if profiling is enabled or not but require the role *M3BE-LogEnabler* to also enable and disable profiling.

When profiling has been disabled or a job with profiling enabled is killed, or a job with profiling enabled has been stopped/closed, the profiler results are stored in a CSV file prefixed by "JOBPROFILER-". It can be downloaded on the **Business Engine Logs** page.

1   Click **Administration Tools > Business Engine Jobs**.
2   Right-click or mark a job from the list and select **Profiling > Enable profiler** to start the profiler on the job that does not have profiling activated.
3   Right-click or mark a job and select **Profiling > Disable profiler** to stop the profiler on a job that has profiling activated.
    Note: When initiated, the profiler logging for a job will run until the job is closed or **Disable profiler** is selected.

Configuring future profiling

The role *M3BE-FndAdmin* is required for configuring future profiling for jobs for other users. Users with the role *M3BE-FndUser* can view their own jobs but require the role *M3BE-LogEnabler* to configure future profiling.

Future profiling can be set up to automatically start profiling on new jobs that are started. Any future profiler configured will be active for at most 60 minutes before it is automatically removed.

1   Click **Administration Tools > Business Engine Jobs**.
2   To configure future profilers, select **Profiling > Configure future profiling**.
3   Right-click or mark a future profiler and select **Remove** to remove it.
4   Select **Add** to create or update an existing future profiling configuration.
    a   Specify the owner of the job.
        Note: This field is only editable if the user has the role *M3BE-FndAdmin*.
    b   Specify the name of the job.
    c   Specify how many jobs profiling should be activated. Maximum is 5.
    d   Click **Save** to save the future profiler configuration or **Close** to discard changes.

Managing Business Engine logs

Viewing and downloading logs can be useful when troubleshooting problems in an M3 Business Engine application. The user must be connected to the role *M3BE-FndUser* or *M3BE-FndAdmin* to access

Administration Tools > Business Engine Logs from the Infor M3 menu in Infor Ming.le. The role *M3BE-FndAdmin* is required for viewing logs for other users.

1 Click **Administration Tools > Business Engine Logs**.

2 Filter by File name, Last modified, or Size to find the log that you wish to view.

3 Mark the file from the list and select the download icon at the top left to download the file.

To view the content of the file, open it in a text editor, for example Notepad.

M3 Cloud File Transfer Client troubleshooting

If there are problems with the printouts being printed by Exstream, or other problems with the files on your local server, the log files can be used to find the causes.

1 Start the Grid Admin UI by running `C:\Infor\InforIONGrid\bin\AdminUI.cmd` as administrator.

2 Locate the CloudFileTransferClient node and click the log text file icon to open the Log file.

Click **Log Levels** and set the log level to debug for M3CFTClient.

All logs generated from this point forward will contain detailed information.

Certificate error

There could be a problem with the certificate between your local server and the Cloud.

1 Locate the CloudFileTransferClient node and click the log text file icon to open the Log file.

In the Find window, search for an entry similar to this example:

```
2015-08-12 15:17:13,442 +0200Z {} ERROR Module com.infor.cft.on
premise.filetransfer.MessagePoller: Failed to start sweeper.
Client can not be created com.infor.cft.common.util.CFTException: Failed
to load keystore content
```

2 In the Grid Admin UI, click the link Application CloudFileTransferClient.

3 Click Configuration and then Edit Properties.

4 Check the Keystore location.

5 Verify that the keystore file `clientcert_FileTransferClient.ks` is found in this location.

If the file cannot be found, you must request a new certificate from Infor Support.

If the file is found, continue to check the password according to the next procedure.

Certificate expired

The certificate is renewed automatically and if the process fails, you can get a new certificate by downloading a new Client installation package.

1   Download a new CFT Client Package from the Cloud File Transfer Agent Administration - Download Client page.

    See "Downloading the M3 Cloud File Transfer Client Package" in the *Infor M3 Cloud Configuration Guide*.

2   Extract the files in the `cftinstaller.zip`.

    The `ctfinstaller.zip` contains these files:

    *   `ctfinstaller.jar`
    *   `clientcert_FileTransferClient.ks`
    *   `FileTransferClient.properties`
    *   `gridinstaller.jar`
    *   `FileTransferClient.gar`

3   Go to the keystore folder, usually located at `C:\Infor\`*`<grid installation>`*`\grid\appli cations\CloudFileTransferClient\appdata\keystore`

4   Rename the `clientcert_FileTransferClient.ks` to client `clientcert_FileTransfer Client_old.ks`.

5   Copy the extracted `clientcert_FileTransferClient.ks` to the keystore folder.

6   Open the FileTransferClient.properties file and copy the *keystore.password* value.

7   Open properties for CloudFileTransferClient and select **Keystore Password**.

8   Click **Edit** and insert the new password.

9   Click **Save**.

Password error

There could be a problem with the password being used.

1   Locate the CloudFileTransferClient node and click the log text file icon to open the Log file.

    In the Find window, search for an entry similar to this example:

    `2015-08-13 09:42:19,461 +0200Z {} ERROR M3CFTClient com.infor.cft.on premise.filetransfer.util.OnPremUtil: Failed to get keystore content.`

    `ErrorKeystore was tampered with, or password was incorrect java.io.IOEx ception: Keystore was tampered with, or password was incorrect`

2   In the Grid Admin UI, click the link Application CloudFileTransferClient.

3   Click Configuration and then Edit Properties.

4   Click on the link for Keystore Password. Set the password again.

    The password must be the same password which you entered at initial download.

See information about downloading the M3 Cloud File Transfer client package in the *Infor M3 Configuration Guide*.

Infor Enterprise Collaborator troubleshooting

You may encounter several issues while using Infor Enterprise Collaborator tools. These known challenges are solved by performing the necessary troubleshooting steps.

Failure to remove a mapping folder

Managing the Resources could not be deleted error message

This error shows if you removed a mapping folder and this error is displayed: Resources could not be deleted.

Solve this by following these steps:

1   Set the Heap memory. Select **Windows > Preferences > General**.
2   Select **Show Heap Status**.
3   Click **Apply**.
4   Click **OK**.
    The heap memory status indicator and a garbage bin icon are displayed at the bottom of Eclipse.
5   Click the garbage bin icon to start garbage collection.
6   Retry the delete operation.

If this does not work, restart Eclipse and then retry the delete operation.

Managing the Failed to remove error message

This error shows if you did not completely remove deleted mappings that are not displayed in Mapping Explorer View (MEV).

Solve this by ensuring that the mapping is deleted.

1   Go to the standard Navigator view.
2   Right-click the same mapping and select delete.

Eclipse hangs when opening the Mapping editor

If this happens, restart Eclipse and then reopen the Mapping Editor.

Remove a repository function

You can only remove a repository function from a repository by using the **Repository function > Delete from Repository** option from the Mapper context menu.

A function must be in a mapping before it can be removed.

Mapping Console is not connected with the Mapper editor

When you reopen Eclipse, it shows the last active Mapper when it was previously closed. The Mapping Console will not be associated with the editor.

If this happens, select the Mapper to reactivate its connection to the console.

Editing function parameters are not instantly reflected in Java editor

If you are simultaneously editing a Java function in the Java editor and editing the function parameters in the Mapper, the changes are not immediately posted in the Java editor.

To update the Java editor, close and then reopen the Java editor.

Missing input or output document elements

Using XML schemas could be very complex, for example, the use of `OAGIS BOD's`. Mapper may lack the support for complex schema constructions which could result to missing document elements or attributes.

If this happens, use the `xdstat` Mapping Console command to check how Mapper deals about the schema. Support for missing constructs will be added, as needed.

Tracing

The **Mapper Log Console** in Mapper is an additional tool only for developers. Developers use this tool to trace or debug the internal components of a mapping.

Expensive tasks error messages

Expensive task errors happen when you have overlapping or conflicting schedules that result in tasks being in a stuck state. There are two ways to know when there is an expensive task in the background. First, if the email messaging option is enabled, you will get an email that shows the error message "Unable to start MEC Scheduled Backup due to a running expensive task.". Second, if the email messaging option is enabled or disabled, an error message is always displayed in the UI when you try to perform a maintenance task while there is a running task in the task page. The stuck task results in a stuck state and when you run another task while there is an existing running task that is already in a stuck state, the new task would not move forward due to the preceding task that is in a stuck state.

You must ensure that there are no overlapping tasks to solve this issue. There must not be any running tasks by the time the next task starts.

1 Search for overlapping tasks by checking your schedules and scheduled times.
2 Locate the overlapping task.
3 Cancel or reschedule the overlapping task.
4 Run the tasks again.

Deleting a task that is in a stuck state

You can delete tasks that are in a stuck state and identify scheduled tasks that can potentially cause a stuck task problem.

1 Check if there are any conflicting enabled scheduled tasks in Schedules page. These are scheduled tasks that run at the same time and have almost the same or exact same setup.
2 Go to **Server > Tasks** page and look for any task status that is not yet "Finished". If the task is running for a long time, toggle that task for deletion or termination. The delete option is available in the latest IEC version.
3 Proceed with the Schedule Backup Archive or any Schedule/Maintenance tasks.

DataTranslator Set Record Limit error

You will encounter stuck messages when you retrieve large data from the M3 DataTranslator.

There is an IEC application property which would allow a specified number of records to be fetched from M3 DataTranslator to IEC application. This property is set through Grid Application property.

Note: M3 DataTranslator is cached through the IEC database to enable fetching of records consistently between DataTranslation changes.

Property	Default limit value	Data Type	Definition
DataTranslator.SetLst-MaxRec	20000	Integer	Sets the number of DataTranslator records to be retrieved. Increasing the value would increase the loading time. Setting this to 0 would fetch all Data-Translator records.

Note: The default limit value of is set to 20000 records.

If the DataTranslator.SetLstMaxRec is set to 0, all DataTranslator records are fetched. The processing of the messages will wait until all the records from M3 have been fetched.

Note: When the DataTranslation data is larger than the record limit set, stuck messages can occur. So limit the amount of DataTranslation records from BE. You can increase the number of datatranslator records to a number that is greater than the default value of 20000 to address this issue, but this will increase the loading time.

If the DataTranslator.SetLstMaxRec is set to a non-zero value, this value becomes the amount of data to be fetched. If the amount of DataTranslation from M3 is greater than this value, this will be logged as an error in the IEC event log but will not block any messages from running once the limit has finished fetching.

Note: Not all of the data is fetched if the records from the DataTranslation in BE is greater than the limit value. Translation messages will fail since not all of the data is in the cache.

Follow these steps to address stuck messages from large values:

1   When the data being fetched from M3 is insufficient (non-zero value in DataTranslation.SetLstMaxRec) and the translation messages failed during processing, you must increase the DataTranslator.SetLstMaxRec value until the data is sufficiently fetched.

2   When there is an increase in the number of DataTranslations in BE and you like to cache all the DataTranslations, you must increase the DataTranslator.SetLstMaxRec value.

Business Document Mapper error messages

When the source or target for one or more links in a mapping is missing or cannot be accessed, it results to broken links.

Managing the Missing document error

You can still open the mapping but, it will contain an additional error for broken links.

1   Open the `SchemaIn/Out.File` and `SchemaIn/Out.Name` properties of the mapping.

2   Verify that these properties refer to a schema file that is present in the mapping folder.

   If there are differences in the file name, use the Eclipse Refactor option to rename the schema file to the expected name and try the mapping once again.

   See

Broken link error

When the Mapper cannot find the source or target for one or more links in a mapping, it results in broken links.

The details for broken links are displayed on the Broken Links tab of the Mapping Console view. On the **Broken links** tab, check the original internal ID for the target and source elements and a statement about which of them is missing. Double-click the link to select the remaining element in the editor.

Note: The broken links persist in the mapping until removed. Before you can save mappings to a Mapper database, you must resolve all broken links.

To see a more detailed information on broken or missing links, use the `linkstat` command on the **Console Commands and Output** tab of the Mapping Console view.

The `linkstat` command output details the number of broken links using these categories:

- Scalar links broken links between variables/constants, function parameters, and input or output document elements.

- Non-scalar links are broken links between loops, function parameters, and input or output document elements.

- Control links are broken links controlling loops that are missing a condition. This link error rarely occurs.

The link details use this format:

```
<SRC_NAME or "Missing!"><SRC_ID> -- <LINK_TYPE>☐ <SRC_NAME or "Miss
ing!"><TGT_ID>
```

These are the variables used:

- `SRC_NAME` is the name of the source mapping element.

- `TGT_NAME` is the name of the target mapping element.

   Note: For a missing mapping element, the NAME is `"Missing!"`

- `SRC_ID` is the ID of the source mapping element.

- `LINK_TYPE` encodes the link type.

- `TGT_ID` is the ID of the target mapping element.

Here is an example:

```
Element link: foreach_orderregels [ LID2 ] -- LM --> Missing! [ MOD3 ]
```

```
Element link: oId [ OID81 ] -- PM --> Missing! [ MOD4 ]
Element link: oFilePath [ OID90 ] -- PM --> Missing! [ MOD5 ]
Element link: oSpecification [ OID92 ] -- PM --> Missing! [ MOD8 ]
Element link: oId [ OID91 ] -- PM --> Missing! [ MOD9 ]
Element link: oDescription [ OID93 ] -- PM --> Missing! [ MOD10 ]
Element link: oPercentage [ OID94 ] -- PM --> Missing! [ MOD11 ]
Element link: oLanguage_Description [ OID83 ] -- PM --> Missing! [ MOD12
 ]
```

This table shows the link types and descriptions used in our example:

Link type	Description
ML	Input Document Element > Loop
LM	Loop > Output Document Element
FL	Function > Loop
PL	Output Parameter > Loop
MP	Document Element > Input Parameter
PM	Output Parameter > Document Element
VP	Variable > Input Parameter
PV	Output Parameter > Variable
KP	Constant > Input Parameter
PP	Output Parameter > Input Parameter
master_control	Control Link
straight	Data Link
variable	Link to/from variable
constant	Link from constant

Note: When a mapping is saved, all broken links are discarded. To keep the linkstat command information for analysis, copy and store the details.

Input or output missing root element error

When a mapping uses an input or output root document element that can no longer be found in the input or output schema, it results in input or output missing root element error.

This could happen if you edit or change a schema or when you import a mapping with inconsistent metadata. In the case of inconsistent metadata, check the mapping.input/output.element from the mapping.properties file and verify that the values are correct. You can manually edit the values and try to open the mapping again.

Invalid Mapping Schema error

You must follow the various rules related to making custom BODs, if not, you will encounter errors. The BOD Schema error "Invalid Mapping Schema for this Custom BOD mapping." is caused when you create custom BOD mappings that violate these rules. To address this issue, you must correct the part of the custom BOD that violates the rules.

Some of these custom BOD mapping rules are explained in Creating a new custom BOD on page 209. You can also consult the knowledge base article for Infor M3 CE Custom BODs.

To know more about the complete set of Mapping Extensibility rules and how to use them, consult Infor Integration Consultants or go through the educational course for Enterprise Collaboration Administration and Business Document Mapper, as these courses discuss in detail how the rules are applied through Java source code.

Unused Warning

Your maps can develop an Unused WARNING problem during validation, if you perform Change Document without clearing first, the user area of your schema document. If you save the mapping to the server repository, you might encounter "Failed to lookup allocated id for schema."

To solve this issue, you must clean up the broken links and run the console command `mcu` to remove the obsolete mapping objects or replacement elements. Follow these steps to address this issue.

1   Go to **Mapping Console Tab > Console Commands and Output**.

2   Specify `mcu`.

3   Press **enter**.

4   Press **y**.

5   Press **enter**.
     The console will display 'Action is DONE' once the cleanup is complete. You can now save the mappings in the server repository.

Appendix A: Infor Enterprise Collaborator Mapper supporting topics

These topics explain the terminologies and elements of Infor Enterprise Collaborator Mapper.

Infor Enterprise Collaborator Java security mappings

You cannot use arbitrary Java code in Enterprise Collaborator Administration mappings due to security reasons, as you can for the on-prem version of Infor Enterprise Collaborator (IEC). Java code that could compromise the Java processes running in the cloud for IEC, or could access system resources in the cloud, is prohibited.

White list

All user-created Java code in a mapping is validated against a white list. This list contains all Java packages, variables and objects that can be used. Any other Java construction cannot be used. If you try to generate or publish a mapping containing prohibited Java constructions, the generation or publishing will fail. You will get an error message in the mapper console with information about the prohibited Java code, which is managed by the IEC server. You can not bypass the mapper and for example, manually edit the .map file to work around the system. A mapping can only be published if it only contains Java code compliant with the white list. All standard BOD mappings comply with the white list.

IEC APIs

Most of the available IEC Application Programming Interfaces (APIs), known as "IEC Utilities", have been altered in IEC CE compared to the on-prem version of IEC. This is to encapsulate the internal IEC functionality so that you cannot gain access to it from mappings. You need to convert on-prem mappings to make these mappings run in IEC CE. A "before and after" table for IEC APIs is linked in the White list knowledge base articles on page 389. The Java Build Path and the Javadoc for the mapper's Java editor is updated with these new IEC APIs. You can view the latest list of console commands to access the Javadoc in Mapping Console commands on page 189.

Example:

Code type	Line
Old code:	`String s = strUUID;`
New code:	`String s = getMapUUID();`

Standard Java restrictions

There are restrictions on the use of standard Java constructions due to security reasons. The mapper's Java editor does not know these restrictions, so you will not get any contextual help when writing standard Java code. If you use restricted standard Java code in a mapping, you will get an error message in the mapper console when you generate or publish the mapping. The white list is linked in this document.

Example:

Code type	Line
Old code:	`System.out.println("Value=" + s);`
New code:	`logDebug("Value=" + s);`

White list knowledge base articles

The white list tables contain the updated versions of the Java packages, variables and objects that you can use. Check these tables regularly for updates.

Follow these links to view the latest white list tables from Infor knowledge base articles:

Knowledge base articles	Article number
IEC Java whitelist for mappings	2060060
M3 CE Integration and Infor OS content	2017962

XML schema basics for wildcard elements

The `xs:any` element enables the author to extend the XML document with elements not specified by the schema. Parent elements for the wildcard element can be `xs:sequence` or `xs:choice`.

For Infor BODs, the wildcard element is used in the complex type `UserAreaType` which is used for all `UserArea` elements. The complex type `UserAreaType` is defined in this generic schema:

`..\..\Resources\Components\Common\Fields.xsd`

This table shows the description of the `namespace` and `processContents` attributes:

Attribute	Description
namespace	Optionally, use this attribute to specify the namespaces containing the elements to use. This attribute can be set to one of these: • `##any` for elements from any namespace is allowed • `##other` for elements from any namespace that is not the namespace of the parent element can be present • `##local` for elements must come from no namespace • `##targetNamespace` for elements from the namespace of the parent element can be present • List of `{URI references of namespaces, ##targetNamespace, ##local}` for elements from a space-de-limited list of the namespaces that can be present
ProcessContents	Optionally, use this attribute to specify how the XML processor should handle validation against the elements specified by this any element. This attribute can be set to one of these: • `strict` the XML processor must obtain the schema for the required namespaces and validate the elements. This is the default. • `lax` the same as strict but, if the schema cannot be obtained, no errors will occur. • `skip` the XML processor does not attempt to validate any elements from the specified namespaces.

Mapper and mapper runtime will only support namespace=`"##any"` and processContents=`"strict"`. Any element defined in any namespace can be used as replacement element. The replacement element must be declared in the schema for the default namespace or in any other custom namespace with a custom schema that is referenced from the instance document.

To validate the instance document, it must have a reference to its schema. The attribute `xmlns` in this example specifies the default namespace declaration:

`xmlns="http://schema.infor.com/InforOAGIS/2"`

This declaration tells the schema-validator that all non-prefixed elements used in this XML document are declared in the namespace `"http://schema.infor.com/InforOAGIS/2"`.

When you have the XML Schema Instance namespace available
`xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"` you can use the attribute
`xsi:schemaLocation`

```
xsi:schemaLocation="http://schema.infor.com/InforOAGIS/2
http://schema.infor.com/2.6.4/InforOAGIS/BODs/Developer/SyncProduc
tionOrder.xsd"
. This attribute has two values separated by a space. The first value is
 the namespace to use. The second value is the location of the XML schema
 to use for that namespace, for example:
```

The schema-validator can validate all elements that belong to the default namespace, that is, non-prefixed elements, and elements that belong to other namespaces defined in the schema.

To use a replacement element that is not declared in the referenced schema, you must declare the replacement element in a custom namespace. Then, give a reference to the custom schema for that namespace.

You can use several custom namespaces in an Instance document.

1 Specify the custom namespace declaration.

   You connect the custom namespace with a prefix, for example
   `xmlns:cust="http://schema.infor.com/InforOAGIS/Custom"`.

   In this example, the prefix "`cust`" is used for all elements that are declared in the custom namespace
   "`http://schema.infor.com/InforOAGIS/Custom`". This attribute has two values separated
   by a space. The first value is the name.

2 Add the location of the custom schema for the custom namespace in the attribute
   "`xsi:schemaLocation`".

   Follow this code:

```
xsi:schemaLocation="http://schema.infor.com/InforOAGIS/2
http://schema.infor.com/2.6.4/InforOAGIS/BODs/Developer/SyncProduc
tionOrder.xsd
http://schema.infor.com/InforOAGIS/Custom CustomHeader.xsd"
```

The schema-validator can find schemas for both the default namespace and for the custom namespaces provided that the URLs for the schema files are correct.

You can have several custom namespaces in an Instance document. You can also have several custom schemas for one custom namespace, using replacement elements declared in the same custom namespace, but from different custom schemas. The custom namespaces and custom schemas are listed as pairs in the `xsi:schemaLocation` attribute value.

This is an example of a root tag for an Infor BOD where you have one custom namespace and two custom schemas:

```
<SyncProductionOrder xmlns="http://schema.infor.com/InforOAGIS/2"
```

```
xmlns:cust="http://schema.infor.com/InforOAGIS/Custom"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://schema.infor.com/InforOAGIS/2
http://schema.infor.com/2.6.4/InforOAGIS/BODs/Developer/SyncProduc
tionOrder.xsd
http://schema.infor.com/InforOAGIS/Custom
custom/UserAreaExtensions/Ferrari/CustomHeader.xsd
http://schema.infor.com/InforOAGIS/Custom
custom/UserAreaExtensions/Ferrari/CustomDetail.xsd" releaseID="9.2" ver
sionID="2.6.4">
```

This is the corresponding `CustomHeader` schema:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns="http://schema.infor.com/InforOAGIS/Custom"
targetNamespace="http://schema.infor.com/InforOAGIS/Custom" elementFormDe
fault="qualified"
attributeFormDefault="unqualified">
 <xs:element name="AdditionalInformation">
  <xs:complexType>
   <xs:sequence>
    <xs:element name="ReceivingWarehouse" type="xs:string"/>
    <xs:element name="SoldToBPName" minOccurs="0" maxOccurs="unbounded">
     <xs:complexType>
      <xs:simpleContent>
       <xs:extension base="xs:string">
        <xs:attribute name="languageID"/>
       </xs:extension>
      </xs:simpleContent>
     </xs:complexType>
    </xs:element>
    <xs:element name="ShipToBPName" minOccurs="0" maxOccurs="unbounded">
     <xs:complexType>
      <xs:simpleContent>
       <xs:extension base="xs:string">
        <xs:attribute name="languageID"/>
       </xs:extension>
      </xs:simpleContent>
     </xs:complexType>
    </xs:element>
    <xs:element name="InterfaceUpdate" type="xs:string"/>
   </xs:sequence>
  </xs:complexType>
 </xs:element>
</xs:schema>
```

Note: The custom namespace must be declared as target namespace in all custom schemas.

In this BOD instance example, the replacement element "`Property`" is declared in the default namespace, while the replacement element "`AdditionalInformation`" is declared in the custom namespace "`http://schema.infor.com/InforOAGIS/Custom`" and uses the prefix "`cust`".

The replacement element "`Property`" is defined in the standard BOD schema, and the replacement element "`cust:AdditionalInformation`" is defined in the `CustomHeader` schema.

```
<UserArea>
 <Property>
  <NameValue name="ln.Priority" type="NumericType">999</NameValue>
 </Property>
 <Property>
  <NameValue name="ln.Owner" type="StringType">SFC</NameValue>
 </Property>
 <Property>
  <NameValue name="ln.Routing" type="StringType">001</NameValue>
 </Property>
 <cust:AdditionalInformation>
  <cust:ReceivingWarehouse>WHAMS1</cust:ReceivingWarehouse>
  <cust:SoldToBPName/>
  <cust:SoldToBPName languageID=""/>
  <cust:ShipToBPName/>
  <cust:ShipToBPName languageID=""/>
  <cust:InterfaceUpdate>No</cust:InterfaceUpdate>
 </cust:AdditionalInformation>
</UserArea>
```

Instead of defining the structure of an XML document only in one schema for the default namespace, the instance document can also use additional custom namespaces and provide references to custom schemas.

General and Wildcard element terminologies

For brevity, use these guidelines to refer to products mentioned in this user guide:

- Business Document Mapper tool - BDM or Mapper.
- Infor Enterprise Collaborator - IEC or EC.
  This product contains the IEC Server and the Business Document Mapper client tool.
- Ion Grid - Grid
- Lifecycle Manager - LCM
- Business Engine - BE
- UVJ function - User defined Void Java function
- UBJ function - User defined Boolean Java function
- REP function - Repository function
- Mapper Database - The database repository for mappings.
  For ION this is one of the repositories. For IEC it is the EC database.

- Mapper Server - The server where you generate, publish, and run mappings. For ION, this is part of ION Runtime Services. For IEC, it is part of the EC server.

- Wildcard element refers to the wildcard schema component, that is the XML schema element "any".

- Replacement element is an element that will be used in instance documents as wildcard replacement contents, that is where the schema specifies a wildcard element.

- Custom namespace is a namespace that is not defined in the main schema. The default namespace is defined in the main schema.

- Custom prefix is a prefix for a custom namespace.

- Custom schema is a schema where a custom namespace is defined. It is referenced from the instance document, not from the schema for the default namespace.

Note: Wildcard element terms are only used in the Mapper. These are not W3C terms.

API Metadata Handling prerequisites

Set up a server connectivity to fetch API Metadata from the server.

See

API Metadata is needed if the MI feature is used. The Mapper internally fetches the API Metadata. You can also explicitly fetch/update these metadata by clicking the **Reload Cache** either in the Server Preference Page or in the New MI API Call wizard page.

M3 API function

Use an M3 API function, or M3A function for short, to access M3 functionality. These are the types of M3 API functions available:

- **Single**, when a single data record is returned from M3.

- **Repeating**, when multiple data records are returned from M3.

  The M3A type is defined by the property **Type**.

  A repeating M3A must control a loop because multiple data records are returned from M3. The main objective is to create repeating elements in the output document based on the result returned from the M3 API call.

  You can add M3 API in a loop using the wizard. Follow these guidelines:

  - If you drop M3 API in the first element of a loop, you can only select Repeating M3 API functions.

  - If you drop M3 API in other locations in a mapping, you can only select Single M3 API functions.

An M3 API function call can fail and the action taken is controlled by the **ErrorHandling** attribute which can have these values:

- Use **Exit map on M3 NOK** to stop the current mapping operation.

- Use **Exit loop on M3 NOK** to stop the current loop operation in progress.

- Use **Ignore M3 NOK** to ignore the error.

If the M3 API may not be installed at the customer, for example a country specific API, and the M3-API is never called if it is not installed, use the **API Installed** attribute which can have these values:

- Use **Required** to require the M3 API to be installed when generating or publishing the mapping. This is the default setting.

- Use **Optional** if the mapping can be executed without calling this M3 API.

    The M3A function must be placed within an **If** widget or within a Loop in the mapping so that it is not used if the customer has not installed the M3 API. For example, if the M3 API is a Russia-specific API, the M3 API must only be called when the current M3 Division is set up for Russia.

    If the M3 API is not installed or not updated, when the mapping is generated or published, only a warning is displayed.

    If the M3 API that is not installed is called when executing the mapping, an error is displayed: "The API <MI program name>.<transaction name> is not installed!". Then, the mapping will close, regardless of how the **ErrorHandling** attribute is set.

M3 Company and Division

M3 API calls use the M3 Settings for the Company (CONO) and Division (DIVI) configurations.

Sequence overview

Follow these guidelines:

- Place an M3 Settings function before the M3 API call.

- Set the desired Company and Division by assigning variables to the Company and Division parameters.

- Instead of using variables, you can also, for example, use input elements that contain the Company and Division.

All succeeding API calls will use the Company and Division as set by the M3 Settings function.

You can use another M3 Settings function later in the sequence to change Company or Division again, if needed.

Use inside a loop

If you use an M3 Settings function inside a loop that is controlled by a repetitive M3 API, the data from the M3 API controlling the loop is fetched using the Company and Division set when entering the loop. It will not use the Company and Division set by the M3 Settings function inside the loop. This is because the actual API call for the loop is only executed once, when entering the loop.

If you execute an M3 Settings function inside a loop in the first iteration, but not in the succeeding iterations, the Company and Division that are set by the M3 Settings function in the loop will be used. This Company and Division set will be used for all succeeding M3 API calls **in** the loop, that is, even for iteration two, three, and more. It will also be used for all succeeding M3 API calls **after** the loop.

Use of default values

If you want to use default Company and Division values as defined by IEC control properties, API reference parameters, or the setting in MNS150, do so by specifying `null` values for both Company and Division, for example, use the widget and do not map anything to the input parameters.

This action may be necessary for these events:

- If you dynamically change the Division by using the M3 Settings function to fetch data from several M3 Divisions, and then you want to go back to the original settings as defined by IEC control properties, API reference parameters, or the setting in MNS150 for the last API call.

- If you want to use default Company or Division values as defined by IEC control properties, API reference parameters, or the setting in MNS150 for all M3 API calls in the mapping. In this event, you do not need to use the M3 Settings function.

Variable values

Use the **String** data type for Company and Division variable values. String values must be enclosed in "" (double quotation marks), for example, `"123"`. Even if the Company is an integer value, use double quotation marks. If the **Division** is specified, **Company** must also be specified, or else a runtime error will occur. In cases where only the Company is specified, the default "blank" Division will be used.

- **Company** must be an integer value between 0 and 999.

  The parameter value has a maximum length 3 of characters and is trimmed before the integer check.

  So, "1", " 1 ", " 1", " 01", and "001", are all valid companies and will be used when establishing a new API connection to M3.

- **Division** must be an alphanumerical value with a maximum length 3 of characters.

  The parameter value is not trimmed.

  The default value is an empty string ("").

  So, "A", "A ", "A ", " A " and " A", are different divisions in M3 and will be used as is, including leading blanks, when establishing a new API connection to M3.

IMS parameters and manifest items

Use IMS parameters to instruct ION how to handle IEC messages and specify preferred values. You can include the following optional IMS parameters as part of the manifest:

Manifest Item	Actual Manifest Constant (in Mapper)	IMS Parameters	Notes
map:ionMessageId	MAP_ION_MESSAGE_ID	messageId	
map:ionToLogicalId	MAP_ION_TO_LOGICAL_ID	toLogicalId	
map:ionAccountingEntity or agr:ionAccountingEntity	MAP_ION_ACCOUNTING_ENTITY AGR_ION_ACCOUNTING_ENTITY	accountingEntity	
map:ionLocation or agr:ionLocation	MAP_ION_LOCATION AGR_ION_LOCATION	location	
map:ionDocumentId or agr:ionDocumentId	MAP_ION_DOCUMENT_ID AGR_ION_DOCUMENT_ID	documentId	
map:ionVariationId	MAP_ION_VARIATION_ID	variationId	Must be a number.
map:ionRevisionId or agr:ionRevisionId	MAP_ION_REVISION AGR_ION_REVISION	revisionId	
map:ionBatchId or agr:ionBatchId	MAP_ION_BATCH_ID AGR_ION_BATCH_ID	batchId	
map:ionBatchSequence or agr:ionBatchSequence	MAP_ION_BATCH_SEQUENCE AGR_ION_BATCH_SEQUENCE	batchSequence	
map:ionBatchSize or agr:ionBatchSize	MAP_ION_BATCH_SEQUENCE AGR_ION_BATCH_SEQUENCE	batchSize	

All of them can be set using the Mapper. Refer to the pattern and sample below:

Sample: setManifestInfo(ManifestConstants.MAP_ION_MESSAGE_ID, "SampleID");

Pattern: setManifestInfo(ManifestConstants.<ActualManifestConstant>, <value>);

Appendix B: Mashup Server (Runtime of Mashups)

The Mashup Server is a grid application which serves as the runtime for web mashups. This is installed together with M3 H5 Enterprise. The Mashup Server also comes with the Mashup Administration Client which is used for maintaining the web mashups.

Use the instructions in this section to deploy web mashups and systems using the Mashup Administration Client. To access this tool, select **Administration Tools > Mashup Administration** from the Infor M3 menu in Infor Ming.le.

The IFS Security role *MASHUP-Administrator* is required to access the **Mashup Administration** page.

Deploying a mashup application

1   From the **Mashup Administration Client** homepage, click **Mashups > Deploy**.
2   Click **Browse** or **Choose File** and locate the .webmashup file that you want to deploy.
3   Select **Public** or **Private**.
   - Public - This option allows you to share the application that you will be deploying.
   - Private - This option allows you to deploy an application for personal use.

   Based on the IFS role that you set, the availability of the mashup can be controlled for different users. Leaving the role blank will make the mashup available for all users who have access to the grid application.
4   Check Enabled.
5   Click **Deploy**.

This table shows the general description of the available options in using Mashup Administration Client after deploying an application:

Option	Description
View All	View the list of deployed applications by you and other users, including applications that you have deployed privately
View Public	View the list of deployed applications made available for public use
View Private	View the list of deployed applications which you have deployed privately

Option	Description
Raw XML Public	View the shared application in raw XML format
Raw XML Private	View the private application in raw XML format

Undeploying a mashup application

1   From the client homepage, click **Mashups > View All**.

2   Click **Undeploy this mashup** on the application that you want to undeploy. Click **Yes**.

Maintaining a mashup application

To maintain an application, click **Mashups > View All** and:

• Click **View Info** to access more details about the ID, Description, and Resources.

• Use the toggle switch to enable or disable a mashup application.

• Specify a new name to change a role in the **Role** text field, and then click **Set**.

   Note: This must be a valid IFS role. Otherwise, user will get an error message when trying to set the role. If nothing is set, the default role will be Mashup-user.

Deploying a system

A system deployed here is a profile XML file which describes the controls, parameters, events, and location of the system.

Note: Only Infor Administrators are allowed to change the Mashup systems.

1   From the client homepage, click **Systems > Deploy system**.

2   Click **Browse** or **Choose File** and select the system profile XML file that you want to deploy.

3   Click **Deploy**.

This table shows the general description of available options in using Mashup Administration Client after deployment:

Option	Description
View All	View the list of systems deployed

Option	Description
Raw XML	View the application in raw XML format

Maintain the system after deploying it.

See

Undeploying a system

Note: Only Infor Administrators are allowed to change the Mashup systems.

1   From the client homepage, click **Systems > View All**.

2   Click **Undeploy this system** on the system that you want to undeploy.

3   Click **Yes**.

Maintaining a system

Note: Only Infor Administrators are allowed to change the Mashup systems.

To maintain a system, click **Systems > View All** and:

- Click **View Info** to access more details about Controls and Description, including the parameters and events of the feature.

- Use the toggle switch to enable or disable a system.

- Edit the link in the text field to change the BaseUrl, and then click **Set**.

Appendix C: Outbound BODs

This list details the outbound BODs that M3 CE Core publishes. The user actions that results in events which generate an outbound BOD from M3 CE Core are determined by the event rules set that is delivered with the M3 BODs. The list is too extensive to detail in this document, see *Infor M3 Business Engine BODs Rules Documentation.*

Verb	Noun/variant
Show	AccountingBookDefinition
Sync	AccountingBookDefinition
Show	AccountingChart
Sync	AccountingChart
Show	AccountingEntity
Sync	AccountingEntity
Show	AccountingJournal
Sync	AccountingJournal
Acknowledge	AdvanceShipNotice
Process	AdvanceShipNotice
Show	AdvanceShipNotice
Sync	AdvanceShipNotice
Acknowledge	AssetMaster
Show	AssetMaster
Sync	AssetMaster
Acknowledge	BillOfMaterials
Process	BillOfMaterials
Show	BillOfMaterials
Sync	BillOfMaterials
Show	BillOfResources
Sync	BillOfResources

Verb	Noun/variant
Acknowledge	BillToPartyMaster
Show	BillToPartyMaster
Sync	BillToPartyMaster
Show	CarrierParty
Sync	CarrierParty
Acknowledge	CarrierRoute
Sync	Catalog
Show	ChartOfAccounts
Sync	ChartOfAccounts
Show	CodeDefinition for Accounting Dimensions
Sync	CodeDefinition for Accounting Dimensions
Show	CodeDefinition for AP Payment Methods
Sync	CodeDefinition for AP Payment Methods
Show	CodeDefinition for AR Payment Methods
Sync	CodeDefinition for AR Payment Methods
Acknowledge	CodeDefinition for Business Sectors
Show	CodeDefinition for Capabilities
Sync	CodeDefinition for Capabilities
Show	CodeDefinition for Cost Centers
Sync	CodeDefinition for Cost Centers
Show	CodeDefinition for Customer Discount Groups
Sync	CodeDefinition for Customer Discount Groups
Show	CodeDefinition for Customer Groups
Sync	CodeDefinition for Customer Groups
Show	CodeDefinition for Customs Statistics Number
Sync	CodeDefinition for Customs Statistics Number
Acknowledge	CodeDefinition for Customs Statistics Number
Show	CodeDefinition for Delivery Terms
Sync	CodeDefinition for Delivery Terms
Show	CodeDefinition for Dynamic Dimension
Sync	CodeDefinition for Dynamic Dimension

Verb	Noun/variant
Show	CodeDefinition for Equipment Classes
Sync	CodeDefinition for Equipment Classes
Show	CodeDefinition for Equipment Groups
Sync	CodeDefinition for Equipment Groups
Show	CodeDefinition for Equipment Meter
Sync	CodeDefinition for Equipment Meter
Show	CodeDefinition for Error Code 1s
Sync	CodeDefinition for Error Code 1s
Show	CodeDefinition for Error Code 2s
Sync	CodeDefinition for Error Code 2s
Show	CodeDefinition for Error Code 3s
Sync	CodeDefinition for Error Code 3s
Acknowledge	CodeDefinition for Feature Options
Show	CodeDefinition for Feature Options
Sync	CodeDefinition for Feature Options
Show	CodeDefinition for Freight Terms
Sync	CodeDefinition for Freight Terms
Show	CodeDefinition for General Code
Sync	CodeDefinition for General Code
Show	CodeDefinition for Harbor & Airports
Sync	CodeDefinition for Harbor & Airports
Show	CodeDefinition for Individual Item Groups
Sync	CodeDefinition for Individual Item Groups
Show	CodeDefinition for Inspection Result Values
Sync	CodeDefinition for Inspection Result Values
Show	CodeDefinition for Item Discount Groups
Sync	CodeDefinition for Item Discount Groups
Show	CodeDefinition for Item Types
Show	CodeDefinition for Item Free Field 1
Sync	CodeDefinition for Item Free Field 1
Acknowledge	CodeDefinition for Item Free Field 1

Verb	Noun/variant
Show	CodeDefinition for Item Free Field 3
Sync	CodeDefinition for Item Free Field 3
Acknowledge	CodeDefinition for Item Free Field 3
Show	CodeDefinition for Item Free Field 4
Sync	CodeDefinition for Item Free Field 4
Acknowledge	CodeDefinition for Item Free Field 4
Show	CodeDefinition for Item Free Field 5
Sync	CodeDefinition for Item Free Field 5
Acknowledge	CodeDefinition for Item Free Field 5
Acknowledge	CodeDefinition for Item Groups
Sync	CodeDefinition for Item Types
Show	CodeDefinition for Misc Cost Types
Sync	CodeDefinition for Misc Cost Types
Show	CodeDefinition for Options
Sync	CodeDefinition for Options
Show	CodeDefinition for Payment Terms
Sync	CodeDefinition for Payment Terms
Show	CodeDefinition for Points of Shipment
Sync	CodeDefinition for Points of Shipment
Sync	CodeDefinition for Price List Tables
Show	CodeDefinition for Price List Tables
Acknowledge	CodeDefinition for Product Groups
Show	CodeDefinition for Product Lines
Sync	CodeDefinition for Product Lines
Show	CodeDefinition for Product Service Extend
Sync	CodeDefinition for Product Service Extend
Show	CodeDefinition for Product Structure Classes
Sync	CodeDefinition for Product Structure Classes
Show	CodeDefinition for Product Structure Types
Sync	CodeDefinition for Product Structure Types
Show	CodeDefinition for Purchase Order Types

Verb	Noun/variant
Sync	CodeDefinition for Purchase Order Types
Show	CodeDefinition for Routes
Sync	CodeDefinition for Routes
Show	CodeDefinition for Sales Order Charges
Sync	CodeDefinition for Sales Order Charges
Show	CodeDefinition for Sales Order Types
Sync	CodeDefinition for Sales Order Types
Acknowledge	CodeDefinition for Seasons
Show	CodeDefinition for Seasons
Sync	CodeDefinition for Seasons
Show	CodeDefinition for Service Action
Sync	CodeDefinition for Service Action
Show	CodeDefinition for Service Action Text
Sync	CodeDefinition for Service Action Text
Show	CodeDefinition for Service Assignment Types
Sync	CodeDefinition for Service Assignment Types
Show	CodeDefinition for Service Centers
Sync	CodeDefinition for Service Centers
Show	CodeDefinition for Service Error Cause
Sync	CodeDefinition for Service Error Cause
Show	CodeDefinition for Service Error Symptom
Sync	CodeDefinition for Service Error Symptom
Show	CodeDefinition for Service Managers
Sync	CodeDefinition for Service Managers
Show	CodeDefinition for Service Order Types
Sync	CodeDefinition for Service Order Types
Show	CodeDefinition for Service Qualifications
Sync	CodeDefinition for Service Qualifications
Show	CodeDefinition for Service Types
Sync	CodeDefinition for Service Types
Acknowledge	CodeDefinition for Stock Locations

Verb	Noun/variant
Show	CodeDefinition for Stock Locations
Sync	CodeDefintion for Stock Locations
Show	CodeDefinition for Supplier Invoice Authorized Users
Sync	CodeDefinition for Supplier Invoice Authorized Users
Show	CodeDefinition for Transportation Methods
Sync	CodeDefinition for Transportation Methods
Show	CodeDefinition for Unit Codes
Sync	CodeDefinition for Unit Codes
Acknowledge	CodeDefinition for Unit Codes
Show	CodeDefinition for VAT Codes
Sync	CodeDefinition for VAT Codes
Show	CodeDefinition for Work Order Types
Sync	CodeDefinition for Work Order Types
Show	ConfiguredAssetMaster for Maintenance Order
Sync	ConfiguredAssetMaster for Maintenance Order
Show	ConfiguredAssetMaster for Service Order
Sync	ConfiguredAssetMaster for Service Order
Acknowledge	ContactMaster
Show	ContactMaster
Sync	ContactMaster
Show	Contract for CO Blanket Agreement
Sync	Contract for CO Blanket Agreement
Acknowledge	Contract for PO Agreement
Show	Contract for PO Agreement
Sync	Contract for PO Agreement
Sync	CreditTransfer
Acknowledge	CurrencyExchangeRateMaster
Show	CurrencyExchangeRateMaster
Sync	CurrencyExchangeRateMaster
Show	CurrencyRateType
Sync	CurrencyRateType

Verb	Noun/variant
Acknowledge	CustomerPartyMaster
Show	CustomerPartyMaster
Sync	CustomerPartyMaster
Show	CustomerReturn
Sync	CustomerReturn
Sync	DebitTransfer
Acknowledge	EmployeeWorkTime
Show	FinancialCalendar
Sync	FinancialCalendar
Acknowledge	FinancialPartyMaster
Show	FinancialPartyMaster
Sync	FinancialPartyMaster
Show	InspectDelivery
Sync	InspectDelivery
Acknowledge	InspectionOrder for Work Order
Show	InspectionOrder for Work Order
Sync	InspectionOrder for Work Order
Acknowledge	InventoryAdjustment
Show	InventoryAdjustment
Sync	InventoryAdjustment
Acknowledge	InventoryCount
Show	InventoryCount
Sync	InventoryCount
Process	InventoryHold
Show	InventoryHold
Sync	InventoryHold
Show	Invoice for BillforPrepayment
Sync	Invoice for BillforPrepayment
Sync	Invoice for CustomerOrder Invoicing
Sync	Invoice for Fixed Asset Invoicing
Sync	Invoice for Leasing Agreement Invoicing

Verb	Noun/variant
Sync	Invoice for Maintenance Order Invoicing
Sync	Invoice for Manual Invoicing
Sync	Invoice for Project Invoicing
Sync	Invoice for Rental Agreement invoicing
Acknowledge	ItemMaster
Show	ItemMaster
Sync	ItemMaster
Sync	LCLTaxReportSupplierTax
Sync	LCLTaxReportVATPeriodic
Sync	LCLTradeStatisticsEUSalesPurchase
Sync	LCLTradeStatisticsIntrastatExtrastat
Acknowledge	Location for Facility
Show	Location for Facility
Sync	Location for Facility
Acknowledge	Location for Warehouse
Show	Location for Warehouse
Sync	Location for Warehouse
Sync	M3EDICarrierRoute
Sync	M3EDICarrierRouteDelivery
Sync	M3EDIInvoiceCommon
Sync	M3EDIPriceList
Sync	M3EDIPurchaseOrder
Sync	M3EDIRemittanceAdvice
Sync	M3EDISalesOrder
Sync	M3EDISalesorderAcknowledge
Sync	M3EDIShipment
Sync	M3EDIShipment Delivery
Sync	M3EDIPickList
Acknowledge	M3ReferenceTransaction
Sync	M3ReferenceTransaction
Show	PayableTransaction

Verb	Noun/variant
Sync	PayableTransaction
Acknowledge	PayFromPartyMaster
Show	PayFromPartyMaster
Sync	PayFromPartyMaster
Acknowledge	Person
Show	Person
Sync	Person
Acknowledge	ProductionOrder
Show	ProductionOrder
Sync	ProductionOrder
Show	ProjectMaster
Sync	ProjectMaster
Show	Promotion
Sync	Promotion
Process	PulseAlert
Process	PulseNotification
Process	PulseTask
Acknowledge	PurchaseOrder
Show	PurchaseOrder
Sync	PurchaseOrder
Acknowledge	QualitySpecification
Show	QualitySpecification for Work Order
Sync	QualitySpecification for Work Order
Acknowledge	QualityTestMaster
Show	QualityTestMaster for Work Order
Sync	QualityTestMaster for Work Order
Acknowledge	Quote
Show	Quote
Sync	Quote
Show	ReceivableTransaction
Sync	ReceivableTransaction

Verb	Noun/variant
Acknowledge	ReceiveDelivery
Show	ReceiveDelivery
Sync	ReceiveDelivery
Show	ReceiveDelivery for Warehouse
Sync	ReceiveDelivery for Warehouse
Acknowledge	RemitToPartyMaster
Show	RemitToPartyMaster
Sync	RemitToPartyMaster
Sync	RemittanceAdvice
Acknowledge	Requisition
Show	Requisition
Sync	Requisition
Acknowledge	SalesOrder
Show	SalesOrder
Sync	SalesOrder
Sync	SecurityPermissionMaster
Sync	SecurityRoleMaster
Acknowledge	ServiceOrder
Show	ServiceOrder
Sync	ServiceOrder
Acknowledge	ServiceOrder for Work Order
Show	ServiceOrder for Work Order
Sync	ServiceOrder for Work Order
Show	ShipFromPartyMaster
Sync	ShipFromPartyMaster
Acknowledge	Shipment
Process	Shipment
Show	Shipment
Sync	Shipment
Show	Shipment for Delivery
Sync	Shipment for Delivery

Verb	Noun/variant
Acknowledge	ShipToPartyMaster
Show	ShipToPartyMaster
Sync	ShipToPartyMaster
Show	SourceSystemGLMovement for Actual
Sync	SourceSystemGLMovement for Actual
Show	SourceSystemGLMovement for Budget
Sync	SourceSystemGLMovement for Budget
Show	SourceSystemJournalEntry
Sync	SourceSystemJournalEntry
Acknowledge	SupplierInvoice
Sync	SupplierInvoice
Acknowledge	SupplierPartyMaster
Show	SupplierPartyMaster
Sync	SupplierPartyMaster
Acknowledge	Transfer
Sync	Transfer
Show	Transfer
Process	Workflow

Appendix D: Inbound BODs

This table shows the incoming BODs that M3 can accept.

See *Infor M3 inbound BOD mapping and descriptions* for detailed information on the area in which the processed inbound data is displayed.

Verb	Noun/variant
Acknowledge	AdvanceShipNotice
Load	AdvanceShipNotice
Process	AdvanceShipNotice
Sync	AdvanceShipNotice
Process	AssetMaster
Load	BankStatement
Acknowledge	BillOfMaterials
Process	BillOfMaterials
Sync	BillOfMaterials
Process	BillToPartyMaster
Process	CarrierRoute
Process	CodeDefinition for Business Sectors
Sync	CodeDefinition for Capabilities
Process	CodeDefinition for Customs Statistics Number
Process	CodeDefinition for Feature Options
Process	CodeDefinition for Item Free Field 1
Process	CodeDefinition for Item Free Field 3
Process	CodeDefinition for Item Free Field 4
Process	CodeDefinition for Item Free Field 5
Process	CodeDefinition for Item Groups
Process	CodeDefinition for Product Groups

Verb	Noun/variant
Sync	CodeDefinition for Roles
Process	CodeDefinition for Seasons
Process	CodeDefinition for Stock Locations
Process	CodeDefinition for UnitCodes
Process	CodeDefinition for Unit Groups
Process	ContactMaster
Process	Contract for PO Agreement
Process	CurrencyExchangeRateMaster
Process	CustomerPartyMaster
Process	EmployeeWorkTime
Sync	ExpenseReport
Process	FinancialPartyMaster
Process	InspectionOrder for Work Order
Process	InventoryAdjustment
Sync	InventoryAdjustment
Process	InventoryCount
Sync	InventoryCount
Sync	InventoryHold
Process	ItemMaster
Process	Location for Facility
Process	Location for Warehouse
Load	M3EDIAdvanceShipNotice
Load	M3EDICustomerReturn
Load	M3EDIInventoryAdjustment
Load	M3EDIPurchaseOrder
Load	M3EDIReceiveDelivery
Load	M3EDIRemittanceAdvice
Load	M3EDISalesOrder
Load	M3EDIShipment
Load	M3EDIShipmentSchedule
Load	M3EDISupplierInvoice

Verb	Noun/variant
Process	M3ReferenceTransaction
Process	PayFromPartyMaster
Process	Person
Sync	Personnel
Process	ProductionOrder
Acknowledge	PulseAlert
Acknowledge	PulseNotification
Acknowledge	PulseTask
Process	PurchaseOrder
Process	QualitySpecification
Process	QualityTestMaster
Process	Quote
Process	ReceiveDelivery
Sync	ReceiveDelivery
Process	RemitToPartyMaster
Process	Requisition
Process	SalesOrder
Sync	SecurityUserMaster
Sync	ServiceConsumption
Process	ServiceOrder
Process	ServiceOrder for Work Order
Acknowledge	Shipment
Process	Shipment
Sync	Shipment
Process	ShipToPartyMaster
Sync	SourceSystemGLMovement for Budget
Load	SupplierInvoice
Process	SupplierInvoice
Process	Transfer
Acknowledge	Workflow

Appendix E: Master tables for initial load

This table lists the M3 CE master tables for sending the initial load.

Verb	Noun	M3 master table for Initial Load
Sync/Show	AccountingBookDefinition	CSYTAB
Sync/Show	AccountingChart	CSYTAB
Sync/Show	AccountingEntity	CMNDIV
Sync/Show	AccountingJournal	CSYTAB
Sync/Show	AdvanceShipNotice	MHEXRH
Sync/Show	AssetMaster	FFASMA
Sync/Show	BillOfMaterials	MPDHED
Sync/Show	BillOfResources	MPDSRC
Sync/Show	BillToPartyMaster	OCUSAD, OCUSMA
Sync/Show	CarrierParty	CIDMAS
Sync	Catalog	CSYTAB
Sync/Show	ChartOfAccounts	FCHACC
Sync/Show	CodeDefinition for Accounting Dimensions	FCHACC
Sync/Show	CodeDefinition for AP Payment Methods	CSYTAB
Sync/Show	CodeDefinition for AR Payment Methods	CSYTAB
Sync/Show	CodeDefinition for Capabilities	CCADEF
Sync/Show	CodeDefinition for Cost Centers	FCHACC
Sync/Show	CodeDefinition for Customer Discount Groups	CSYTAB
Sync/Show	CodeDefinition for Customer Groups	CSYTAB
Sync/Show	CodeDefinition for Customs Statistics Number	CSYCSN
Sync/Show	CodeDefinition for Delivery Terms	CSYTAB
Sync/Show	CodeDefinition for Dynamic Dimension	FGLEDG

Verb	Noun	M3 master table for Initial Load
Sync/Show	CodeDefinition for Equipment Classes	MIEQCL
Sync/Show	CodeDefinition for Equipment Groups	MIEQGR
Sync/Show	CodeDefinition for Equipment Meter	SINDMD
Sync/Show	CodeDefinition for Error Code 1s	CSYTAB
Sync/Show	CodeDefinition for Error Code 2s	CSYTAB
Sync/Show	CodeDefinition for Error Code 3s	CSYTAB
Sync/Show	CodeDefinition for Feature Options	MPDFHE
Sync/Show	CodeDefinition for Freight Terms	CSYTAB
Sync/Show	CodeDefinition for General Code	CSYTAB
Sync/Show	CodeDefinition for Harbor & Airport	CSYTAB
Sync/Show	CodeDefinition for Individual Item Groups	CSYTAB
Sync/Show	CodeDefinition for Inspection Result Values	MPDIRV
Sync/Show	CodeDefinition for Item Discount Groups	CSYTAB
Sync/Show	CodeDefinition for Item Free Field 1	CSYTAB
Sync/Show	CodeDefinition for Item Free Field 3	CSYTAB
Sync/Show	CodeDefinition for Item Free Field 4	CSYTAB
Sync/Show	CodeDefinition for Item Free Field 5	CSYTAB
Sync/Show	CodeDefinition for Item Types	MITTTY
Sync/Show	CodeDefinition for Misc Cost Types	ACUEXP
Sync/Show	CodeDefinition for Options	MPDOPT
Sync/Show	CodeDefinition for Payment Terms	CSYTAB
Sync/Show	CodeDefinition for Points of Shipment	CSYTAB
Sync/Show	CodeDefinition for Price List Tables	OPLTMA
Sync/Show	CodeDefinition for Product Lines	CRPDLN
Sync/Show	CodeDefinition for Product Service Extend	SITCOM
Sync/Show	CodeDefinition for Product Structure Classes	CSYTAB
Sync/Show	CodeDefinition for Product Structure Types	CSYTAB
Sync/Show	CodeDefinition for Purchase Order Types	MPORDT
Sync/Show	CodeDefinition for Routes	DROUTE
Sync/Show	CodeDefinition for Sales Order Charges	ODCHRG

Verb	Noun	M3 master table for Initial Load
Sync/Show	CodeDefinition for Sales Order Types	OOTYPE
Sync/Show	CodeDefinition for Seasons	CSEAMA
Sync/Show	CodeDefinition for Service Action	SMECOD
Sync/Show	CodeDefinition for Service Action Text	SMETXT
Sync/Show	CodeDefinition for Service Assignment Types	SASSTP
Sync/Show	CodeDefinition for Service Centers	MPDWCT
Sync/Show	CodeDefinition for Service Error Cause	SERCOD
Sync/Show	CodeDefinition for Service Error Symptom	SERSYM
Sync/Show	CodeDefinition for Service Managers	SSRESP
Sync/Show	CodeDefinition for Service Order Types	SSTYPE
Sync/Show	CodeDefinition for Service Qualifications	CSYTAB
Sync/Show	CodeDefinition for Service Types	CSYTAB
Sync/Show	CodeDefinition for Stock Locations	MITPCE
Sync/Show	CodeDefinition for Supplier Invoice Authorized Users	FAPRCD
Sync/Show	CodeDefinition for Transportation Methods	CSYTAB
Sync/Show	CodeDefinition for Unit Codes	CSYTAB
Sync/Show	CodeDefinition for VAT Codes	CSYTAB
Sync/Show	CodeDefinition for Work Order Types	MMORDT
Sync/Show	ConfiguredAssetMaster for Maintenance Order	MILOIN
Sync/Show	ConfiguredAssetMaster for Service Order	SINDIV
Sync/Show	ContactMaster	CCUCON
Sync/Show	Contract for CO Blanket Agreement	OAGRHE
Sync/Show	Contract for PO Agreement	MPAGRH
Sync/Show	CurrencyExchangeRateMaster	CCURRA
Sync/Show	CurrencyRateType	CSYTAB
Sync/Show	CustomerPartyMaster	OCUSMA
Sync/Show	CustomerReturn	OCHEAD
Sync/Show	FinancialCalendar	CSYPER
Sync/Show	FinancialPartyMaster	CBANMA

Verb	Noun	M3 master table for Initial Load
Sync/Show	InspectDelivery	MPLIND
Sync/Show	InspectionOrder for Work Order	MMOOPS
Sync/Show	InventoryAdjustment	MITTRA
Sync/Show	InventoryCount	MITTKV
Sync/Show	InventoryHold	MITTRA
Sync/Show	Invoice for BillforPrepayment	OOPINH
Sync/Show	Invoice for CustomerOrder Invoicing	CINHED
Sync/Show	Invoice for Leasing Agreement Invoicing	CINHED
Sync/Show	Invoice for Maintenance Order Invoicing	CINHED
Sync/Show	Invoice for Manual Invoicing	CINHED
Sync/Show	Invoice for Project Invoicing	CINHED
Sync/Show	Invoice for Rental Agreement Invoicing	CINHED
Sync/Show	ItemMaster	MITMAS
Sync/Show	Location for Facility	CFACIL
Sync/Show	Location for Warehouse	MITWHL
Sync/Show	PayableTransaction	FPLEDG
Sync/Show	PayFromPartyMaster	OCUSMA
Sync/Show	Person	CSYUSR
Sync/Show	ProductionOrder	MWOHED
Sync/Show	ProjectMaster	BPROJS
Sync/Show	Promotion	OPROMH
Sync/Show	PurchaseOrder	MPHEAD
Sync/Show	QualitySpecification for Work Order	MPDIID
Sync/Show	QualityTestMaster for Work Order	MPDIRE
Sync/Show	Quote	OOQUOH
Sync/Show	ReceivableTransaction	FSLEDG
Sync/Show	ReceiveDelivery	MPLIND
Sync/Show	ReceiveDelivery for Warehouse	MHDISH
Sync/Show	RemitToPartyMaster	CIDMAS
Sync/Show	Requisition	MPOPLP

Verb	Noun	M3 master table for Initial Load
Sync/Show	SalesOrder	OOHEAD
Sync	SecurityPermissionMaster	CMNPSU
Sync	SecurityRoleMaster	CMNROL
Sync/Show	ServiceOrder	SSHEAD
Sync/Show	ServiceOrder for Work Order	MMOHED
Sync/Show	ShipFromPartyMaster	CIDMAS
Sync/Show	Shipment	DCONSI
Sync/Show	Shipment for Delivery	MHDISH
Sync/Show	ShipToPartyMaster	OCUSAD, OCUSMA
Sync/Show	SourceSystemGLMovement for Actual	FBAVAL
Sync/Show	SourceSystemGLMovement for Budget	FBUDET
Sync/Show	SourceSystemJournalEntry	FGLHED
Sync	SupplierInvoice	FPLEDG
Sync/Show	SupplierPartyMaster	CIDMAS
Sync/Show	Transfer	MGHEAD

Appendix F: BODs used in integrations with Infor M3 CE

This section contains the list of BODs that are available with M3 CE.

Outbound BODs from M3 CE to integrated applications

This table shows the BODs that are available with M3 CE. Where **To applications** is empty, the BOD is generated by M3 CE, but is not currently used by interfacing products. The BOD is available to be processed through ION and any application that is set up to receive it.

Verb	Noun/variant	To applications
Sync	AccountingEntity	Infor Ming.le
		Infor M3 Supplier Rebate
		Infor Governance, Risk and Compliance - Authorizations Insight
		Infor Localization Services Platform
Sync	AccountingChart	Infor Localization Services Platform
Acknowledge	AdvanceShipNotice	Infor Nexus
Process	AdvanceShipNotice	Infor WMS
Process	BillOfMaterials	Infor PLM for Process
Sync	BillOfMaterials	Infor Rhythm for Commerce
Acknowledge	BillToPartyMaster	Infor CRM
Sync	BillToPartyMaster	Infor CPQ
		Infor CRM
		Infor WMS
		Infor Rhythm for Commerce

Verb	Noun/variant	To applications
Sync	CarrierParty	Infor CRM
		Infor WMS
		Infor Rhythm for Commerce
Acknowledge	CarrierRoute	Infor Nexus
Sync	Catalog	Infor Rhythm for Commerce
Sync	ChartOfAccounts	Infor Governance, Risk and Compliance - Process Insight
		Infor Localization Services Platform
Sync	CodeDefinition for AP Payment Methods	Infor M3 Supplier Rebate
Sync	CodeDefinition for AR Payment Methods	Infor CPQ
		Infor CRM
		Infor Rhythm for Commerce
Acknowledge	CodeDefinition for Business Sectors	Infor PLM for Fashion
Sync	CodeDefinition for Cost Centers	Infor EAM
		Infor Expense Management
Acknowledge	CodeDefinition for Customs Statistics Number	Infor PLM for Fashion
Sync	CodeDefinition for Delivery Terms	Infor CPQ
		Infor Rhythm for Commerce
Acknowledge	CodeDefinition for Feature Options	Infor PLM for Fashion
Sync	CodeDefinition for Feature Options	Infor Rhythm for Commerce
Sync	CodeDefinition for FreightTerms	Infor EAM
Sync	CodeDefinition for General Code	Infor PLM for Fashion
		Infor Rhythm for Commerce
		Infor Supplier Exchange
Sync	CodeDefinition for Harbor & Airports	Infor Rhythm for Commerce
Sync	CodeDefinition for Item Free Field 1	Infor PLM for Fashion
Sync	CodeDefinition for Item Free Field 3	Infor PLM for Fashion
Sync	CodeDefinition for Item Free Field 4	Infor PLM for Fashion
Sync	CodeDefinition for Item Free Field 5	Infor PLM for Fashion

Verb	Noun/variant	To applications
Acknowledge	CodeDefinition for Item Groups	Infor PLM for Fashion
Sync	CodeDefinition for Item Types	Infor Rhythm for Commerce Infor PLM for Fashion
Sync	CodeDefinition for Payment Terms	Infor CPQ Infor CRM Infor Supplier Exchange Infor M3 Supplier Rebate Infor Rhythm for Commerce
Acknowledge	CodeDefinition for Product Groups	Infor PLM for Fashion
Sync	CodeDefinition for Product Structure Classes	Infor PLM for Process
Sync	CodeDefinition for Routes	Infor Rhythm for Commerce
Sync	CodeDefinition for Stock Locations	Infor EAM
Sync	CodeDefinition for Transportation Methods	Infor CPQ Infor CRM Infor Rhythm for Commerce
Sync	CodeDefinition for Unit Codes	Infor EAM Infor Rhythm for Commerce Infor Supplier Exchange Infor M3 Supplier Rebate
Acknowledge	CodeDefinition for Unit Codes	Infor PLM for Fashion
Acknowledge	ContactMaster	Infor CRM
Sync	ContactMaster	Infor CRM
Sync	CreditTransfer	Infor Localization Services GEMS
Sync	CurrencyExchangeRateMaster	Infor CPQ Infor EAM Infor Expense Management Infor M3 Supplier Rebate

Verb	Noun/variant	To applications
Sync	CustomerPartyMaster	Infor CPQ
		Infor CRM
		Infor PLM for Process
		Infor WMS
		Infor Rhythm for Commerce
		Infor Governance, Risk and Compliance - Process Insight
		Infor Localization Services Platform
Sync	CustomerReturn	Infor Governance, Risk and Compliance - Process Insight
Sync	DebitTransfer	Infor Localization Services GEMS
Sync	FinancialCalendar	Infor Localization Services Platform
		Infor M3 Supplier Rebate
Acknowledge	FinancialPartyMaster	Infor Supplier Exchange
Sync	FinancialPartyMaster	Infor Supplier Exchange
Sync	InspectDelivery	Infor EAM
Sync	Invoice for Customer Order Invoicing	Infor CRM
		Infor Governance, Risk and Compliance - Process Insight
		Infor Localization Services Platform
		Infor Rhythm for Commerce
Sync	Invoice for Fixed Asset Invoicing	Infor Governance, Risk and Compliance - Process Insight
	Invoice for Leasing Agreement Invoicing	Infor Localization Services Platform
	Invoice for Maintenance Order Invoicing	
	Invoice for Manual Invoicing	
	Invoice for Project Invoicing	
	Invoice for Rental Agreement Invoicing	
Acknowledge	InventoryAdjustment	Infor EAM
Sync	InventoryCount	Infor EAM
Acknowledge	ItemMaster	Infor PLM for Fashion
		Infor PLM for Process
		Infor WMS

Verb	Noun/variant	To applications
Sync	ItemMaster	Infor CPQ
		Infor CRM
		Infor EAM
		Infor Nexus
		Infor PLM for Fashion
		Infor PLM for Process
		Infor WMS
		Infor Supplier Exchange
		Infor M3 Supplier Rebate
		Infor Rhythm for Commerce
		Infor Governance, Risk and Compliance - Process Insight
		Infor Localization Services Platform
Sync	LCLTaxReportSupplierTax	Infor Localization Services GEMS
Sync	LCLTaxReportVATPeriodic	Infor Localization Services Platform
Sync	LCLTradeStatisticsEUSalesPurchase	Infor Localization Services GEMS
Sync	LCLTradeStatisticsIntrastatExtrastat	Infor Localization Services GEMS
Sync	Location for Warehouse	Infor CRM
		Infor M3 Supplier Rebate
		Infor Rhythm for Commerce
Sync	PayFromPartyMaster	Infor CRM
Sync	PayableTransaction	Infor M3 Supplier Rebate
		Infor Governance, Risk and Compliance - Process Insight
		Infor Localization Services Platform
Sync	Person	Infor CRM
		Infor Ming.le
		Infor Governance, Risk and Compliance - Authorizations Insight
Sync	ProjectMaster	Infor Expense Management
Sync	Promotion	Infor Rhythm for Commerce
Acknowledge	PurchaseOrder	Infor Nexus
		Infor Supplier Exchange

Verb	Noun/variant	To applications
Sync	PurchaseOrder	Infor EAM
		Infor WMS
		Infor Nexus
		Infor Supplier Exchange
		Infor M3 Supplier Rebate
		Infor Governance, Risk and Compliance - Process Insight
Acknowledge	QualitySpecification	Infor PLM for Process
Acknowledge	Quote	Infor CRM
Sync	Quote	Infor CRM
		Infor Governance, Risk and Compliance - Process Insight
Sync	ReceivableTransaction	Infor CRM
		Infor Rhythm for Commerce
		Infor Governance, Risk and Compliance - Process Insight
		Infor Localization Services Platform
Acknowledge	ReceiveDelivery	Infor EAM
Sync	ReceiveDelivery	Infor EAM
		Infor Nexus
		Infor Supplier Exchange
		Infor M3 Supplier Rebate
		Infor Governance, Risk and Compliance - Process Insight
Acknowledge	RemitToPartyMaster	Infor Supplier Exchange
Sync	RemitToPartyMaster	Infor Supplier Exchange
Sync	RemittanceAdvice	Infor Expense Management
Acknowledge	Requisition	Infor EAM
Sync	Requisition	Infor Governance, Risk and Compliance - Process Insight
Acknowledge	SalesOrder	Infor CPQ
		Infor CRM
		Infor Rhythm for Commerce

Verb	Noun/variant	To applications
Sync	SalesOrder	Infor CRM
		Infor Rhythm for Commerce
		Infor Governance, Risk and Compliance - Process Insight
Sync	SecurityPermissionMaster	Infor Governance, Risk and Compliance - Authorizations Insight
Sync	SecurityRoleMaster	Infor Ming.le
		Infor Governance, Risk and Compliance - Authorizations Insight
Sync	ShipFromPartyMaster	Infor WMS
Process	Shipment **Note:** One BOD can contain multiple Deliveries, triggered for changes in (DRS100) for Transportation Management.	Infor WMS
Sync	Shipment **Note:** One BOD can contain multiple Deliveries, triggered for changes in (DRS100) for Transportation Management.	Infor Localization Services Platform
Sync	Shipment for Delivery **Note:** One BOD can contain one Delivery only.	Infor CRM Infor Rhythm for Commerce Infor Localization Services Platform
Acknowledge	ShipToPartyMaster	Infor CRM
Sync	ShipToPartyMaster	Infor CPQ Infor CRM Infor WMS Infor Rhythm for Commerce
Sync	SourceSystemGLMovement for Actual	Infor Localization Services Platform
Sync	SourceSystemJournalEntry	Infor Governance, Risk and Compliance - Process Insight Infor Localization Services Platform
Acknowledge	SupplierInvoice	Infor Nexus Infor M3 Supplier Rebate

Verb	Noun/variant	To applications
Sync	SupplierInvoice	Infor Document Capture - Supplier Invoices
		Infor Governance, Risk and Compliance - Process Insight
		Infor Localization Services Platform
Acknowledge	SupplierPartyMaster	Infor Supplier Exchange
Sync	SupplierPartyMaster	Infor EAM
		Infor PLM for Fashion
		Infor PLM for Process
		Infor WMS
		Infor M3 Supplier Rebate
		Infor Supplier Exchange
		Infor Governance, Risk and Compliance - Process Insight
		Infor Localization Services Platform

Inbound BODs to M3 CE from integrated applications

This table shows the BODs that can be received and processed by M3 CE. Where the application is blank, the BOD is not currently used by interfacing application.

Verb	Noun/variant	From applications
Load	AdvanceShipNotice	Infor Supplier Exchange
Process	AdvanceShipNotice	Infor Nexus
Load	BankStatement	Infor Localization Services GEMS
		Infor Nexus
Acknowledge	BillOfMaterials	Infor PLM for Process
Process	BillToPartyMaster	Infor CRM
Process	CarrierRoute	Infor Nexus
Process	CodeDefinition for Business Sectors	
Process	CodeDefinition for Customs Statistics Number	Infor PLM for Fashion
Process	CodeDefinition for Feature Options	Infor PLM for Fashion

Verb	Noun/variant	From applications
Process	CodeDefinition for Item Free Field 1	Infor PLM for Fashion
Process	CodeDefinition for Item Free Field 3	Infor PLM for Fashion
Process	CodeDefinition for Item Free Field 4	Infor PLM for Fashion
Process	CodeDefinition for Item Free Field 5	Infor PLM for Fashion
Process	CodeDefinition for Item Groups	Infor PLM for Fashion
Process	CodeDefinition for Product Groups	Infor PLM for Fashion
Process	CodeDefinition for Stock Locations	Infor EAM
Process	CodeDefinition for Unit Codes	Infor PLM for Fashion
Process	ContactMaster	Infor CRM
Process	CustomerPartyMaster	Infor CRM
Sync	ExpenseReport	Infor Expense Management
Process	FinancialPartyMaster	Infor Supplier Exchange
Process	InventoryAdjustment	Infor EAM
Sync	InventoryAdjustment	Infor WMS
Sync	InventoryCount	Infor EAM
Sync	InventoryHold	Infor WMS
Process	ItemMaster	Infor PLM for Fashion Infor PLM for Process Infor WMS
Process	PayFromPartyMaster	Infor CRM
Process	PurchaseOrder	Infor Nexus Infor Supplier Exchange
Process	QualitySpecification	Infor PLM for Process
Process	Quote	Infor CRM
Process	ReceiveDelivery	Infor EAM
Sync	ReceiveDelivery	Infor WMS
Process	RemitToPartyMaster	Infor Supplier Exchange

Verb	Noun/variant	From applications
Process	Requisition	Infor EAM
Process	SalesOrder	Infor CPQ
		Infor CRM
		Infor Rhythm for Commerce
Sync	SecurityUserMaster	Infor Ming.le
		Infor Governance, Risk and Compliance - Authorizations Insight
Sync	ServiceConsumption	Infor EAM
Process	Shipment	Infor EAM
Sync	Shipment	Infor WMS
Process	ShipToPartyMaster	Infor CRM
Sync	SourceSystemGLMovement **Note:** Used for Budget.	Infor d/EPM
Load	SupplierInvoice	Infor Document Capture - Supplier Invoices
Process	SupplierInvoice	Infor Nexus
		Infor M3 Supplier Rebate
Process	SupplierPartyMaster	Infor Supplier Exchange