Infor Fashion PLM Connector Toolbox User Guide

Release 15.1.9.3

Important Notices

The material contained in this publication (including any supplementary information) constitutes and contains confidential and proprietary information of Infor.

By gaining access to the attached, you acknowledge and agree that the material (including any modification, translation or adaptation of the material) and all copyright, trade secrets and all other right, title and interest therein, are the sole property of Infor and that you shall not gain right, title or interest in the material (including any modification, translation or adaptation of the material) by virtue of your review thereof other than the non-exclusive right to use the material solely in connection with and the furtherance of your license and use of software made available to your company from Infor pursuant to a separate agreement, the terms of which separate agreement shall govern your use of this material and all supplemental related materials ("Purpose").

In addition, by accessing the enclosed material, you acknowledge and agree that you are required to maintain such material in strict confidence and that your use of such material is limited to the Purpose described above. Although Infor has taken due care to ensure that the material included in this publication is accurate and complete, Infor cannot warrant that the information contained in this publication is complete, does not contain typographical or other errors, or will meet your specific requirements. As such, Infor does not assume and hereby disclaims all liability, consequential or otherwise, for any loss or damage to any person or entity which is caused by or relates to errors or omissions in this publication (including any supplementary information), whether such errors or omissions result from negligence, accident or any other cause.

Without limitation, U.S. export control laws and other applicable export and import laws govern your use of this material and you will neither export or re-export, directly or indirectly, this material nor any related materials or supplemental information in violation of such laws, or use such materials for any purpose prohibited by such laws.

Trademark Acknowledgements

Publication Information

Release: Infor Fashion PLM 15.1.9.3
Publication Date: December 3, 2019
Document code: ifplm_15.1.9.3_ifplmctug__en-us

Contents

About this guide

This guide shows you how to use the administrative and user functions of Infor Fashion PLM (IFPLM) Connector Toolbox and how to complete specific setup and maintenance tasks.

Intended audience

This guide is intended for system administrators who are responsible for setting up, configuring, and implementing the integration between the Infor Fashion PLM application and external systems.

Prerequisite knowledge

To fully understand the information presented in this guide, you must have knowledge and experience in system configuration and administration of these applications:

- Infor Fashion PLM
- Infor ION
- M3 Business Engine

Organization

This table describes the sections of this guide:

Section	Description
Overview	This section describes what you need to know about the application and its common features. It also describes the architecture and setup of the integration.
Configuring the application	This section explains the configuration steps that must be done after installation to start using the application.
Integration Designer overview	This section describes what you need to know about the integration tasks, integration task templates, and integration flows. It also describes available integration templates in the application.
Designing integration flows	This section explains how to use Manage Templates and Flow Designer to manage integration tasks and integration flows.
Setting up work units	This section describes how to execute integration flows between Infor Fashion PLM and external systems by setting up workers and defining conditions and schedule settings.

Section	Description
Migrating data	This section describes what you need to know about the data migration feature of Connector Toolbox. It also explains the necessary procedures to migrate data from an SQL or OLE DB-compliant database to the Infor Fashion PLM database.
Field mappings for M3 Business Engine	This section lists supported fields exported to M3 BE and imported fields from M3 BE to Infor Fashion PLM.

Related documents

You can find these documents in the Infor Xtreme Support portal:

- Infor Fashion PLM
  - *Infor Fashion PLM Connector Toolbox Installation Guide*
  - *Infor Fashion PLM Planning Application Host Installation Guide*
  - *Planning Application Host Administration Guide* (Also available as *PAH Online Help*)
  - *Infor Fashion PLM User Guide* (Also available as *Infor Fashion PLM Online Help*)
  - *Inbound Infor Fashion PLM BOD Mapping and Descriptions*
  - *Outbound Infor Fashion PLM BOD Mapping and Descriptions*
- M3 Business Engine
  - *M3 ION Integration Guide for Infor Fashion PLM*
  - *Cross BOD Mapping and Descriptions - M3 Business Engine BODs with Infor Fashion PLM*
- Infor ION
  - *Infor ION Installation Guide*
  - *Infor ION Connect Administration Guide*

Contacting Infor

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

If we update this document after the product release, we will post the new version on the Infor Support Portal. To access documentation, 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: Overview

Learn the concepts and key features of the application.

Infor Fashion PLM Connector Toolbox overview

Infor Fashion PLM Connector Toolbox is an application that enables integration between Infor Fashion PLM and external systems, such as M3 Business Engine (M3 BE) and other ERP systems. It can execute point-to-point transactions and, alternatively, connect Infor Fashion PLM to external systems through Infor Intelligent Open Network (ION). Infor ION is an integration platform that enables different business systems to share information by translating it into standardized XML.

Connector Toolbox serves as the interface between Infor Fashion PLM and other systems, and is also the tool that is used to set up, configure, and manage the integrations. Through Connector Toolbox, you can process information that are created in Infor Fashion PLM, such as styles, discrete items, and SKUs, to M3 BE and get information from M3 BE back to Infor Fashion PLM.

You can also use Connector Toolbox to migrate data directly from a system that uses SQL or OLE DB-compliant database to the Infor Fashion PLM database.

Integration overview

Infor Fashion PLM Connector Toolbox integrates data created in the Infor Fashion PLM application with external systems. In an integrated setup through Connector Toolbox, Infor Fashion PLM serves as the planning tool and the external system serves as the operating system.

Infor Fashion PLM Connector Toolbox can access data from three sources:

- Infor Fashion PLM application database
- Infor Fashion PLM Integration XML file
- Other Microsoft SQL Server databases

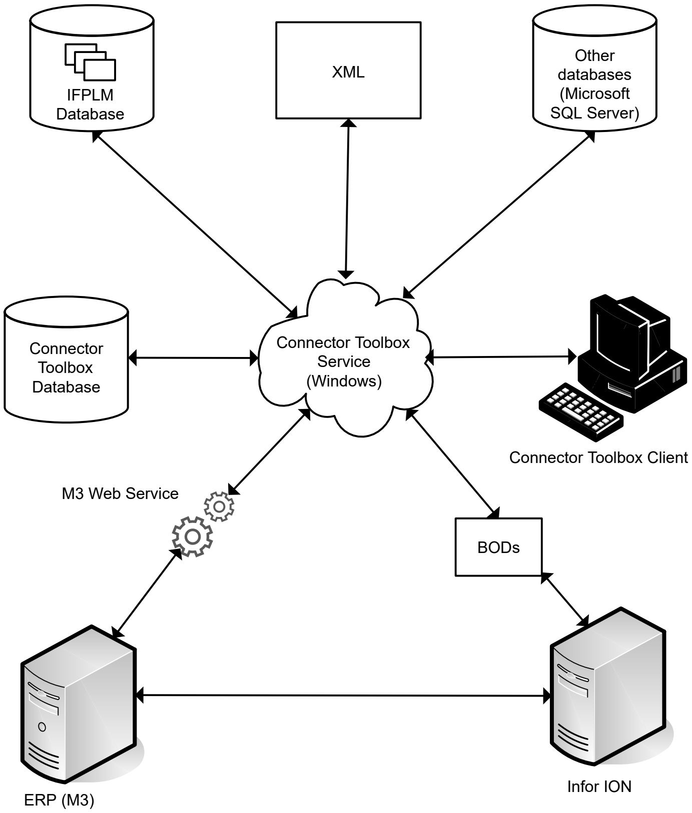
Data from these sources is processed in the Connector Toolbox Service. From the service, data can then be organized in various integration flows through the Connector Toolbox Client. After processing,data is then placed in the Connector Toolbox database.

The Connector Toolbox Service can establish the integration these ways:

- M3 web services can be used to import and export data between Infor Fashion PLM and Infor M3Business Engine (M3 BE).
  Note: This point-to-point integration applies to integration with M3 BE only.
- Asynchronous interface, which is executed by using business object documents (BODs), can transmit extensive amounts of data between Infor Fashion PLM and Infor external systems through Infor ION.
- Database to database integration.
- XML integration.

In a point-to-point integration setup, data gathered from the application database is processed in Connector Toolbox and then transferred directly to the ERP (M3) through M3 Web Service. When using BODs for integration, data is sent to the ION Inbox/Outbox. Infor ION then obtains the data from the ION Inbox and sends the data to the ERP.

This diagram shows the integration architecture between Infor Fashion PLM and external systems through Connector Toolbox:


IFPLM Database
XML
Other databases (Microsoft SQL Server)
Connector Toolbox Database
Connector Toolbox Service (Windows)
Connector Toolbox Client
M3 Web Service
BODs
ERP (M3)
Infor ION

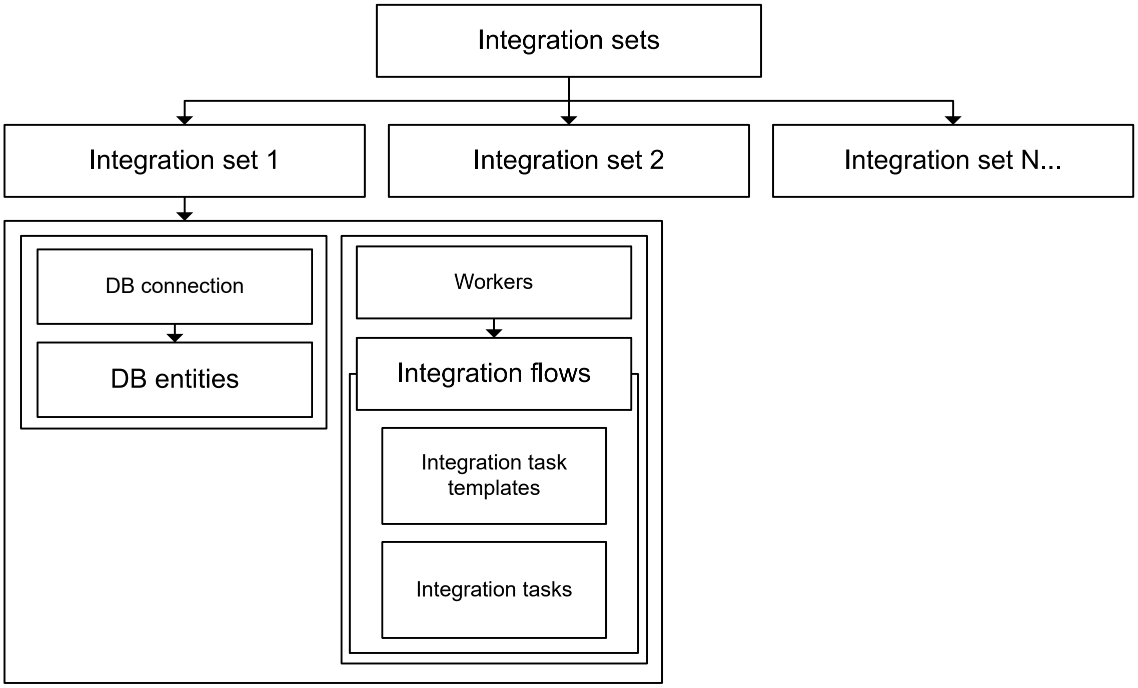
Integration components

In the Connector Toolbox, you can create and save multiple integration sets. This lets you manage integration of Infor Fashion PLM with multiple platforms. Each integration set has its own set of components. You can manage integration sets by using the Manage Integration Sets module of Connector Toolbox.

This table shows the different components of an integration set:

Component	Description
DB connections	To start using Connector Toolbox, you must configure the application to establish a connection to the Infor Fashion PLM database and any other external ERP system that uses Microsoft SQL Server database. DB connections are defined per integration set. **Note:** Default database connections for PLM, ION, and Connector Toolbox are defined during installation.
DB entities	After connecting Connector Toolbox to a database, you must identify the supported tables in the database that you will use in the application.
Integration tasks	Integration tasks are the building blocks of an integration flow. Integration tasks contain settings, parameters, and values that are performed when launching an integration flow.
Integration task templates	Integration tasks are managed as integration task templates. Integration task templates are used when creating tasks in an integration flow. You can create new and update existing integration task templates that can be reused when adding or updating tasks in an integration flow. You can create, update, or delete integration task templates by using the Manage Templates module of Connector Toolbox.
Integration flows	Integration flows contain all executable integration tasks for a corresponding work unit. You can create, update, or delete integration flows by using the Flow Designer module of Connector Toolbox.
Workers	Workers are background monitoring processes that perform batch processing of integration flows and its corresponding integration tasks. These are polling jobs that execute the integration flows that you created in the Flow Designer based on conditions that are set in the Worker Setup. **Note:** Workers are processed sequentially. The worker will run only after all records under the same worker are processed.

This diagram provides an overview of how integration sets are set up in Connector Toolbox:



Available tasks

These are the available functions in the application:

- Export Style (New)
- Export Style (Update)
- Export SKU
- Import Item Type
- Import Suppliers

Common application features

Infor Fashion PLM Connector Toolbox contains features for monitoring tasks and handling message. This application also displays user interface functions for managing records and groups.

Monitoring tasks

Integration in Infor Fashion PLM Connector Toolbox requires you to execute tasks to process planning data. These tasks may be referred to as user-initiated tasks, which are executed remotely on the computer or remote server where Connector Toolbox is installed.

Connector Toolbox manages user-initiated tasks as background tasks. These must be executed in the background without affecting other tasks that you are executing. Background tasks can be scheduled to run immediately or at a later date and time. Scheduled on a remote server, a background task in Connector Toolbox can run continuously until the task is complete.

You can observe the status of a background task by launching the task monitor. A task monitor is a utility built into Infor Fashion PLM Connector Toolbox that uses Windows services to monitor user-initiated background tasks. The task details are displayed in the task monitor to enable to user to track the updates in integration of tasks. To launch the task monitor, rest on Task Monitor on the right sidebar. If you want to resend failed integration flows, click the Reset Retry Count button.

Handling messages

Messages enable you to view the status of background tasks that you execute in Infor Fashion PLM Connector Toolbox. To view messages, click Messages on the sidebar of the application. The Messages pane displays the name, module, and date of a message. It also displays the processing details of a message.

Right-click features for column headers

Right-click option	Description
Sort Ascending	Arranges your column records in ascending order
Sort Descending	Arranges your column records in descending order
Clear Sorting	Removes any grouping order to your column records
Group By This Column	Arranges your records by column group
Show Group Panel	Shows your records by column groups
Show Column Chooser	Enables you to arrange and remove columns
Best Fit	Centers columns on the screen for maximum visibility
Best Fit (all columns)	Centers all columns on the screen for maximum visibility
Show Filter Editor	Filters column records by specifying specific columns to display
Show Search Panel	Displays the search panel

Chapter 2: Configuring the application

Learn the necessary configuration procedures to start using the application.

Specifying M3 Business Engine account

Optionally, you can set up the integration between Infor Fashion PLM and M3 BE. Configuring the application enables the support for M3 Rest API connection. Perform this configuration if the integration is supporting M3 Rest API. You must specify an M3 BE account in the Connector Toolbox application.

1. In the navigation pane, select **Setup > M3 Rest API Settings**.
2. Based on the selected M3 BE Rest API setup, specify this information:
   - Basic Authentication

     M3 Username
     Specify the M3 username.

     M3 Password
     Specify the password for the M3 user name.

   - OAuth Authentication

     Consumer key
     Specify the certificate consumer key.

     Secret key
     Specify the secret key.
     Note: Contact your administrator for the consumer key and secret key details.

3. Optionally, add a client certificate configuration:
   a. Select **Use SSL**.
   b. Specify this information:

      Client Certificate
      Specify the location of the client certificate.

      Certificate Password
      Specify the certificate password.

4 Click **Save**.

Defining the service settings

Optionally, you can define the service settings to automatically set the status of a selected integration set to Set to Production.

1 In the Navigation pane, select **Setup > Service Settings**.
2 Select the available option.
3 Click **Save**.

Managing integration sets

In the Connector Toolbox, you can manage multiple integration sets, which gives you the flexibility to manage integration of Infor Fashion PLM with multiple platforms. By using the Manage Integration Sets module of Connector Toolbox, you can create, copy, or delete an integration set, set an integration set to production, and start or stop workers for an integration set.

1 In the navigation pane, select **Integration Sets > Manage Integration Sets**.

  A default integration set is available on the Files section. This represents the current integration flow that is available in the Connector Toolbox. By default, this integration set is set to production and is set to running state.

  The Preview section displays a list of integration flows and workers that are available for the selected integration set.

2 Optionally, click an integration flow on the Integration Flows section to view all associated integration flow tasks.
3 To create a new integration set, click **New Integration Set**.

  To create a copy of an existing integration set, right-click an integration set and select **Copy**.

4 Specify an integration set name.

  Note: The name must only contain alphanumeric characters.

5 Click **Create**.

Set the integration set to production

1 Select the integration set.
2 To generate the necessary entities that are required for the integration flows and workers, select **Generate Entities on Set to Production**.

If Generate Entities on Set to Production check box is not selected, the integration set will use previously generated entities, if there are any, and the integration set will perform faster since entities will not be generated.

Note: If you modify or make changes in the Database Entities module or if the current state of the database entities such as tables and columns has changed, you must select Generate Entities on Set to Production before setting the integration set to production. You can leave the Generate Entities on Set to Production check box blank on the succeeding set to production actions, unless there are changes on the database entities.

See Identifying supported database tables on page 17.

3 Select the integration set and click **Set to Production**.

Alternatively, right-click the integration set and select **Set to Production**.

Stop or start integration set

1 Right-click the integration set.

2 Select **Stop** or **Start**.

Note: When an integration set is started, the State column displays Running. When the integration set is stopped, the State column displays Stopped.

View details of integration set

1 In the navigation pane, select an integration set from the **Integration > View Integration Set** list.

2 Optionally, view and edit the components of the currently selected integration set by using the Integration Designer functions.

Selecting an integration set for viewing does not start or set the integration set to production.

Establishing database connections

To start using Connector Toolbox, you must configure the application to establish a connection to the Infor Fashion PLM database or to any other Microsoft SQL Server database.

During installation, connection strings for PLM, ION, and Connector Toolbox were modified to establish connections to the databases of these applications.

1 In the navigation pane, select **Integration Designer > DB Connections**.

2 To modify an existing database connection, in the DB Connections panel, click the connection string and update the values of the parameters.

3 To establish a new database connection, in the DB Connections panel, select **Click here to add a new row** and specify this information:

Name
Specify a name for the database connection string.

Connection String
Specify the connection string to the database.

4 Click **Save**.
5 To verify that your connection strings are correct, click **Test Connection**.

Identifying supported database tables

After connecting Connector Toolbox to a database, you must identify the supported tables in the database that you will use in the application.

1 In the navigation pane, select **Integration Designer > Database Entities**.
2 In the Entity Generation section, specify this information:

Entity Alias

Select an entity alias from the list. Optionally, specify a new entity alias and click **Add**. This alias serves as the unique identifier of the database within the application.

Database

Select a database from the list. Available options depend on the databases that you connected to the application.

See Establishing database connections on page 16.

Schema
By the default, FSH1 is selected.

To switch to another version or schema, select the schema that you intend to use. Available options depend on the database that you selected.

Note: When integrating data from the Infor Fashion PLM database, ensure that you select the schema that is associated with the Infor Fashion PLM version that you intend to use. See “Managing versions” in *Infor Fashion PLM Online Help*.

3 Select the corresponding Include? check box for the database entity that you intend to use in the application.
4 Click **Save**.

Chapter 3: Integration Designer overview

Learn the integration tasks, integration task templates, and integration flows of the application.

Integration tasks and templates

Integration Designer is the key feature of the application, enabling you to design and control the flow of the integration between Infor Fashion PLM and external systems through Infor ION. To use Integration Designer, you must first understand the different components of an integration set.

See Integration components on page 11.

Integration task types

These types of integration tasks serve as the building blocks of an integration flow:

Task type	Description
Connector	Contains generic parameters that can be reused at any point in the integration flow
Query	Performs a query-like operation but in a form of a task
Integration	Defines the source or end destination of data that is processed for integration **Note:** The source and destination of an integration task can be an Entity, M3 web service, or BOD.
QueryTransform	A combination of Query task and Integration task Performs a Query task with an entity resource type and saves the information to a BODArea integration type
Cleanup	Performs a database table clean-up (delete statement) operation

Default integration task templates

You can reuse, update, and delete existing integration task templates and also create new templates that can be used when creating integration flows. By default, these integration task templates are available in the application:

Task type	Template	Description
Connector	CONNECTORTOOLBOX_GENERIC_PARAMETERS	Contains the Connector Toolbox database connection string, and is used as a parameter by other tasks through parameter binding.
Connector	FASHIONPLM_GENERIC_PARAMETERS	Contains generic Infor Fashion PLM parameters that are needed and consumed by other tasks through parameter binding.
Connector	GENERIC_TASK	Used to signify the end point of an integration flow.
Connector	ION_CONNECTOR_PARAMETERS	Contains ION-specific parameters that are needed and consumed by other tasks through parameter binding.
Connector	M3GENERICPARAMETERS	Contains necessary M3-specific parameters that are needed and consumed by other tasks through parameter binding.
Connector	MISCELLANEOUS	Contains all miscellaneous or generic parameters that are needed and consumed by other tasks through parameter binding.
Query	ION_INBOX_ENTRY	A query task template the queries an inbound BOD in the ION database.
Query	ION_INBOX_HEADER_BODTYPE	A query task template contain the metadata to get the BodType header value of COR_INBOX_HEADERS ION database table.
Query	ION_INBOX_HEADER_FROMLOGICALID	A query task template that contains metadata to get the BOD header FromLogicalId value from COR_ INBOX_HEADERS ION database table.

Task type	Template	Description
Query	ION_INBOX_HEADER_MESSAGEID	A query task template that contains the metadata to get the BOD header MessageId value from COR_INBOX_HEADERS ION database table.
Query	ION_INBOX_HEADER_TENANTID	A query task template that contains metadata to get the BOD header TenantId value from COR_INBOX_HEADERS ION database table.
Query	ION_INBOX_HEADER_TOLOGICALID	A query task template that contains metadata to get the BOD header ToLogicalId value from COR_INBOX_HEADERS ION database table.
Query	POLL_ACKNOWLEDGE_CODEDEFINITION	A polling query task template that periodically checks an ION Inbox for any received AcknowledgeCodeDefinition BOD.
Query	POLL_ACKNOWLEDGE_ITEMMASTER	A polling query task template that periodically checks an ION Inbox for any received AcknowledgeItemMaster BOD.
Query	QUERY_INBOX_VIEW	A query task template that obtains values from InboxView in the ION database.
Query	QUERY_INBOX_VIEW_BY_REFERENCE	A query task template that obtains values filtered by ReferenceID column from InboxView in the ION database.
Query	QUERY_M3_ITEM_TYPE	A query task template that contains metadata to obtain values for importing item types from M3 BE through MI Program/Transaction MMS200MI - SellItemType.
Query	QUERY_M3_SUPPLIER	A query task template that contains metadata to obtain values for importing suppliers from M3 BE through MI Program/Transaction CRS620 - LstSuppliers.

Task type	Template	Description
Query	QUERY_SEND_STYLE_ADD_FEATURE_WORKUNIT_X	A query task template that contains metadata to get the equivalent M3 'X' features of a style in the Infor Fashion PLM database.
Query	QUERY_SEND_STYLE_ADD_FEATURE_WORKUNIT_Y	A query task template that contains metadata to get the equivalent M3 'Y' features of a style in the Infor Fashion PLM database.
Query	QUERY_SEND_STYLE_ADD_FEATURE_WORKUNIT_Y_FLOW2	A query task template that contains the metadata to get the Color records associated to a Style filtered by Style Id. Gets value from V_INTGSTYLECOLOR database view.
Query	QUERY_SEND_STYLE_ADD_FEATURE_WORKUNIT_Z	A query task template that contains metadata to get the equivalent M3 'Z' features of a style in the Infor Fashion PLM database.
Query	QUERY_SEND_STYLE_SKU_WORKUNIT	A query task template that queries the SKU entity from the Infor Fashion PLM database
Query	QUERY_SEND_STYLE_WORKUNIT	A query task template that contains metadata to query the style entity from the Infor Fashion PLM database.
Query	QUERY_STYLE_BY_INTCODE	A query task template that contains the metadata to get SKU records (Features & Options) filtered by Interface Item number. Gets value from V_INTGSTYLESKU database view.
Query	QUERY_STYLE_BY_INTCODE	A query task template that contains the metadata to get Style records filtered by Interface Item number. Gets value from V_QRYSTYLE database view.

Task type	Template	Description
Integration	INSERT_ION_OUTBOX_ENTRY	A save-to-entity task template that contains metadata to insert a COR_OUTBOX_ENTRY entity. The C_XML parameter is the value responsible for the XML body of a particular BOD.
Integration	INSERT_ION_OUTBOX_HEADER_BODTYPE	A save-to-entity task template that contains metadata to insert the BOD header BodType value to COR_OUTBOX_HEADERS ION database table.
Integration	INSERT_ION_OUTBOX_HEADER_FROMLOGICALID	A save-to-entity task template that contains metadata to insert the BOD header FromLogicalId value to COR_OUTBOX_HEADERS ION database table.
Integration	INSERT_ION_OUTBOX_HEADER_MESSAGEID	A save-to-entity task template that contains the metadata to insert the BOD header MessageId value to COR_OUTBOX_HEADERS ION database table.
Integration	INSERT_ION_OUTBOX_HEADER_TENANTID	A save-to-entity task template that contains metadata to insert the BOD header TenantId value to COR_OUTBOX_HEADERS ION database table.
Integration	INSERT_ION_OUTBOX_HEADER_TOLOGICALID	A save-to-entity task template that contains metadata to insert the BOD header ToLogicalId value to COR_OUTBOX_HEADERS ION database table.
Integration	INSERT_PLM_ITEM_TYPE	A save-to-entity task template that contains the metadata to insert values to PLM table named OPITEMTEMP. This table holds the Item Type records.

Task type	Template	Description
	INSERT_PLM_SUPPLIER	A save-to-entity task template that contains the metadata to insert values to PLM table named SUPPLIER. This table holds the Supplier records.
Integration	INSERT_STYLE	An update-entity task template that updates the NETWEIGHT, GROSSWEIGHT, FREECAPUNIT of a related PLM STYLE table record.
Integration	INSERT_SYNC_LOG	An integration task template that contains the metadata to save the data time of the Integration Flow for easier keeping track of sent items during integration. Saves in SCAH SYNCHLOG database table.
Integration	TRANSFORM_FROM_ACKNOWLEDGE_ITEMMASTER	A transform-from-XML task template that extracts the ResponseExpression and ID values of AcknowledgeItemMaster BOD.
Integration	TRANSFORM_FROM_ACKNOWLEDGE_ITEMMASTER_ON_REJECTED	A transform-from-XML task template that extracts the ResponseExpression and Reason values of AcknowledgeItemMaster BOD.
Integration	TRANSFORM_FROM_SYNCITEMMASTER	A transform-from-XML-element task template that extracts values from a SyncItemMaster BOD.
Integration	TRANSFORM_TO_BUYER_REFERENCE	A transform-to-XML-element task template that transforms its parameter values to an XML element-<BuyerReference> with its attribute 'IDs/ID@accountingEntity' defaulted to an accounting entity. The generated element will be appended to a BOD to a specified path 'DataArea/ItemMaster/ItemLocation/UserArea'.

Task type	Template	Description
Integration	TRANSFORM_TO_PROPERTY_AREA_PRINT_FEATURE	An integration task template that contains the metadata to transform the XML to a specific area in the BOD specifically placing the Property Area Print Feature value in the Code Definition element.
Integration	TRANSFORM_TO_PROPERTY_AREA_PRINT_OPTION	An integration task template that contains the metadata to transform the xml to a specific area in the BOD specifically placing the Property Area Print Option value in the Code Definition element.
Integration	TRANSFORM_TO_PROPERTY_AREA_FEATURE_TYPES	An integration task template that contains the metadata to transform the xml to a specific area in the BOD specifically placing the Property Area Feature Types value in the Code Definition element.
Integration	TRANSFORM_TO_ITEMMASTER	A transform-to-BOD task template that transforms its parameter values to ProcessItemMaster BOD.
Integration	TRANSFORM_TO_SPECIFICATION_ID	A transform-to-XML task template that transforms its parameter values to an XML element <Specification>. The generated element will be appended to a BOD to a specified path 'DataArea/ItemMaster/ItemMasterHeader'.
Integration	TRANSFORM_TO_SPECIFICATION_PURCHASEPRICE	A transform-to-XML-element task template that transforms its parameter values to an XML element <Specification>. The generated element will be appended to a BOD to a specified path 'DataArea/ItemMaster/ItemMasterHeader'.

Task type	Template	Description
Integration	TRANSFORM_TO_SPECIFICATION_SALESPRICE	A transform-to-XML-element task template that transforms its parameter values to an XML element <Specification>. The generated element will be appended to a BOD to a specified path 'DataArea/ItemMaster/ItemMasterHeader'.
Integration	TRANSFORM_FROM_SYNCSUPPLIERPARTYMASTER	A transform-from-XML- element task template that extracts values from a SyncSupplierPartyMaster BOD.
Integration	TRANSFORM FROM_SYNCITEMCODEDEF	A transform-from-XML-element task template that extracts values from a SyncCodeDefinitionBOD.
Integration	TRANSFORM_TO_CLASSIFICATION	A transform-to-XML-element task template that transforms its parameter values to an XML element <Classification>. The generated element will be appended to a BOD to a specified path 'DataArea/ItemMaster/ItemMasterHeader'.
Integration	TRANSFORM_TO_CLASSIFICATION_ITEMLOCATION	A transform-to-XML-element task template that transforms its parameter values to an XML element <ItemLocation>. The generated element will be appended to a BOD and to aspecified path 'DataArea/ItemMaster/ ItemLocation'.
Integration	TRANSFORM_TO_DESCRIPTION_SHORT_NAME	A transform-to-XML-element task template that transforms its parameter values to an XML element--<Description> with its attribute 'Description@type' set to 'ShortName'. The generated element will be appended to a BOD to a specified path 'DataArea/ItemMaster/ItemMasterHeader'.

Task type	Template	Description
Integration	TRANSFORM_TO_HTS_CUSTOMSTAT	A transform-to-XML element that transforms its parameter values to an XML element <Property> with NameValue@name-CustomsStatNumber. The generated element will be appended to BOD to a path 'DataArea/ItemMaster/ItemLocation/UserArea'.
Integration	TRANSFORM_TO_ITEMID	An integration task template that contains the metadata to transform the xml to a specific area in the BOD specifically placing the ITEMID value.
Integration	TRANSFORM_TO_PLANNER_REFERENCE	A transform-to-XML-element task template that transforms its parameter values to an XML element <PlannerReference> with its attribute 'IDs/ID@accountingEntity' defaulted to an accounting entity. The generated element will be appended to the BOD to a path 'DataArea/ItemMaster/ItemLocation/UserArea'.
Integration	TRANSFORM_TO_PROCESS_CODE_DEFINITION	A transform-to-BOD task template that transforms parameter values into ProcessCodeDefinition BOD.
Integration	TRANSFORM_TO_PROCESS_ITEMMASTER_ADD	A transform-to-BOD task template that transforms its parameter values to ProcessItemMaster BOD.
Integration	TRANSFORM_TO_PROCESS_ITEMMASTER_ADD_NW	An integration task template that contains the metadata to transform the xml for adding the ProcessItemMaster BOD for no warehouse.

Task type	Template	Description
Integration	TRANSFORM_TO_PROPERTY_AREA_FEATURE_TYPE	A transform-to-XML-element task template that transforms its parameter values to an XML element <Property> with its attribute 'NameValue@name' defaulted to StyleFeatureTypes. The generated element will be appended to ProcessCodeDefinition BOD to a specified path 'DataArea/ CodeDefinition'.
Integration	TRANSFORM_TO_PROPERTY_AREA_OPTION_FLOW2	An integration task template that contains the metadata to transform the xml to a specific area in the BOD specifically placing the Property Area Option values in the Code Definition element.
Integration	TRANSFORM_TO_SPECIFICATION	A transform-to-XML-element task template that transforms its parameter values to an XML element-<Specification> with its attribute 'ID@accountingEntity' defaulted to an accounting entity. The generated element will be appended to a BOD to a specified path 'DataArea/ItemMaster/ItemMasterHeader'.
Integration	TRANSFORM_TO_SUPPLIER	A transform-to-XML-element task template that transforms its parameter values to an XML element <DefaultSource> with its attribute 'SupplierParty/PartyIDs/ID@accountingEntity' defaulted to an accounting entity. The generated element will be appended to a BOD to a specified path: 'DataArea/ItemMaster/ItemLocation/ProcurementParameters'.

Task type	Template	Description
Integration	TRANSFORM_TO_SUPPLIER_NEW	A transform-to-XML-element task template that transforms its integration TRANSFORM_TO-SUPPLIER parameter values to an XML element <Default-Source> with its attribute 'SupplierParty/PartyIDs/ID@accountingEntity" that is defaulted to an accounting entity. The generated element will be appended to a BOD to this specified path: DataArea/ItemMaster/ItemMasterHeader/ProcurementParameters.
Integration	TRANSFORM_TO_USER_AREA	A transform-to-XML-element task template that transforms its parameter values to an XML element <Property>. The generated element will be appended to a BOD to path 'DataArea/ItemMaster/ItemMasterHeader/UserArea'.
Integration	TRANSFORM_TO_USER_AREA_INTERFACE_STYLE_NUMBER	A transform-to-XML-element task template that transforms its parameter values to an XML element-<Property> with its attribute 'NameValue@name' defaulted to InterfaceStyleNumber. The generated element will be appended to the BOD to a path 'DataArea/ItemMaster/ItemLocation/UserArea'.
Integration	TRANSFORM_TO_USER_AREA_SEASON_CODE	An integration task template that contains the metadata to transform the xml to a specific area in the BOD specifically placing the Property Area Season Code values in the Item-User Area element.

Task type	Template	Description
Integration	TRANSFORM_TO_USER_AREA_SEASON_CONTROL	A transform-to-xml-element task template that transforms its parameter values to an XML element <Property> with its attribute 'NameValue@name' defaulted to 'SeasonControl'. The generated element will be appended to a BOD with path 'DataArea/ItemMaster/ItemMasterHeader/UserArea'.
Integration	TRANSFORM_TO_USER_AREA_SKU_FEATUREX	A transform-to-XML-element task template that transforms its parameter values to an XML element <Property> with its attribute 'NameValue@name' defaulted to 'StyleFeatureX'. The generated element will be appended to the BOD to a path 'DataArea/ItemMaster/ItemMasterHeader/UserArea'.
Integration	TRANSFORM_TO_USER_AREA_SKU_FEATUREY	A transform-to-XML element that transforms equivalent Y features as parameter values to an XML element <Property> then appends them to <UserArea> in BOD.
Integration	TRANSFORM_TO_USER_AREA_SKU_FEATUREZ	A transform-to-XML-element task template that transforms its parameter values to an XML element <Property> with attribute 'NameValue@name' defaulted to 'StyleFeatureZ'. The generated element will be appended to the BOD to a path 'DataArea/ItemMaster/ItemMasterHeader/UserArea'.
Integration	TRANSFORM_TO_USER_AREA_SKU_OPTIONX	A transform-to-XML element that translates equivalent X Options as parameter values to an XML element <Property>, which is then later appended to BOD path 'DataArea/ItemMaster/ItemMasterHeader/UserArea'.

Task type	Template	Description
Integration	TRANSFORM_TO_USER_AREA_SKU_OPTIONX_SEQ	A transform-to-XML-element task template that transforms equivalent X Option sequence as parameter values to an XML element <Property>, which is then later appended to the BOD with a path DataArea/ItemMaster/ItemMasterHeader/UserArea'.
Integration	TRANSFORM_TO_USER_AREA_SKU_OPTIONY	A transform-to-XML-element task template that transforms equivalent Y Options as parameter values parameter values to an XML element-<Property> with its attribute 'NameValue@name' defaulted to 'StyleOptionY'. The generated element will be appended to the BOD to a path 'DataArea/ItemMaster/ItemMasterHeader/UserArea'.
Integration	TRANSFORM_TO_USER_AREA_SKU_OPTIONY_SEQ	A transform-to-XML-element task template that transforms equivalent Y Options as parameter values parameter values to an XML element-<Property> with its attribute 'NameValue@name' defaulted to 'StyleOptionYSequence'. The generated element will be appended to the BOD to a path 'DataArea/ItemMaster/ItemMasterHeader/UserArea'.
Integration	TRANSFORM_TO_USER_AREA_SKU_OPTIONZ	A transform-to-XML-element task template that transforms equivalent Z Options as parameter values to an XML element--<Property> with its attribute 'NameValue@name' defaulted to 'StyleOptionZ'. The generated element will be appended to the BOD to a path 'DataArea/ItemMaster/ItemMasterHeader/UserArea'.

Task type	Template	Description
Integration	TRANSFORM_TO_USER_AREA_SKU_OPTIONZ_SEQ	A transform-to-XML-element task template that transforms equivalent Z Option Sequence as parameter values to an XML element--<Property>. The generated XML element will then be appended to a BOD with path 'DataArea/ItemMaster/ItemMasterHeader/UserArea'.
Integration	TRANSFORM_TO_USER_AREA_STYLEFEATURE	A transform-to-XML-element task template that transforms its parameter values to an XML element <Property> with its attribute 'NameValue@type' defaulted to StringType. The generated element will be appended to BOD to a path 'DataArea/ItemMaster/ItemLocation/UserArea'.
Integration	TRANSFORM_TO_USER_AREA_STYLEITEMTYPE	A transform-to-XML-element task template that transforms its parameter values to an XML element <Property> with its attribute 'NameValue@name' defaulted to 'StyleItemType'. The generated element will be appended to a BOD to path 'DataArea/ItemMaster/ItemMasterHeader/UserArea'.
Integration	TRANSFORM_TO_WAREHOUSE_LOCATION	A transform-to-XML element that translates the paramaters to an <ID> XML element, which is then later appended to a BOD path 'DataArea/ItemMaster/ItemLocation/WarehouseLocation'.
Integration	UPDATE_FAIL_ION_INBOX	An update-entity task template that changes the related ION table record COR_INBOX_ENTRY to C_WAS_PROCESSED = 2. This marks that the particular ION inbox entry has already been processed but failed.

Task type	Template	Description
Integration	UPDATE_FAIL_SEND_STYLE_SKU_WORKUNIT	An update-entity task template that changes the related PLM table record SKU to ISPHANTOM = 1. This marks that the particular SKU has been integrated.
Integration	UPDATE_FAIL_SEND_STYLE_WORKUNIT	An update-entity task template that changes the related PLM table record STYLE to INTGSTATUSMHS to Not Successful. This marks that the particular style has not been successfully integrated.
Integration	UPDATE_ION_INBOX	An update-entity task template that changes the related ION table record COR_INBOX_ENTRY to C_WAS_PROCESSED = 1. This marks that the particular ION inbox entry has already been processed.
Integration	UPDATE_SEND_STYLE_SKU_WORKUNIT	An update-entity task template that changes the related PLM table record SKU to ISPHANTOM to 0. This marks that the particular SKU has been integrated.
Integration	UPDATE_SEND_STYLE_WORKUNIT	An update-entity task template that changes the related PLM table record STYLE to ISPHANTOM to 0 and INTGSTATUSMHS to Successful. This marks that the particular style has been integrated.
Integration	UPDATE_STYLE_FROM_ACK	An integration task template that contains the metadata to update the Style record of the integration status. Updates value in STYLE database table.

Task type	Template	Description
Integration	UPDATE_TO_MHS001MI	An MI-call-task template that contains the metadata to update data by transacting to a MI Program/Transaction –MHS001MI/ChgIntItmMst. **Note:** This template is contained in an ION flow task. Any changes made to the current ION flow task must consider updating this template as well.
Integration	UPDATE_TO_MHS002MI	An MI-call-task template that contains the metadata to update data by transacting to a MI Program/Transaction –MHS002MI/ChgIntItmWhs. **Note:** This template is contained in an ION flow task. Any changes made to the current ION flow task must consider updating this template as well.
Integration	UPDATE_TO_PDS055MI	An integration task template that contains the metadata to update values through the M3 Api web service in MI Program PDS055MI.
QueryTransform	QUERY_TRANSFORM_TO_PROPERTY_AREA_STYLE_OPTIONS_SEND_STYLE_X	A Query Transform task template that contains the metadata to get Style Size values and transform them to Property areas in the BOD. Gets the value from V_INTGSTYLESIZE database table.
QueryTransform	QUERY_TRANSFORM_TO_PROPERTY_AREA_STYLE_OPTIONS_SEND_STYLE_Y	A Query Transform task template that contains the metadata to get Style Color values and transform them to Property areas in the BOD. Gets the value from V_INTGSTYLECOLOR database table.

Task type	Template	Description
QueryTransform	QUERY_TRANSFORM_TO_PROPERTY_AREA_STYLE_OPTIONS_SEND_STYLE_Z	A Query Transform task template that contains the metadata to get Style Characteristic values and transform them to Property areas in the BOD. Gets the value from V_INTGSTYLECHARACTERISTIC database table.
QueryTransform	QUERY_TRANSFORM_TO_USER_AREA_STYLEFEATUREX	A Query Transform task template that contains the metadata to get Feature group X dimension values and transform them to User areas in the BOD. Gets the value from V_INTGSTYLEFEATURE database table.
QueryTransform	QUERY_TRANSFORM_TO_USER_AREA_STYLEFEATUREY	A Query Transform task template that contains the metadata to get Feature group Y dimension values and transform them to User areas in the BOD. Gets the value from V_INTGSTYLEFEATURE database table.
QueryTransform	QUERY_TRANSFORM_TO_USER_AREA_STYLEFEATUREY_FLOW_2	A Query Transform task template that contains the metadata to get Style Color Feature group values and transform them to User areas in the BOD. Gets the value from V_INTGSTYLECOLOR database table.
QueryTransform	QUERY_TRANSFORM_TO_USER_AREA_STYLEFEATUREZ	A Query Transform task template that contains the metadata to get Feature group Z dimension values and transform them to User areas in the BOD. Gets the value from V_INTGSTYLEFEATURE database table.
Cleanup	CLEAN_UP_SYNC_ITEM_MASTER_WORKUNIT	A Clean up task template that contains the metadata to perform a clean-up operation or delete records in the ION database table. Deletes records in COR_INBOX_ENTRY table in ION database.

Integration flows

All types of integration flows can be configured in Connector Toolbox, but by default, these integration flow templates are already set and are available in the Connector Toolbox with settings, parameters, and values to launch an integration flow successfully. You can reuse, update and delete existing integration flow templates, create a new integration flow from an existing integration flow template, or create a new integration flow.

Default integration flows

These default integration flows use the Infor Fashion PLM Style Number as the Feature Code, and the Infor Fashion PLM Color Code as the Option Code:

Integration flow	Description
ION_UPDATE_DISCRETE_ITEM	Integrates updates to discrete items through ION BODs
ION_SEND_STYLE	Executes style integration through ION BODs
ACK_ITEM_MASTER_FLOW	Executes the Acknowledge Item Master flow through ION BODS to be sent back items to PLM
ION_UPDATE_STYLE	Integrates updates to styles through ION BODs
ION_SEND_DISCRETE_ITEM	Executes the integration of discrete items through ION BODs
ION_SEND_STYLE_SKU	Sends SKUs for integration through ION BODs
ION_SYNCITEMMASTER_ADD	Processes incoming ION SyncItemMaster BOD with the specific verb 'Add'
CLEANUP_SYNC_ITEM_MASTER_ADD	
ION_SYNCITEMMASTER_ADD_SINKTASK	
ION_SYNCCODEDEFINITION_ITEM-TYPES	Imports item types to Infor Fashion PLM from incoming ION SyncCodeDefinition
ION_SYNCSUPPLIERPARTYMASTER	Imports suppliers to Infor Fashion PLM from incoming ION SyncSupplierPartyMaster

Secondary integration flows

These secondary integration flows use the Infor Fashion PLM Color Code as the Feature Code and Option Code:

Integration flow	Description
ION_SEND_STYLE_FLOW_2	Executes the integration of Style items through ION BODS
ION_SEND_STYLE_SKU_FLOW_2	Executes the integration of Style SKU items through ION
ION_UPDATE_STYLE_FLOW_2	Executes updates of Style items through ION BODS

Additional integration flows

Integration flow	Description
ION_UPDATE_DISCRETE_ITEM_NW	Executes updates of discrete items through ION BODs without the warehouse
ION_UPDATE_STYLE_NW	Executes updates of Style items through ION BODs without the warehouse
ION_SEND_STYLE_SKU_NW	Executes the integration of Style SKU items through ION BODS without the warehouse
ION_SEND_DISCRETE_ITEM_NW	Executes the integration of discrete items through ION without the warehouse
ION_SEND_STYLE_NW	Executes the integration of Style items through ION BODS without the warehouse

Chapter 4: Designing integration flows

Learn how to manage integration tasks templates and create integration flows.

Managing integration task templates

Integration tasks are managed as integration task templates. Integration task templates are used when creating tasks in an integration flow. You can create new and update existing integration task templates that can be reused when adding or updating tasks in an integration flow. You can create, update, or delete integration task templates by using the Manage Templates module of Connector Toolbox.

Manage connector templates

1. In the navigation pane, select **Integration Designer > Manage Templates**.
2. Specify this information:

   Task type
   Select Connector as the task type.

   Templates
   Select a template from the list.

   - To copy the selected template, click **Copy** and specify a template name.
   - To delete the selected template, click **Delete**.
   - To create a new template, click **New** and specify a template name.

3. On the Parameters tab, select **Click here to add new row**.
4. Specify this information or update the existing parameters:

   Name
   Specify a parameter name.

   Field Type
   Select a field type from the list.

   Value
   Specify a text value or select a system value from the list.

5 On the Conditions table, specify this information or update the existing conditions:

Field Name
Specify a field name.

Condition Type
Select a condition type from the list.

Condition Value
Specify a condition value.

6 Click **Save**.

Note: To see the integration flows that uses the selected integration template, click the **Where Used** tab.

Manage query templates

1 In the navigation pane, select **Integration Designer > Manage Templates**.

2 Specify this information:

Task type
Select Query as the task type.

Templates
Select a template from the list.

- To copy the selected template, click **Copy** and specify a template name.
- To delete the selected template, click **Delete**.
- To create a new template, click **New** and specify a template name.

Resource type
Select any of these resource types from the list:

- Entity

  This resource type performs query tasks from a database.
- M3WS

  This resource type performs a query tasks from M3 Web Service.

Multiple results
Select to enable your query to return multiple records.

Is Polling
Select to enable polling support. If selected, the query continually performs the task until specified conditions occur.

Polling Timeout
Specify a polling timeout in milliseconds. The default timeout is 60000 milliseconds.

Polling Interval
Specify the polling interval in milliseconds. The default interval, if not specified, is 5000 milliseconds.

Polling Interval will allow the system to poll the record for a certain interval before requesting it again, rather than a successive request of polls which affects the performance.

3 Based on the selected resource type, specify this information:

- Entity resource type:

  Entity Alias
  Select an entity alias from the list.

  You can add entity aliases in the Database Entities module.

  See Identifying supported database tables on page 17.

  Note: Entity aliases depend on the selected connection string.

  Entity
  Select an entity from the list.
  Note: The list of entities are based on the specified connection string and entity alias.

- M3WS resource type:

  Program
  Select a program from the list.
  Note: The list contains the available and supported MI programs. To add an MI program, specify an MI program name and click **Add**.

  Transaction
  Select a transaction from the list.
  Note: The list contains all available transactions for the selected MI program.

4 On the Parameters tab, specify this information or update the existing parameters:

Name
Specify a parameter name.

Column Name
Select a column name from the list.
Note: This column is available only if you select Entity as the resource type.

Field Name
Select a field name from the list.
Note: This column is available only if you select M3WS as the resource type.

Default Value
Specify a default value.

5 On the Conditions table, specify this information or update the existing conditions:

Field Name
Specify a field name.

Condition Type
Select a condition type from the list.

Condition Value
Specify a condition value.

Use as Variable
Select if you want to use the field name as variable to be used when mapping flows. This field is automatically checked if the Condition Value contains the Field Name preceded with a hash.

6. Select a sorting option in the Sort Info section and specify a field where the sorting will apply. To disable sorting, leave this field blank.
7. Click **Save**.

   Note: To see the integration flows that uses the selected integration template, click the **Where Used** tab.

Manage integration templates

1. In the navigation pane, select **Integration Designer > Manage Templates**.
2. Specify this information:

   Task type
   Select Integration as the task type.

   Templates
   Select a template from the list.
   - To copy the selected template, click **Copy** and specify a template name.
   - To delete the selected template, click **Delete**.
   - To create a new template, click **New** and specify a template name.

   Integration Type
   Select any of these integration types from the list:
   - Entity
   - BOD

     This integration type performs a BOD/XML type operation.
   - BODArea

     This integration type maps the value from the flow to a specific BOD or XML area.
   - M3WS

     This integration type performs an MI transaction operation.
3. Based on the selected integration type, specify this information:
   - Entity integration type:

     Entity Alias
     Select an entity alias from the list.

You can add entity aliases in the Database Entities module.

See Identifying supported database tables on page 17.

Note: Entity aliases depend on the selected connection string.

Entity
Select an entity from the list.

Transaction Type
Select a transaction type from the list.
Note: The available transaction types are insert and update.

If Exist
Select skip or update from the list.

If you select skip and the condition on the Condition table is satisfied, the integration task will perform a skip operation and the data will not be processed.

If you select update and the condition on the Condition table is satisfied, data with the corresponding values will perform an update instead of inserting the data or record.

Leave this field blank if you don't want to perform any of the available operations. Values selected on this operation will look-up from the Conditions table.

- BOD integration type:

  BOD Type
  Select any of these BOD types from the list:

  - Acquire from XML Parameter

    This BOD type obtains the BOD/XML value from the xml parameter within the flow.

  - Acquire from File

    This BOD type obtains the BOD/XML value from a file.

    Note: This is applicable for Work unit type FileUnit.

  - Dump to XML Parameter

    This BOD type saves the BOD/XML value to the xml parameter within the flow.

  - Dump to File

    This BOD type saves the BOD/XML value to a file.

    Note: This is applicable for Work unit type FileUnit.

  Integrate From
  Specify a path where the values are acquired.
  Note: This field is available only if you select Acquire from XML Parameter or Acquire from File as the BOD type.

  Integrate To
  Specify a path where the values are dumped.
  Note: This field is available only if you select Dump to XML Parameter or Dump to File as the BOD type.

Folder
Specify a folder path where the XML files are located. Click the ellipsis button to browse for a folder.
Note: This field is available only if you select Dump to File as the BOD type.

Prefix
Specify a prefix to be used for the file names of the XML files.
Note: This field is available only if you select Dump to File as the BOD type.

Null Parameter Skip
If selected, parameters that contain null values will be skipped during the processing of the integration task.

- BODArea integration type:

  Note: In BODArea integration types, you can specify a condition field to target in which element the BODArea is placed. A parameter should be created first then indicate the parameter in the Conditions area for proper placement of the XML element.

  Integrate To
  Specify a path where the values are dumped.

  Null Parameter Skip
  If selected, parameters that contain null values will be skipped during the processing of the integration task.

- M3WS integration type:

  Program
  Select a program from the list.
  Note: The list contains the available and supported MI programs. To add an MI program, specify an MI program name and click **Add**.

  Transaction
  Select a transaction from the list.
  Note: The list contains all available transactions for the selected MI program.

4 On the Parameters tab, specify this information or update the existing parameters:

Note: When integrating to BOD or BODArea types and specifying an XML attribute or '@ parameter', place the parameter below the XML element where the parameter should be generated. For example, to create Item/Item@ItemId, the Item/Item parameter field should precede the parameter.

Name
Specify a parameter name.

Column Name
Select a column name from the list.

Field Name
Specify a field name.
Note: This field is available only if you select M3WS as the integration type.

Inline Type
Select an inline type from the list.
Note: This field is available only if you select Entity as the integration type.

Default Value
Specify a default value.

5 On the Conditions table, specify this information or update the existing conditions:

Field Name
Specify a field name.

Condition Type
Select a condition type from the list.

Condition Value
Specify a condition value.

Use as Variable
Select if you want to use the field name as variable to be used when mapping flows. This field is automatically checked if the Condition Value contains the Field Name preceded with a hash.

6 Click **Save**.

Note: To see the integration flows that uses the selected integration template, click the **Where Used** tab.

Manage query transform templates

1 In the navigation pane, select **Integration Designer > Manage Templates**.

2 Specify this information:

Task type
Select QueryTransform as the task type.

Templates
Select a template from the list.

- To copy the selected template, click **Copy** and specify a template name.
- To delete the selected template, click **Delete**.
- To create a new template, click **New** and specify a template name.

Resource type
By default, the resource type is set to Entity. This resource type performs query tasks from a database.

Multiple results
Select to enable your query to return multiple records.

Is Polling
Select to enable polling support. If selected, the query continually performs the task until specified conditions occur.

Polling Timeout
Specify a polling timeout in milliseconds. The default timeout is 6000 milliseconds.

Integration Type
By default, the integration type is set to BODArea. This integration type performs a BOD/XML type operation.

3 Specify entity information:

Entity Alias
Select an entity alias from the list.

You can add entity aliases in the Database Entities module.

See Identifying supported database tables on page 17.

Note: Entity aliases depend on the selected connection string.

Entity
Select an entity from the list.
Note: The list of entities are based on the specified connection string and entity alias.

4 Based on the selected integration type, specify this information:

- Entity integration type:

  Entity Alias
  Select an entity alias from the list.

  You can add entity aliases in the Database Entities module.

  See Identifying supported database tables on page 17.

  Note: Entity aliases depend on the selected connection string.

  Entity
  Select an entity from the list.

  Transaction Type
  Select a transaction type from the list.

  If Exist
  Select skip or update from the list.
  Note: This field is available only if you select Insert as the transaction type.

- BOD integration type:

  BOD Type
  Select any of these BOD types from the list:

  - Acquire from XML Parameter

    This BOD type obtains the BOD/XML value from the xml parameter within the flow.

  - Acquire from File

This BOD type obtains the BOD/XML value from a file.

Note: This is applicable for Work unit type FileUnit.

- Dump to XML Parameter

  This BOD type saves the BOD/XML value to the xml parameter within the flow.

- Dump to File

  This BOD type saves the BOD/XML value to a file.

  Note: This is applicable for Work unit type FileUnit.

Integrate From
Specify a path where the values are acquired.
Note: This field is available only if you select Acquire from XML Parameter or Acquire from File as the BOD type.

Integrate To
Specify a path where the values are dumped.
Note: This field is available only if you select Dump to XML Parameter or Dump to File as the BOD type.

Folder
Specify a folder path where the XML files are located. Click the ellipsis button to browse for a folder.
Note: This field is available only if you select Dump to File as the BOD type.

Prefix
Specify a prefix to be used for the file names of the XML files.
Note: This field is available only if you select Dump to File as the BOD type.

Null Parameter Skip

If selected, parameters that contain null values will be skipped during the processing of the integration task.

This check box is automatically selected if the task type is QueryTransform.

- BODArea integration type:

  Entity Alias
  Select an entity alias from the list.

  You can add entity aliases in the Database Entities module.

  See Identifying supported database tables on page 17.

  Note: Entity aliases depend on the selected connection string.

  Entity
  Select an entity from the list.

  BOD Type
  Select any of these BOD types from the list:

  - Acquire from XML Parameter

    This BOD type obtains the BOD/XML value from the xml parameter within the flow.

  - Acquire from File

This BOD type obtains the BOD/XML value from a file.

Note: This is applicable for Work unit type FileUnit.

- Dump to XML Parameter

  This BOD type saves the BOD/XML value to the xml parameter within the flow.

- Dump to File

  This BOD type saves the BOD/XML value to a file.

  Note: This is applicable for Work unit type FileUnit.

Integrate From
Specify a path where the values are acquired.
Note: This field is available only if you select Acquire from XML Parameter or Acquire from File as the BOD type.

Integrate To
Specify a path where the values are dumped.
Note: This field is available only if you select Dump to XML Parameter or Dump to File as the BOD type.

Folder
Specify a folder path where the XML files are located. Click the ellipsis button to browse for a folder.
Note: This field is available only if you select Dump to File as the BOD type.

Prefix
Specify a prefix to be used for the file names of the XML files.
Note: This field is available only if you select Dump to File as the BOD type.

Null Parameter Skip

If selected, parameters that contain null values will be skipped during the processing of the integration task.

This check box is automatically selected if the task type is QueryTransform.

- M3WS integration type:

  Program
  Select a program from the list.
  Note: The list contains the available and supported MI programs. To add an MI program, specify an MI program name and click **Add**.

  Transaction
  Select a transaction from the list.
  Note: The list contains all available transactions for the selected MI program.

5 On the Parameters tab, specify this information or update the existing parameters:

Name
Specify a parameter name.

Field Name
Specify a field name.

Note: This field is available only if you select M3WS as the integration type.

Column Name
Select a column name from the list.

Default Value
Specify a default value.

6 On the Conditions table, specify this information or update the existing conditions:

Field Name
Specify a field name.

Condition Type
Select a condition type from the list.

Condition Value
Specify a condition value.

7 Click **Save**.

Note: To see the integration flows that uses the selected integration template, click the **Where Used** tab.

Manage cleanup templates

1 In the navigation pane, select **Integration Designer > Manage Templates**.

2 Specify this information:

Task type
Select Cleanup as the task type.

Templates
Select a template from the list.

- To copy the selected template, click **Copy** and specify a template name.
- To delete the selected template, click **Delete**.
- To create a new template, click **New** and specify a template name.

Entity Alias
Select an entity alias from the list.

You can add entity aliases in the Database Entities module.

See Identifying supported database tables on page 17.

Note: Entity aliases depend on the selected connection string.

Entity
Select an entity from the list.

3 On the Parameters tab, specify this information or update the existing parameters:

Name
Specify a parameter name.

Column Name
Select a column name from the list.

4 On the Conditions table, specify this information or update the existing conditions:

Field Name
Specify a field name.

Condition Type
Select a condition type from the list.

Condition Value
Specify a condition value.

5 Click **Save**.

Note: To see the integration flows that uses the selected integration template, click the **Where Used** tab.

Managing integration flows

In the Connector Toolbox, you can create a new integration flow or manage existing integration flow templates. Use the Flow Designer to manage integration flows.

Creating an integration flow

To create an integration flow, you must define information about the integration flow, create tasks, fail tasks, and validation rules.

Create a new integration flow

1 In the navigation pane, select **Integration Designer > Flow Designer**.

2 In the Integration Flow field, specify a name and then click **Add**.

A default Integration Flow Information task is added to the integration flow.

3 Optionally, on the Task Flow tab, click **Integration Flow Information** and specify this information:

Task Name
Specify a name.

Task Details
Specify task details.

Integration Done Message
Specify a message indicating that the task has been completed.

4 Click **Save**.

Create a task

Before you start creating a task, ensure that you first select the previous task on the Task Flow tab or the Fail Task Flow tab to denote that the new task should follow immediately after it.

1 On the Task Flow tab, click **New Task**.

If a task fails in the integration flow, you can manage how the integration flow will proceed by defining tasks on the Fail Task Flow tab. To manage the fail task flow, click the **Fail Task Flow** tab.

2 On the Details panel, specify this information:

Task Type
Select a task type.

Templates
Select an integration task template from the list.

See Integration task types on page 18.

Task Name
Specify a name.

Fail Task
Select a fail task from the list.

Available options include tasks defined on the Fail Task tab for the selected integration flow.

The list contains the tasks defined on the Fail Task Flow tab of the selected integration flow. All tasks that follow the selected fail task will be ran.

3 On the Field Mapping tab, specify this information:

Source
Select a source from the list.

Available options include parameters from all previous tasks.

Destination
Select a destination from the list.

Available options include all parameters from the selected task.

Condition Type
Select a condition type from the list.

Condition Value
Specify a condition value.

4 Click **Save**.

Note: To delete a task, select the task on the Flow section and click **Delete Task**.

To preview a task, click the **Preview** tab.

Create fail tasks

Fail tasks are added to the Fail Task Flow tab to indicate a fail flow execution if the selected task fails or does not meet the conditions for the integration.

1 Select a task.

2 Right-click and select **New Fail Task**.

The fail task is added to the Fail Task Flow tab.

Note: You must have at least one fail task that is available in the Fail Task Flow tab before you create a fail flow task.

To create a fail flow task, see Creating an integration flow on page 48.

Create validation rules

You can extend the functionality of an integration task by adding validation rules in the form of JavaScript. A pre-validation script is ran before starting an integration task and a post validation script is ran after completing an integration task.

1 Click the **Validation Rules** tab.

2 To add a pre-validation script, on the Pre-Validation Script section, click **Add**.

To add a post-validation script, on the Post-Validation Script section, click **Add**.

The Script dialog box is displayed.

3 Specify a script in this format:

```
var brokenrule;
function Evaluate(n, x)
{
    if (n==x)
    {
    return true;
    }
    else
    {
    brokenrule = "is not equal!";
    return false;
    }
}
Evaluate(membername.Param1, membername.Param2)
```

4 Modify the conditions and the variables on the script and on the second evaluate statement, replace the values of the parameters, where *membername* is the name of the integration task and *Param* is a parameter on the integration task.

   Note: The parameters on the evaluate statement must be from the currently selected task or from previous tasks.

5 Click **Save**.

   To edit an existing validation rule, double-click the validation rule. To delete an existing validation rule, select a validation rule and click **Remove**.

   You can add multiple pre-validation scripts and post-validation scripts in an integration task.

   The scripts are ran according to the sequence that is defined in the Pre-Validation Script section and the Post-Validation Script section.

Managing existing integration flows

1 In the navigation pane, select **Integration Designer > Flow Designer**.

2 In the Integration Flow field, select an integration flow from the list.

Edit tasks of an existing integration flow

1 Select the task on the Flow section.

2 Update the task details on the Details section.

3 Click **Save**.

Create copy of an existing integration flow

1 Click **Copy Flow**.

2 Specify a name for the integration flow.

3 Click **Copy**.

Delete an existing integration flow

1 Select the task on the Flow section.

2 Click Delete Flow.

Chapter 5: Setting up work units

Learn the concepts and procedures on setting up work units.

Worker overview

Workers are background monitoring processes that perform batch processing. They are polling jobs that execute the integration flows that you created in Integration Designer based on conditions.

Worker templates

You can create your own workers to execute integration flows, but by default, these worker templates are already set with conditions and schedule settings to execute integration flows successfully:

Worker template	Description
SYNC_ITEM_MASTER_WORKER	Executes the integration flow template ION_SYNCITEMMASTER_ADD
SEND_DISCRETE_ITEM_WORKER	Executes the integration flow ION_SEND_DISCRETE_ITEM
UPDATE_DISCRETE_ITEM_WORKER	Executes the integration flow template ION_UPDATE_DISCRETE ITEM
SEND_STYLE_WORKER	Executes the integratin flow template ION_SEND_STYLE
UPDATE_STYLE_WORKER	Executes the integration flow template ION_UPDATE_STYLE
SEND_STYLE_SKU_WORKER	Executes the integration flow template ION_SEND_STYLE_SKU
IMPORT_ITEMTYPE_WORKER	Executes the integration flow template ION_SYNCCODEDEFINITION_ITEMTYPES

Worker template	Description
IMPORT_SUPPLIER_WORKER	Executes the integration flow template ION_SYNCSUPPLIERPARTYMASTER
SYSTEM_RESERVED_DELETE_FLOW_STATE_FILES_AT_ERROR_STATE	Executes the deletion of error state files
SYSTEM_RESERVED_DELETE_FLOW_STATE_FILES_AT_FAILED_STATE	Executes the deletion of failed state files
SYSTEM_RESERVED_DELETE_FLOW_STATE_FILES_AT_FAILED_STATE	Executes the deletion of finished state files
ACK_ITEM_MASTER_WORKER	Executes the integration flow template ACK_ITEM_MASTER_FLOW

Work units

Work units are logical groupings of data for integration. In Connector Toolbox, work units are classified into these types:

Work units	Description
Entity Units	Entity units support entity-based transactions. This work unit type requires a database connection, token, and condition to execute the worker.
File Units	File Units are work units that cannot be classified as entity or file units. Custom units are the usual entry point for M3-related workers or for import tasks.
Custom Units	Custom Units are work units that cannot be classified as entity or file units. Custom units are the usual entry point for M3-related workers or for import tasks.
Reserved Units	Reserved units are system-defined work units. They cannot be modified or assigned to an integration flow, but they can be scheduled by users.

Setting up work units

In the Connector Toolbox, you can create and set up new workers or manage existing workers by using the Worker Setup module. Workers are managed per integration set. To set up a worker, you must define the work unit details, job schedule, and the task conditions.

Note: Before you modify an existing worker, you must first stop the integration set that it is associated with from running.

SeeStop or start integration set on page 16 .

Manage workers

1 In the navigation pane, select **Worker Setup**.

2 In the Workers section, specify this information:

Worker ID
Specify a worker ID or name.

Workunit Type
Select a work unit type.

Flow

Select an integration flow that the worker must execute. Available options depend on existing integration flows in the application.

See Creating an integration flow on page 48.

Define unit details

Note: This task only applies to entity units and file units.

1 Select a worker.

2 In the Unit Details section, specify this information:

Input
For entity units, select the data source of the input.

For file units, specify the folder path where the XML files are located.

Note: This field only applies to entity units and file units.

Connection String
Specify the database connection to be used.
Note: This field only applies to entity units.

Token Field
Select the token field.
Note: This field only applies to entity units.

Prefix
Specify the file name prefix of the XML files.

Suffix
Specify the file name suffix of the XML files.

3 To add conditions for entity units, specify this information:

Condition
This field is automatically updated.

Name

Select an entity. Available options depend on the database connection string that you specified.

Value
Specify a value.

Define the schedule job

1 In the Schedule section, select a trigger type from the list.

2 Based on the selected trigger type, specify this information:

- Manual
  Note: If you select manual as the trigger type and the selected integration set is currently set to production, a Manual Run button is displayed. To run the worker manually, click the **Manual Run** button.
- Frequently:

  Every
  Select the frequency value from the list.

- Daily:

  Time
  Specify the time in this format: HH:MM:SS.

- OneTime:

  Date Time
  Specify the date and time in this format: DD/MM/YYYY HH:MM.

- Weekly:

  Days of the week
  Select applicable days from the list.

  Time
  Specify the time in this format: HH:MM:SS.

- Monthly:

  Months
  Select applicable months from the list.

  Monthly Type
  Select the monthly type from the list.

Days
Select the applicable days from the list.
Note: This field is available only if you select ByDays as the monthly type.

On
Select the applicable value from the list.
Note: This field is available only if you select ByNthDays as the monthly type.

Days of week
Select the applicable days from the list.
Note: This field is available only if you select ByNthDays as the monthly type.

Time
Specify the time in this format: HH:MM:SS.

3. Click **Save** or **Manual Run**.

Specify the notification class

Notification setup enables the worker to send notifications to Infor Fashion PLM.

1. Go to Notification Setup section.
2. Select a notification class.

Chapter 6: Migrating data

Learn about the data migration feature of Connector Toolbox. It also explains the necessary procedures to migrate data from an SQL or OLE DB-compliant database to the Infor Fashion PLM database.

Migration workspace overview

Migration Workspace enables you to establish database connection to any external tool that uses SQL or OLE DB-compliant database and migrate its contents to the Infor Fashion PLM database. For each migration task in a migration workspace, you can select database objects and identify the fields that can be mapped from the source database to the Infor Fashion PLM database. By using the data migration feature of Connector Toolbox, you can migrate information from and to these sources:

- From one database source to another
- From one schema to another
- From one table to another in the same database

Migration workspace window

The Migration Workspace window is divided into several sections that represent its main configuration functions. This table shows the sections of the Migration Workspace window:

Section	Description
Migration Workspace	This section contains the list of migration workspaces and the action buttons for the selected migration workspace.
Migration Tasks	This section shows the migration tasks for the selected migration workspace and the New Task button.
Properties	In this section, you can modify and defined additional properties of the selected migration task.
Mapping	This section contains the field mapping of the selected migration task.

Section	Description
Destination Dependencies	In this section, you can specify the tables that are related to the selected destination table. If the destination table requires a foreign key from another table, you can specify in this section the lookup foreign key of the applicable field.
Connections	This section shows the available database connections for the selected migration workspace.

Creating a migration workspace

In the Connector Toolbox, you can create and save multiple migration workspaces. This lets you manage and run migration tasks for multiple platforms. In each migration workspace, you can establish database connections to Infor Fashion PLM and any other tool that uses OLE DB-compliant or Microsoft SQL Server.

Note: You can also upload migration contents to the Infor Fashion PLM Connector Toolbox Service folder. See *Infor Fashion PLM Connector Toolbox Installation Guide*.

1. In the navigation pane, select **Migration > Migration Workspace**.
2. On the Migration Workspace field, specify a migration workspace name and click **Add**.

   Note: To manage an existing migration workspace, select a migration workspace from the list.

Establishing database connection

To start using the data migration feature of the Connector Toolbox, you must configure the application to establish a connection to the Infor Fashion PLM database and to any other OLEDB or Microsoft SQL Server.

You can also import data by using .csv, .xls, or .xlsx files. If you want to import a .xlsx file, ensure that you have these applications installed:

- Microsoft Access Database Engine 2010 Redistributable
- 2007 Office System Driver: Data Connectivity Components

Note: By default, an Infor Fashion PLM database connection is available on the Connections section. To update the connection information, double click **InforFashionPLM**.

1. In the navigation pane, select **Migration > Migration Workspace**.
2. Select a migration workspace from the list or create a new migration workspace by clicking **Add**.
3. Specify this information:

   Provider
   Select a provider from the list.
   Note: For .csv, .xls, and .xlsx files, select OLEDB.

Name
Specify a name for the connection.

4 On the Accounts section, specify this information and press **Enter** on your keyboard to add the account information:

Access Type
Select an access type from the list.

Account Id
Specify the account ID.

Password
Specify the password.

Note: If you are importing information from .csv, .xls, and .xlsx files, select **Full** on the Access Type and leave Account ID and Password blank.

5 On the Connection Properties section, set these properties:

Note: Connection properties may vary depending on the provider that you are using. Your connection properties may have more information depending on your server configuration.

a If you selected SQL Server, select **Click** here to add new row and specify this information:

Property
For each row, specify this information:

- Initial Catalog
- Data Source
- Integration Security
- User ID
- Password

Value
For each row, specify the corresponding connection information for these properties:

- Initial Catalog

  Specify the database name.
- Data Source

  Specify the instance/server name of the database.
- Integration Security

  Specify true or false. If value is set to true, Windows Authentication will be used instead of SQL Server Authentication.
- User ID

  Specify the SQL Server user ID.
- Password

  Specify the SQL Server password.

b If you selected OLEDB, click **File Connection Properties** and specify the file information. Details on the Connection Properties section are automatically updated.

6 Click **Save**.

Note: To add another connection, click **New Connection** from the Migration Workspace window.

To switch to a different Infor Fashion PLM destination version, click the **Utility** button, specify the schema, and then click **Switch**.

Creating a migration task

By creating migration tasks, you can select database objects and identify the fields that can be mapped from the source database or file to the Infor Fashion PLM database.

1 In the navigation pane, select **Migration > Migration Workspace**.

2 Select a migration workspace from the list.

3 Click **New Task**.

4 Specify a task name.

5 On the Source tab and the Destination tab, specify this information:

Connection
Select a connection from the list.

The list contains the database connections defined for the selected migration workspace.

See Establishing database connection on page 58.

Object

Select a table from the list. For .csv, .xls, and .xlsx files, the available sheet names are displayed on the list.

6 Select the fields that will be mapped from the source to destination.

7 Click **Save**.

The migration task is added on the Migration Tasks section.

8 Optionally, modify the migration task properties on the Properties section.

Modify the migration task properties

Note: This task is optional.

1 On the Properties section, under Mapping Properties, modify the name of the migration task.

2 Under Source Properties, specify an `SQL WHERE` clause.

3 Under Destination Properties, specify this information:

WhereClause
Specify an `SQL WHERE` clause.

DuplicateCheckFields
Specify the field names that will be checked for duplicate values.

The field must be from the destination table. When a migration task is ran, records with matching values with the defined fields in `DuplicateCheckFields` will not be inserted to the destination table. You can specify multiple fields, separated by comma.

SkipIfNullFields
Specify the field names that will be checked for null or empty values.
Note: The fields must be from the destination table. When the migration task is ran, records with null or empty values on the fields that are specified in `SkipIfNullFields` will be skipped during the process of inserting information to the destination table. You can specify multiple fields, separated by comma.

Set the destination dependencies

Perform this task if the destination table requires a foreign key from another table. Destination dependencies only display tables that are available on the current workspace.

1 Select **Destination Dependencies**.
2 Specify this information:

Migration Object
Select a migration object from the list.

The list contains the all tables from the Infor Fashion PLM database.

Parent Field
Select a parent field from the list.

The list contains the fields of the selected migration object.

Object Field
Select an object field from the list.

The list contains the fields of the destination object for the selected migration task.

Note: All migration tasks with objects that are dependency fields for another migration task are ran first.

Map the fields for migration

1 On the Mapping section, specify this information:

Source
Select a source field from the list.

The list contains the fields selected on the Source tab of the migration task.

Destination
Select a destination field from the list.

The list contains the fields selected on the Destination tab of the migration task.

QuerySql
Specify a query statement.

The query statement is used to retrieve a field from the database that is passed on to the destination field. You can use query statements when retrieving a foreign key value that is not available on the current workspace but is available on the database.

See Set the destination dependencies on page 61.

You can use this format for query statements:

```
SELECT fieldname FROM tablename WHERE condition field = source parameter
```

Note: Source parameter is a field from the source table or object with the prefix @, such as @color_code.

For example, For example `SELECT COLORRNG FROM FSH1. COLORRNG WHERE COLORRNGCODE = @color_code`.

2 Click **Save**.

Run the migration tasks

1 Select the task.

2 Click Run.

A message indicated if the migration is successful or if it contains an error.

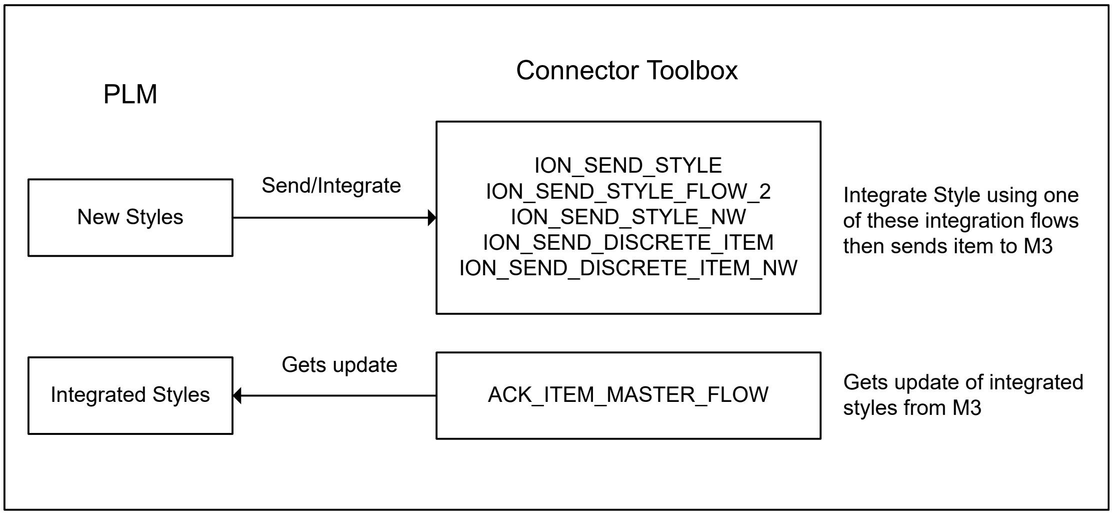
Note: The migration task bar is displayed in yellow to indicate that the migration is in progress. A green migration task bar indicates that the migration is completed.

Chapter 7: Field mappings for M3 Business Engine

Connector Toolbox process flows

Infor Fashion PLM contains default integration flows that are used to integrate Infor Fashion PLM with M3.

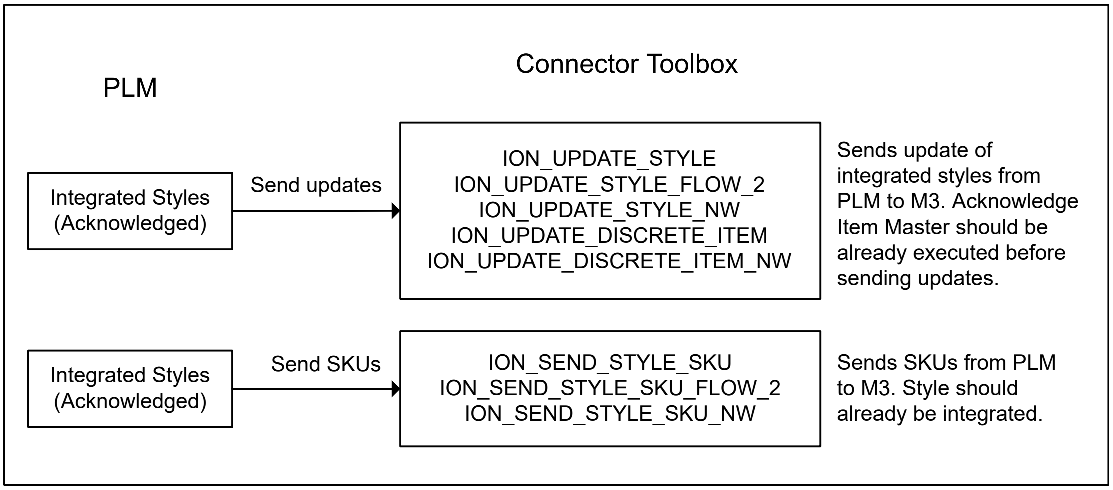
This diagram shows how new styles are sent to M3:



Integrate Style using one of these integration flows then sends item to M3

Gets update of integrated styles from M3

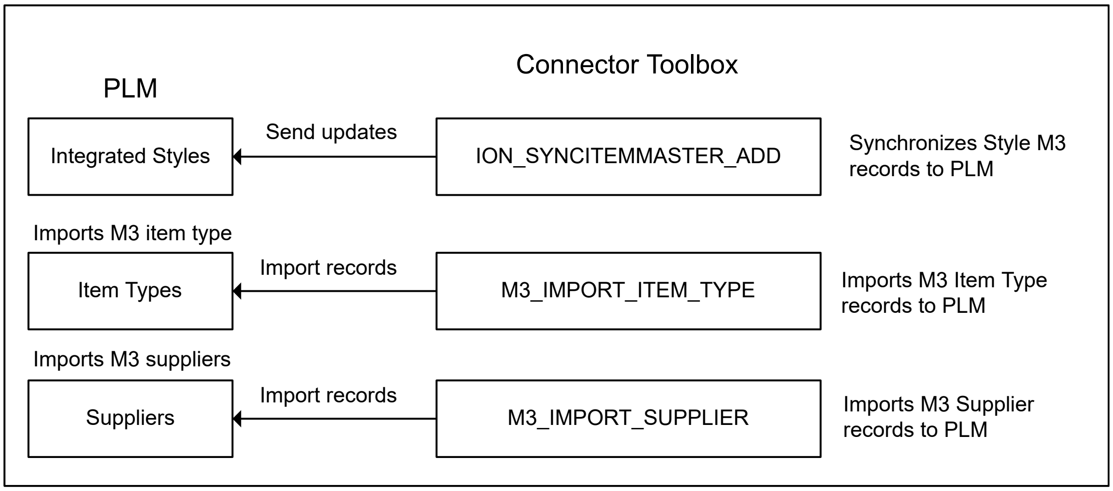
This diagram shows how updated styles are sent to M3:



Sends update of integrated styles from PLM to M3. Acknowledge Item Master should be already executed before sending updates.

Sends SKUs from PLM to M3. Style should already be integrated.

This diagram shows how Infor Fashion PLM gets updates from M3:



Supported fields exported to M3 BE

Note: Dynamic fields can be mapped to other fields in M3 Business Engine.

M3 field	Field name	M3 function	Mapping to IFPLM style field	Dynamic or static?
Item Name	ITDS	MMS001/E	Style Name	Static
Item Description	FUDS	MMS001/E	Description	Static
Item Type	ITTY	MMS001/E	Item Type	Static
Item Group	ITGR	MMS001/E	Sub Category	Dynamic
Product Group	ITCL	MMS001/E	Category	Dynamic
Business Area	BUAR	MMS001/E	Brand	Dynamic
UOM	UNMS	MMS001/E	UOM	Dynamic
Free Field 1	CFI1	MMS001/G	Season	Dynamic
Free Field 3	CFI3	MMS001/G	Collection	Dynamic
Free Field 4	CFI4	MMS001/G	Gender	Dynamic
Free Field 5	CFI5	MMS001/G	Division	Dynamic
Supplier	SUNO	MMS001/H	Preferred/Main Supplier	Static
Sales Price	MMSAPR	MMS001/H	Retail Price	Dynamic
Purchase Price	PUPR	MMS001/H	Purchase Price	Dynamic

M3 field	Field name	M3 function	Mapping to IFPLM style field	Dynamic or static?
Specifications	SPE1	MMS001/I	Style Number	Dynamic
Season control	SECH	MMS001/L	Season (If the season is entered, it activates season process.)	Static
Planner	RESP	MMS002/E	Planner	Static
Supplier	SUNO	MMS002/E	Preferred/Main Supplier	Static
Buyer	BUYE	MMS002/G	Buyer	Static
Country Of Origin	ORCO	MMS003/E	Country of Origin	Static
Custom Stat No	CSNO	MMS003/E	HTS Code	Static
Free Field 01	FM01	MMS016/E	Theme	Dynamic
Free Field 02	FM02	MMS016/E	Numeric Value 1	Dynamic
Free Field 03	FM03	MMS016/E		Dynamic
Free Field 04	FM04	MMS016/E	User Defined Field 1	Dynamic
Free Field 05	FM05	MMS016/E	User Defined Field 2	Dynamic
Free Field 06	FM06	MMS016/E	User Defined Field 3	Dynamic
Free Field 07	FM07	MMS016/E	User Defined Field 4	Dynamic
Free Field 08	FM08	MMS016/E		Dynamic
Free Field 09	FM09	MMS016/E	Carryover Item	Dynamic
Free Field 10	FM10	MMS016/E	Variant Item	Dynamic
Free Field 11	FM11	MMS016/E	Free Field 1	Dynamic
Free Field 12	FM12	MMS016/E	Free Field 2	Dynamic
Free Field 13	FM13	MMS016/E	Free Field 3	Dynamic
Free Field 14	FM14	MMS016/E	Free Field 4	Dynamic
Free Field 17	FM17	MMS016/E	Market Field 3	Dynamic
Free Field 18	FM18	MMS016/E	Market Field 4	Dynamic
Free Field 19	FM19	MMS016/E	Market Field 5	Dynamic
Free Field 20	FM20	MMS016/E		Dynamic

Imported fields from M3 BE to Infor Fashion PLM

Note: Fields that are imported from M3 Business Engine are reflected in the Integration tab of a style. See “Integrating styles with ERPs” in *Infor Fashion PLM Online Help*.

M3 field	Field name	M3 function	Mapping to IFPLM style field
Item Number	ITNO	MMS001/E	Style Number
Item Name	ITDS	MMS001/E	Style Name
Item Description	FUDS	MMS001/E	Description
Make/Buy Code	MABU	MMS001/E	Make/Buy code
Gross Weight	GRWE	MMS001/F	Gross Weight
Net weight	NEWE	MMS001/F	Net Weight
Volume	VOL3	MMS001/F	Volume
Free capacity units	FCU1	MMS001/F	Free capacity units

Chapter 8: Using # keywords in Task Templates

Integration task templates # keywords

You can use # keywords in managing integration task templates to support the handling of string data or records that is used to filter the integration process.

This table shows the # keywords that are available in managing integration task templates:

# Keyword	Description
#GetCurrentDate	Used to get the system current date
#GetUtcNow	Used to get the system current date in UTC format that is supported by Infor Fashion PLM
#substr[<start_index>,<end_index>,<prop>]	Substring command that has the same function in most programming languages This command uses a either a part or the entire string data. The starting point is defined by <start_index> and the <end_index> determines the last index that retrieves the data. The data source field is defined by the <prop> string.
#SplitStr[<delim>,<index>,<prop>	Split string command that is available in most programming languages This command splits the string, data, or word depending on the provided delimiter. This command only take a part of the text that you want to retrieve. • <delim> is the character that specifies where the split will apply. • <index> is the index location of the text that you want to use after the split • <prop> is the field for the split.
#NewGuid	Used to generate a new GUID string
#TickCount	Used to create a unique string by using date and time stamp
#String[<static_text>]	Used to place a static text in a field
#Config[<key>]	Used to get the value from the application configuration file

Note: Some of these # keywords are not supported in editing integration task templates in Connector Toolbox. You can use integration.ict.xml to edit the unsupported # keywords.