Sterling Standards Library

IBM

Using SWIFTNet

Version 8.04.1 for Sterling B2B Integrator 5.24.2

Sterling Standards Library

Using SWIFTNet

Version 8.04.1 for Sterling B2B Integrator 5.24.2

Contents

Chapter 1. SWIFT and SWIFTNet with Sterling B2B Integrator

Sterling B2B Integrator supports the use of Society for Worldwide Interbank Financial Telecommunications (SWIFT), a standard for the financial industry from SWIFT™ that uses the SWIFTNet transport protocol to enable real-time store-and-forward financial messaging through the InterAct and FileAct file protocols.

The SWIFT standards data dictionary is optionally installed from the Deployment > Standards page. It supports all SWIFT Standards Release messages loaded in the standards database. The SWIFT standards data dictionary also contains the special exception and code word validations, and the codes words and qualifiers necessary for the validation of the ISO 15022 messages (500 series). This information is used to automatically generate the translator_swift.properties.in file, which is used by the translator to perform the validations.

Note: You can create a map for all SWIFT Standards Release messages through the Sterling B2B Integrator Map Editor. All messages are validated by Sterling B2B Integrator for syntax (that is, field types, field lengths, and so forth). All SWIFT message are validated for syntax *and* semantics.

The following messages are supported with both inbound and outbound syntax validation and semantic validation (that is, validating the message rules) of the messages between Sterling B2B Integrator and SWIFT:

- All SWIFT 2005 message types
- All SWIFT 2006 message types
- All SWIFT 2007 message types
- All SWIFT 2008 message types
- All SWIFT 2009 message types
- All SWIFT 2010 message types
- All SWIFT 2011 message types
- All SWIFT 2012 message types
- All SWIFT 2013 message types
- All SWIFT system messages (Category 0 messages)
- SWIFT MX Funds (versions 3.0, 4.0, 4.1, 4.4, and 4.5) and Funds Rulebook (version 4.0, 4.1, 4.4, and 4.5)
- SWIFT MX Alternative Funds (version 4.1) and Alternative Funds Rulebook (version 4.1)
- SWIFT MX Fund Processing Passport (versions 4.2 and 4.3) and Funds Rulebook (versions 4.2 and 4.3)
- SWIFT MX Cash Reporting (versions 1.0, 3.0, 3.1, 3.2, 4.0, and 5.0) and the Cash Reporting Rulebook (versions 3.1, 4.0, and 5.0)
- SWIFT MX Cash Management version 4.0 (Standards Release November 2008), 4.0 (Standards Release July 2009), and SWIFT MX Cash Management (Standards Release November 2013)
- SWIFT MX Transaction Reporting (version 1.0)
- FpML (version 1.0)
- SWIFT MX Exceptions & Investigations (versions 1.0, 1.1, and 1.2)

- SWIFT MX Trade Services (versions 1.0 and 2.0)
- SWIFT MX Bulk Payments (versions 2.0 and 2.1)
- SWIFT SCORE (versions 2.0, 2.5, and 2.6)
- SWIFT Proxy Voting (versions 1.0, 1.1, and 1.2)
- SWIFT MX eMandate (version 1.0)
- SWIFT MX Corporate Actions (versions 1.0, 1.1, 1.2, 1.3)
- SWIFT MX Collateral Management (version 1.0)
- SWIFT MX Clearing (versions 1.0 and 2.0)
- SWIFT MX SCORE (versions 2.5 and 2.6)
- ISO 20022 messages
- MQHA Delivery Notification Reconciliation
- ISO 20022 Basic Application Header
- XML Format 2 Header (version 2.0.3)

Additionally, Sterling B2B Integrator allows you to create maps to support the translation of market practices, and provides the following two Market Practices:
- Germany: MT515: Trade Confirmation (Broker to Asset Manager)
- United States: MT536: ISITC-IOA: Statement of Transactions

For Market Practices, the SWIFT standard validations are performed unless the Market Practice has customized a particular validation (for example, restricting a code word validation list). Market Practice code word and qualifier validations are maintained in property files separate from the SWIFT standard property files. By keeping the Market Practices you create separate, we can deliver updates to the SWIFT standard without overwriting your custom-implemented Market Practices. Similarly, semantic validation rules for Market Practices are stored in an extended rule library separate from the library that implements the SWIFT standard semantic validation rules. Please note that the standard property files and semantic validation rules are used when the implemented Market Practice does not override them.

Sterling B2B Integrator also supports and maintains SWIFT System Messages (MT category 0; for example, MT096 FIN Copy to Server Destination Message) which relate to the sending or receiving of messages used to customize your FIN operating environment, including SWIFT user-to-user messages (MT categories 1-9), User-to-SWIFT messages (for example, Delivery Notifications, Retrievals, and so forth), and SWIFT-to-user messages (for example, Retrieved Messages and Non-Delivery Warnings).

In addition, support for SWIFT requires you to create inbound and outbound SWIFT envelopes. You must also create a business process or processes to order the flow of Sterling B2B Integrator activities so you can accomplish your business objectives with SWIFT.

For more information on the SWIFT standard, see

http://www.swift.com

Using SWIFT with Sterling B2B Integrator

This table describes how Sterling B2B Integrator supports SWIFT:

Functionality	Enhancements	Action You Need to Take to Implement SWIFTNet
IBM® Sterling Standards Database for SWIFT	• Provides all validation for field tags and structures for SWIFT messages. • References a SWIFT extended rule library that is automatically checked in so you can use it with your maps. • SWIFT_FINMessageTypes code list, supplied as part of the install, which contains a list of valid SWIFT message types. • If you purchase a Financial Services license, you have access to XML standards once you download them.	• Populate the external codes lists: SWIFT_Addresses, SWIFT_BaseAddresses (only needs to be populated when address verification is enabled), SWIFT_Currencies, SWIFT_Countries (this SWIFT code list contains countries and regions), IBAN_Formats, SWIFT_BICPlusIBAN, and SEPARouting. • If you want to use BICPlusIBAN validation, you must also populate the BICPlusIBAN code list. See Chapter 2, "SWIFT Envelopes," on page 13. • Install the Sterling Standards Database for SWIFT. **Note:** The data dictionary is optionally installed from the Deployment > Standards page. The SWIFT rule libraries for 2008 and forward are installed by installing the Sterling Standards Database for SWIFT.
SWIFT Market Practices	The following two market practices are available in Sterling B2B Integrator: • Germany: MT515: Trade Confirmation (Broker to Asset Manager) • United States: MT536: ISITC-IOA: Statement of Transactions You can create a new message type by customizing a standard SWIFT message type map according to the specifications of the Market Practice. The customized map can be imported into the SWIFT Market Practices data dictionary through the SWIFTRuleImporter utility.	• Install the Sterling Standards Database for SWIFT. The data dictionary is optionally installed from the Deployment > Standards page. The SWIFT rule libraries for 2008 and forward are installed by installing the Sterling Standards Database for SWIFT. • You can create a new message type by implementing the additional restrictions related to the desired Market Practice.

Functionality	Enhancements	Action You Need to Take to Implement SWIFTNet
Services	• Generic Envelope service envelopes SWIFT messages. **Note:** The Generic envelope service assumes that SWIFT expects a starting CRLF (carriage return/line feed) but not an ending CRLF. • Generic Deenvelope service deenvelopes SWIFT messages. **Note:** The Generic Deenvelope service assumes that SWIFT expects a starting CRLF (carriage return/line feed) but not an ending CRLF. • EDI Deenvelope service accepts SWIFT messages. • SWIFTNet Server Adapter (for SWIFT version 6.x only). • SWIFTNet Client Service (for SWIFT version 6.x only). • SWIFTNet Reconciliation Service. • SWIFTNet HTTP Server Adapter ((preconfigured instance for SWIFT version 6.x only) • SWIFTNet7 Adapter (for SWIFTNet 7 only). • SWIFTNet7 Adapter Scheduler (for SWIFTNet 7 only). • SWIFTNet7 Client Service (for SWIFTNet 7 only). • SWIFTNet7 Server Service (for SWIFTNet 7 only). • SWIFTNet 7 RMA Service (for SWIFTNet 7 only). • SWIFTNet7 ASP Service (for SWIFTNet 7 only). • SWIFTNet7 HTTP Server Adapter (preconfigured instance for SWIFTNet 7). • SWIFTNet7 HTTP Client Adapter (preconfigured instance for SWIFTNet 7). • SWIFTNet7 File System Adapter (preconfigured instance for SWIFTNet 7). • An application, SWIFTNet Server, supports FileAct and InterAct processing for SWIFT version 6.x, as well as failover processing. • An application, SWIFTNet7 Server, supports FileAct and InterAct processing for SWIFT version 7, as well as failover processing.	• Install the SWIFTNet Server on a machine running either the Sun Solaris 5.10, Windows Server 2003 (Standard or Enterprise Edition), or AIX® 5.3 operating system. • For SWIFT version 7, install the MEFGCommServer7 and MEFGSWIFTNetServer7 on a machine running either the Sun Solaris 10, Windows 2008, or AIX 6.1operating system. • For SWIFT version 6.x, configure the SWIFTNet Client Service. • Configure the SWIFTNet Server adapter (for SWIFT version 6.x), or the SWIFTNet7 Adapter and SWIFTNet Adapter Scheduler (for SWIFT version 7).
Envelopes	• Inbound SWIFT envelope wizard to implement inbound SWIFT messages. • Outbound SWIFT envelope wizard to implement outbound SWIFT messages. • The SWIFT_FINMessageTypes code list is automatically installed with Sterling B2B Integrator. This code list contains a list of valid message types. • Deenvelope functionality supports SWIFT system messages and user-to-user messages.	• Create the appropriate SWIFT envelopes for each message type you are sending and receiving. • You need to populate four code lists to use in conjunction with the SWIFT_FINMessageTypes code list to perform SWIFT validations. • Configure the EDI Encoder service for use with the outbound SWIFT messages.

Functionality	Enhancements	Action You Need to Take to Implement SWIFTNet
Predefined business processes	• SWIFTDeenvelope business process • SWIFTEnvelope business process • SWIFTNetClient business process • SWIFTNetClientFA business process • handleSWIFTNetServerSnFRequest • handleSWIFTNetInboundCorrelation • handleSWIFTNetOutboundCorrelation • handleSWIFTNetServerFADelNotif • handleSWIFTNetServerFAEvent • handleSWIFTNetServerFARequest • handleSWIFTNetServerFASnFDelNotif • handleSWIFTNetServerFASnFRequest • handleSWIFTNetServerRequest • handleSWIFTNetServerSnFDelNotif • handleSWIFTNetServerFASnFEvent • handleSWIFTNetSnFInboundCorrelation • handleSWIFTNetSnFOutboundCorrelation • SWIFTMessageEntryOutbound • handleSWIFTNetOpenInputChannel • SWIFTNetOpenInputChannel • SWIFTMessageEntryOutbound • SWIFTNet3rdPartyClientForceRefusal • SWIFTNet3rdPartyClientNotification • SWIFTNetClientRenewSecContext • SWIFTNetClientResend • SWIFTNetCloseInputChannel • SWIFTNetCreateInputChannel • SWIFTNetDeleteInputChannel • SWIFTNet7WaitRequest • SWIFTNet7CloseInputChannel • SWIFTNet7CloseOutputChannel • SWIFTNet7CreateInputChannel • SWIFTNet7CreateOutputChannel • SWIFTNet7DeleteInputChannel • SWIFTNet7DeleteOutputChannel • SWIFTNet7FileActFetch • SWIFTNet7FileActNotification • SWIFTNet7ImportASP • SWIFTNet7ImportRMA • SWIFTNet7OpenInputChannel • SWIFTNet7OpenOutputChannel • SWIFTNet7QueryReplication • SWIFTNet7RenewSecContext	The business processes that are related to the SWIFTNet workflow must have the Document Tracking option enabled when you check in or edit the business process. Additionally, you need to configure other parameters in the SWIFTNetClient or SWIFTNet7Client business process to support SWIFTNet.

Functionality	Enhancements	Action You Need to Take to Implement SWIFTNet
Predefined business processes, continued	• SWIFTNet7ThirdPartyForceRefusal • handleSWIFTNet7ContextReport • handleSWIFTNet7Events • handleSWIFTNet7FileActDelNotif • handleSWIFTNet7FileActEvent • handleSWIFTNet7FileActRequest • handleSWIFTNet7FileActSnFRequest • handleSWIFTNet7InterActRequest • handleSWIFTNet7InterActSnFRequest • handleSWIFTNet7OpenChannels • handleSWIFTNet7ResponseInfo • handleSWIFTNet7SnFDelNotif • Schedule_SWIFTNet7AdapterScheduler • SWIFTNet7AbortTransfer • SWIFTNet7Client • SWIFTNet7ClientResend	
Sterling B2B Integrator Map Editor	• Sterling B2B Integrator Map Editor wizard enables you to generate a file layout using the MT or MX message that you select. Included in the SWIFT map are groups, records, composites, and fields that are defined by SWIFT. • Properties dialog boxes that enable you to define and modify SWIFT map components. • Autolink function automatically creates links between input and output fields that have the same name or business name. This function can be used with any data format. • Extended Rules Library function (used with SWIFT and any other data format) contains a list of rules in a separate file outside of the Sterling B2B Integrator Map Editor source. Sterling B2B Integrator Map Editor stores the name of the library in its source file, so when you load a map the library is also loaded and compiled. This enables you to create a library of extended rules and then add it to any other map, so you do not have to recreate those extended rules after the first time. • Extended rules are used to validate SWIFT messages. • SWIFT MX validations are performed and generated maps define the parameters used to look up validations that are defined in a properties file.	• Download Sterling B2B Integrator Map Editor. • Create a map or maps to translate your SWIFT messages. • Place custom extended rule code inside a validation block.

Functionality	Enhancements	Action You Need to Take to Implement SWIFTNet
Extended Rule Libraries	The extended rule libraries that are loaded automatically when you download SWIFT and SWIFTNet Financial Services XML standards are: • SWIFT_2002.erl • SWIFT_2005.erl • SWIFT_2006.erl • SWIFT_2007.erl • SWIFT_2008.erl • SWIFT_2009.erl • SWIFT_2010.erl • SWIFT_2011.erl • SWIFT_2012.erl • SWIFT_2013.erl • SWIFTMX_ v1-0.erl • SWIFTMX_BulkPayments_v2-0.erl • SWIFTMX_BulkPayments_v2-1.erl • SWIFTMX_CashManagement_v4-0.erl • SWIFTMX_CashManagement_v2009.erl • SWIFTMX_CashManagement_v2013.erl • SWIFTMX_CashReporting_v1-0.erl • SWIFTMX_CashReporting_v3-0.erl • SWIFTMX_CashReporting_v3-1.erl • SWIFTMX_CashReporting_v3-2.erl • SWIFTMX_CashReporting_v4-0.erl • SWIFTMX_CashReporting_v5-0.erl • SWIFTMX_EI_v1-0.erl • SWIFTMX_EI_v1-1.erl • SWIFTMX_EI_v1-2.erl • SWIFTMX_e-Mandate_v1-0.erl • SWIFTMX_FpML_v1-0.erl • SWIFTMX_Funds_v2-0.erl • SWIFTMX_Funds_v2-1.erl • SWIFTMX_Funds_v2-2.erl • SWIFTMX_Funds_v3-0.erl • SWIFTMX_Funds_v3-1.erl • SWIFTMX_Funds_v4-0.erl • SWIFTMX_Funds_v4-1.erl • SWIFTMX_Funds_v4-2.erl • SWIFTMX_Funds_v4-4.erl • SWIFTMX_Funds_v4-5.erl • SWIFTMX_Clearing_v1-0.erl • SWIFTMX_Clearing_v2-0.erl • SWIFTMX_CorporateActions_v1-0.erl • SWIFTMX_CorporateActions_v1-1.erl • SWIFTMX_CorporateActions_v1-2.erl • SWIFTMX_CorporateActions_v1-3.erl	None

Functionality	Enhancements	Action You Need to Take to Implement SWIFTNet
Extended Rule Libraries, continued	• SWIFTMX_CollateralManagement_v1-0.erl • SWIFTMX_CollateralManagement_v1-2.erl • SWIFTMX_ProxyVoting_v1-0.erl • SWIFTMX_ProxyVoting_v1-1.erl • SWIFTMX_ProxyVoting_v1-2.erl • SWIFTMX_SCORE_v2-0.erl • SWIFTMX_SCORE_v2-5.erl • SWIFTMX_FundsProcessingPassport_v4-2.erl • SWIFTMX_FundsProcessingPassport_v4-3.erl • SWIFTMX_AlternativeFunds_v4-1.erl • SWIFTMX_TradeServices_v1-0.erl • SWIFTMX_TradeServices_v2-0.erl • SWIFTMX_TransactionReporting_v1-0.erl	
SWIFTNet Correlations	• Enables you to search for SWIFTNet messages using specific criteria.	• Run inbound and outbound data and then use the search interface to search for messages. • If you are upgrading from a previous version to the IBM Sterling Standards Library Version 5.3, you must run the **SwnetCorrMigrate.cmd** script to migrate correlation detail records to new database tables.
Document Tracking	• Support for monitoring SWIFTNet data flows. • Support for tracking SWIFTNet messages within Sterling B2B Integrator. The tracking and correlation functionality automatically tracks the SWIFTNet messages the same manner as the other standards are tracked. • Support for document repair and resend.	• Use the data flow interface to track inbound and outbound SWIFTNet processes. • All the business processes that are related to the SWIFTNet workflow (SWIFTNetClient, handleSWIFTNetServerRequest, and handleSWIFTNetServerSnfRequest) must have the Document Tracking option enabled when you check in or edit the business processes. • Access messages with Ready To Edit status, repair, and resend them.
SWIFTNet Routing Rule	• Interface to support creation, modification, and deletion of SWIFTNet routing rules for SWIFT version 6.x and 7. • The criteria specified in the SWIFTNet Routing Rule enables a user to set a specific business process to process the incoming request from SWIFTNet (for version 6.x). For SWIFTNet version 7, it allows you to select the action associated with the routing rule (including specifying a business process, store it into the mailbox, or store it into the file system). • You can also use wildcards in the SWIFTNet routing rule.	• Create a business process or modify the SWIFTNetClient or SWIFTNet7Client business process. • Create a SWIFTNet routing rule that is linked to the business process you created.

Functionality	Enhancements	Action You Need to Take to Implement SWIFTNet
SWIFTNet Service Profile	For SWIFTNet version 6.x, enables you to associate SWIFTNet Request Type with a Schema for Header Validation. **Note:** With the introduction of the Application Service Profile (ASP) in SWIFTNet 7.0, this feature is no longer required.	• For SWIFTNet version 6.x, create a new SWIFTNet Service Profile (not needed for SWIFTNet version 7.0).
SWIFTNet Copy Service Profile	Enables you, as a third party, to specify the relationship between a Service Name and Copy Mode (T or Y Copy). **Note:** If the useASP flag is set to TRUE, this feature is not used.	• For SWIFTNet version 6.x, create a SWIFTNet Copy Service Profile if you are using T or Y Copy.
SWIFTNet Client Profile	SWIFTNet Client Profile Enables you, as a requestor, to send a request using an ASP-driven wizard. This minimizes the occurrence of unnecessary errors when sending a request.	Create a SWIFTNet Client Profile if you are using SWIFT version 7.
SWIFT Message Editor	• Interface to enable a user to correct a SWIFT MT or MX message that was returned due to an error either in transmission or translation. • Support for validating messages.	• Assign permissions for two different roles: the Editor and the Reviewer and give appropriate permission sets to each role. • Correct appropriate messages and notify the appropriate user that activity is required. • Review change history for messages.
SWIFT Message Entry Workstation	Interface to enable manual message entry using a structured message editor.	• Assign permissions for the three different roles (Creator, Verifier, Tracker). • Create, verify, view (read-only), and track new SWIFT messages created through this interface.
Input Channel	The SWIFTNet 6.1 release introduces the concept of an input channel. Currently, Sterling B2B Integrator only supports the use of input channel for InterAct messages in store-and-forward (SnF) mode. An input channel can be used by the messaging interface to establish an input session with SWIFT. The input session starts when the messaging interface opens the input channel and ends when the messaging interface closes the input channel. The input channel also supports sender-to-receiver first-in-first-out (FIFO), which means that each message is delivered only one time and minimizes the number of possible duplicates.	You must configure the following three application components to use the input channel: • SWIFTNet Server Adapter or SWIFTNet7 Adapter • Resend Scheduler (SWIFT 6.x only) • SWIFTNet Client Service or SWIFTNet7 Client Service (including configuring a predefined business process to create a new input channel)

Functionality	Enhancements	Action You Need to Take to Implement SWIFTNet
Output Channel	The SWIFTNet 7.0 release introduces output channels and queue sharing. In previous versions, a queue could not be shared by different messaging interfaces at the same time because SWIFT only allowed one queue session to be opened at one time. By introducing the output channel in version 7.0, SWIFT allows queue sharing by enabling a queue to use different output channels and the interface to open several output channels on the same queue at the same time.	You must configure the following application components to use the input channel: • SWIFTNet7 Adapter 　SWIFTNet7 Adapter Scheduler 　SWIFTNet7 Client Service (including configuring a predefined business process to create a new output channel)
SWIFTNet Monitor	Displays the status of the SWIFTNet queues and channels, including the input channel.	View the SWIFTNet Monitor.

Prerequisite for Using SWIFTNet

The audience using this software should be familiar with Sterling B2B Integrator, the SWIFTNet standard, and using InterAct and/or FileAct.

Note: This documentation is not intended to explain the SWIFTNet standard.

Use SWIFTNet

About this task

To use SWIFTNet with the Sterling B2B Integrator, you must complete the following tasks:

Procedure

1. Download the SWIFT standards data dictionary. See "Download and Install the SWIFT Standards Data Dictionary" on page 11.
2. Create inbound and outbound SWIFT envelopes. See Chapter 2, "SWIFT Envelopes," on page 13.
3. Create the necessary code lists and maintain them as needed. See "Maintain the External Code Lists" on page 18.
4. Enable the document tracking option for all the business processes that are related to the SWIFTNet workflow. See Chapter 10, "Document Tracking for SWIFTNet Transport," on page 203 and Chapter 3, "SWIFT and SWIFTNet Business Processes," on page 67.
5. Configure the EDI Encoder service for use with outbound SWIFTNet messages. See "Configuring the EDI Encoder Service for SWIFT Outbound Messages" on page 62.
6. Create the appropriate SWIFT maps. See Chapter 4, "Creating SWIFT Maps," on page 81.
7. Configure the SWIFTNet routing rule. See "Create a SWIFTNet Routing Rule" on page 230.
8. If you are using SWIFTNet 6.x, create an instance of the SWIFTNet Server Adapter and the SWIFTNet Client service.
9. If you are using SWIFTNet 7.0, create an instance of the SWIFTNet7 Adapter, the SWIFTNet7 Server Service, and the SWIFTNet7 Client Service.

10. For SWIFTNet version 6.x, create a new SWIFTNet Service Profile (not needed for SWIFTNet version 7.0).
11. Create a SWIFTNet Copy Service Profile.
12. Create a SWIFTNet Client Profile (SWIFTNet7 only).
13. Use SWIFT Alliance Access (SAA) for SWIFT support.
14. Use the WebSphere® MQ adapter to communicate with SAA.

Use InterAct

About this task

To use InterAct with the Sterling B2B Integrator, you must complete the following tasks:

Procedure

1. Configure the appropriate service parameters, including failover processing configuration.
2. Configure the SWIFTNet Server for InterAct processing. See Chapter 7, "SWIFTNet Server," on page 113.

Use FileAct

About this task

To use FileAct with the Sterling B2B Integrator, you must complete the following tasks:

Procedure

1. Configure the appropriate service parameters.
2. Configure the SWIFTNet Server for FileAct processing. See Chapter 7, "SWIFTNet Server," on page 113.

Download and Install the SWIFT Standards Data Dictionary

About this task

Before you install the SWIFT standards data dictionary (standards database) on your desktop, consider these guidelines:

- Download the Sterling B2B Integrator Map Editor.
- For the most current version of the SWIFT standards, contact IBM Customer Support.
- Be sure your desktop meets the Windows Client requirements listed in the Sterling B2B Integrator *System Requirements*.

To download and install the standards database:

Procedure

1. From the Sterling B2B Integrator **Deployment** menu, select **Standards**.

   Note: You must download and install both the Sterling B2B Integrator Map Editor and the desired Sterling Standards Database for SWIFT from the same Sterling B2B Integrator release.

2. In the Download and Install section next to **Download SWIFT Standards**, click **Go!**
3. Click **Save** to save the executable file and then double-click the file and click **Run** to start the install.
4. You are prompted to choose the setup language (that is, the language in which the installation instructions will be given). Select the setup language and click **Next**.

   Note: The Standards installation program is available in the following languages: English, Français (France), Deutsche (Germany), Italiano (Italy), Japanese (Japan), Korean (Korea), Português (Brazil), Español (Spain), Simplified Chinese (China), and Traditional Chinese (Taiwan).
5. In the Welcome window, click **Next**.
6. In the Choose Destination Location window, select where you want to install the standards database:
   - If you accept the default location, click **Next**.
   - If you want to specify a different location, click **Browse**, specify the path to the folder, click **OK**, and click **Next**.

     If you specify a folder name that does not exist, the installation program displays a message asking if you want to create that folder.
7. In the Select Components window, verify that **SWIFTStandardDatabase**, **SWIFTMarketPractice**, and **ExtendedRuleLibrary** are selected and click **Next**.

   The download wizard installs the Sterling Standards Database for SWIFT database.
8. In the Setup Complete window, click **Finish**.

Chapter 2. SWIFT Envelopes

A *document envelope* consists of control information that enables organizations to effectively exchange messages. This information is added in headers and trailers to messages. Document envelopes are specific to the message protocol used. Creating document envelopes is necessary to use SWIFTNet with your trading partners.

SWIFT enveloping supports transformation and message validation using WebSphere Transformation Extender. With WebSphere Transformation Extender installed and integrated with Sterling B2B Integrator, the option **Use WebSphere Transformation Extender Compliance Checking** becomes available for the use of WebSphere Transformation Extender compliance checking. The valid values for this option are Yes and No (default).

SWIFTNet has only one level of envelope, which you must modify appropriately to reflect your information and your trading partner's information. Envelopes specify whether the message is inbound or outbound:

* The **Inbound SWIFT envelope** identifies messages that are received by IBM Sterling B2B Integrator so they can be properly routed. Inbound envelopes also give you the option to translate messages when you choose to check messages for compliance. By choosing to translate messages from within the envelope, you can reduce message processing time because you do not need to specify a separate Translation service step in the business process. You need to create an Inbound SWIFT envelope to configure deenveloping information for MT or MX messages. See "Inbound SWIFT Envelope" on page 21.
* The **Outbound SWIFT envelope** identifies messages so they can be sent to and received by trading partners. You need to configure an Outbound SWIFT envelope to configure enveloping information for MT or MX messages. See "Outbound SWIFT Envelope" on page 37.

When you envelope an outbound SWIFTNet message, the SWIFTNet header and trailer are created. For an inbound message, the envelope contains the header information (the trailer information is a summary appended to the SWIFTNet data). The header types are MX or MT messages.

MT and MX messages are included within the Body element. MT messages are base-64 encoded and MX messages are XML.

Sterling B2B Integrator also enables you to use SWIFT XML Format 2, the envelope supported by SWIFT Alliance Access 6.

As part of SWIFTNet enveloping, the Sterling B2B Integrator uses code lists to validate the data. The Sterling B2B Integrator uses code pairs in code lists to identify items in transactions between two or more trading partners. A trading partner code list consists of one or many pairs of code values containing a sender code and a receiver code. Each code pair has one description and up to four additional codes relating to the pair. Code lists are dynamic and are stored in a database.

The SWIFT_FINMessageTypes code list is automatically installed with the Sterling B2B Integrator. This code list contains a list of valid SWIFT message types. The three-digit message number is entered for sender code and receiver code, and the

description is set to **SWIFT Message Type**. However, you need to populate these additional code lists to perform SWIFTNet validations:

- **SWIFT_Addresses**—used to check the sender and receiver IDs within the message. This code list is shared with message authentication. The code list used is the same code list that is used for message validations. You will type the address in the Sender Code, Receiver Code, and Description parameters, and use the Text1 parameter to indicate the subtype of the address. The SWIFT_Addresses and SWIFT_BaseAddresses code lists are used to differentiate between bad base addresses and bad branch codes when necessary. The SWIFT_Addresses code list is also used for verification of those addresses that are contained within the body of a message. BIC and BEI addresses are validated against entries in the SWIFT_Addresses code list

- **SWIFT_BaseAddresses**—this is a list of the 8-character address (the BIC minus the branch code) that are valid as part of a sender address when generating a message. You will type the eight-digit code in the Sender Code, Receiver Code, and Description parameters. The SWIFT_Addresses and SWIFT_BaseAddresses code lists are used to differentiate between bad base addresses and bad branch codes when necessary.

  Note: You only need to populate the SWIFT_BaseAddresses code list when you have enabled address verification. See "Enabling and Disabling Address Verification" on page 17.

- **SWIFT_Currencies**—this is a list of the valid currencies that can be used in a SWIFT message. You will use the Text1 parameter to indicate the maximum number of digits after the decimal point that the currency supports. Amount and Currency values are validated against ISO entries in the SWIFT_Currencies code list.

- **SWIFT_Countries (this SWIFT code list contains countries and regions)**—this is a list of the valid countries or regions that can be used as part of the address when generating a SWIFT message. IBAN values and Country or Region codes are validated against entries in the SWIFT_Countries (this SWIFT code list contains countries and regions) code list.

- **SWIFT_IBANFormat**—this is a list of specific IBAN formats that countries or regions may use. This code list is used with BICPlusIBAN validation.

  The details of this code list are as follows:

  Note: A plus sign indicates those fields are concatenated together. Underscore and semi-colon characters act as delimiters and are added to the code list data.

  Parameter
  Field Name (refers to the name defined in the SWIFT data files)

  SenderID
  IBAN Country or Region Code 3

  ReceiverID
  IBAN Country or Region Code 3

  Description
  IBAN Country or Region Code 3

  Text1 IBAN Country or Region Code Position 4; IBAN Country or Region Code Length 5

  Text2 IBAN Check Digits Position 6; IBAN Check Digits Length 7

  Text3 Bank Identifier Position 8; bank Identifier Length 9

Text4 Branch Identifier Position 10; branch Identifier Length 11

Text5 IBAN National ID Length 12;

Text6 Account Number Position 13; account Number Length 14

Text7 IBAN Total Length 15

- **SWIFT_BICPlusIBAN and BICPlusIBAN**—these are lists of the clearing codes used for validating BICPlusIBAN combinations and clearing codes. These lists replace the older BIC+ database (and in the Sterling B2B Integrator, the SWIFT_ClearingCodes code list). Depending on which application you are using, you will use one of these lists. Financial institutions can look up any missing BICs linked to the IBANs for every financial institution in the 31 SEPA countries or regions. They can also validate the IBANs and BICs, including their relationship.

  Note: The BICPlusIBAN directory is a *replacement* for the BIC+ database.

  The details of this code list are as follows:

  Note: A plus sign indicates those fields are concatenated together. Underscore and semi-colon characters act as delimiters and are added to the code list data.

  Parameter
  Field Name (refers to the name defined in the SWIFT data files)

  SenderID
  IBAN Country or Region Code 19 + Unique IBAN National ID, or Clearing Code 21

  ReceiverID
  BIC Code 7 + Branch Code 8

  Description
  Institution Name 4

  Text1 Parent Bank Code 15

  Text2 IBAN BIC Code 11

  Text3 IBAN Branch Code 12

  Text4 Routing BIC Code 13

  Text5 Routing Branch Code 14

  Text6 Subtype Indicator 15

  Text7 Service Codes 26

  Text8 CHIPS UID 24

- **SWIFT_SEPARouting**—this contains the SEPA Routing Directory. With the SEPA Routing Directory, banks sending SEPA payments can verify whether the operational BICs of their correspondent are SEPA-adherent and operationally ready for SEPA, and can verify the channel through which they can be reached for routing payments. Therefore, the SEPA Routing Directory provides the operational information necessary to exchange SEPA payments with the institutions listed in the EPC Register of Participants. As recommended by the EPC, the directory only contains data related to adherent institutions whose reference BIC is published in the EPC Register of Participants. The directory contains information on receiving banks that are SEPA compliant and shows the supported channels for each, across Clearing and Settlement Mechanisms (CSMs), Automated Clearing Houses (ACHs), and intermediary banks. The details of this code list are as follows:

Note: A plus sign indicates those fields are concatenated together. Underscore and semi-colon characters act as delimiters and are added to the code list data.

Parameter
  Description

SenderID
  BIC Code 4 + Branch Code 5 _ Service Level 9 _ Scheme Instrument 10

ReceiverID
  BIC Code 4 + Branch Code 5 _ Service Level 9 _ Scheme Instrument 10

Description
  Institution Name 6

Text1 Branch Code 5

Text2 Service Level 9

Text3 Scheme Instrument 10

Text4 Country or Region Code 8

Text5 Operational Readiness Date 12

Text6 Valid From; Valid To 18

Text7 Adherent Institution Flag 11

Text8 Intermediary Institution BIC 16

Text9 [Payment Channel Id 13 : Reachability Type 15: Preferred Channel Flag 14]0-n

- **NISOLanguage**—Language codes are validated against ISO entries in the NISOLanguage code list.

See "Maintain the External Code Lists" on page 18.

SWIFT codes list validations are applied to both SWIFT MT and MX messages for currencies, Country or Region codes, BIC/BEI addresses, and International Bank Account Number checksum validation (IBAN). The Sterling B2B Integrator allows you to define codes lists for currencies, countries or regions, and BIC or BEI addresses (which are validated against the SWIFT_Addresses code list). IBAN data contains a Country or Region code that is validated against the SWIFT_Countries code list (this SWIFT code list contains countries and regions) in the Sterling B2B Integrator, and additional IBAN validation is handled internally by the translator.

The validation of the SWIFT special functions <CUR>, <SWIFTBIC>, <NON-SWIFTBIC>, <CC>, and <IBAN> use these code lists. You must update and maintain these codes lists, as necessary.

For outbound SWIFTNet messages, you also need to configure the EDI Encoder service to include the proper values for the following parameters:
- AccepterLookupAlias
- ReceiverID
- SenderID
- ReceiverIDQual
- SenderIDQual

See "Configuring the EDI Encoder Service for SWIFT Outbound Messages" on page 62.

Note: When editing a SWIFT envelope, if you skip wizard pages by clicking **Save** or clicking on a later step, the final confirmation page will display unexpected values. Also, if you skip page in the envelope wizard and then use the **Back** button, incorrect pages may be displayed. If you experience this issue, save from the confirmation screen without using the **Back** button and the envelope is saved correctly. However, if you use the **Back** button, an unexpected page may be displayed (that is, a page that would not normally be displayed based on the envelope values), and the page may hang or cause you to enter an improper value. If this occurs, cancel out of the envelope wizard, and start editing the SWIFT envelope again.

Enabling and Disabling Address Verification

Sterling B2B Integrator allows you to enable or disable address verification. Address verification is performed using the SWIFT_Addresses and SWIFT_BaseAddresses code lists. See "Maintain the External Code Lists" on page 18 for more information on creating these code lists.

Note: When using WebSphere Transformation Extender compliance checking, Sterling B2B Integrator compliance checking of BIC sender and receiver addresses is not available. To use Sterling B2B Integrator compliance checking with BIC addresses, clear the Validate BIC Addresses option in WebSphere Transformation Extender compliance checking.

Configure Inbound Address Verification
About this task

The **enveloping.verify_addresses_while_deenveloping.SWIFT_FIN_INBOUND** property enables and disables inbound address verification. The default is FALSE (disabled), but you can configure it to TRUE.

To enable *inbound* address verification, complete these steps:

Procedure
1. Access the *install_dir*/**properties/enveloping.properties.in** file, and change the line **enveloping.verify_addresses_while_deenveloping.SWIFT_FIN_INBOUND** to **TRUE**, as noted below:

   ```
   enveloping.verify_addresses_while_deenveloping.SWIFT_FIN_INBOUND=True
   ```

   Note: If you want to then disable inbound address verification, you can do so by accessing the *install_dir*/**properties/envelope.properties.in** file, and change the line **enveloping.verify_addresses_while_deenveloping.SWIFT_FIN_INBOUND=FALSE**.
2. Save and close the **enveloping.properties.in** file.
3. Stop Sterling B2B Integrator.
4. Run the setupfiles script using one of the following steps:
   - (UNIX or Linux) - From the *install_dir*/bin directory, run the **setupfiles.sh** command.
   - (Windows) - From the *install_dir*\bin directory, run the **setupfiles.cmd** command.
5. Start Sterling B2B Integrator.

Configuring Outbound Address Verification

To enable or disable *outbound* address verification, use the Outbound SWIFT envelope parameter **Validate Sender and Receiver**. This parameter allows you to enable (by choosing **Yes**) or disable (by choosing **No**) address verification. The default is **No** (sender and receiver verification is disabled). See "Outbound SWIFT Envelope" on page 37 for more information.

FIN Copy Messages

IBM Sterling B2B Integrator processes FIN Copy messages and automatically extracts both layers of envelopes (one envelope is the original one as created by the sender entity and the second enveloping level is added by the SWIFT network). IBM Sterling B2B Integrator processes FIN Copy messages as follows:

Maintain the External Code Lists

About this task

All necessary code lists are automatically installed but except for the Message Type code lists, they are not automatically populated with codes. Therefore, you need to populate the following five external code lists for use in conjunction with the SWIFT_MessageTypes code list (all code lists are automatically installed with the Sterling B2B Integrator—the SWIFT_MessageTypes code list already contains a list of valid message types) to perform SWIFTNet validations.

The external code lists for which you need to populate SenderID, ReceiverID, Description, and occasionally Text1 are:
- SWIFT_Addresses
- SWIFT_BaseAddresses
- SWIFT_Currencies
- SWIFT_Countries (this SWIFT code list contains countries and regions)
- SWIFT_BICPlusIBAN
- SWIFT_IBANFormats
- SWIFT_SEPARouting

You populate these code lists automatically by using the HIPAA codelist conversion map for countries or regions to populate it. However, prior to using the HIPAA codelist conversion map, you need to modify the import file created using the conversion map (the import file is used to populate the codelist). To modify the import file you need to change the codelist name from xxx to SWIFT_xxx (for example, change BICPlusIBAN to **SWIFT_BICPlusIBAN)**.

To populate the each code list automatically:

Procedure
1. From a command line, go to the **tp_import** directory.
2. Type the following command to start the conversion and import process, where <map name> is the name of the map to use during translation (without the file extension) and <code list path and filename> is the fully qualified name of the code list to translate, including filename extension, if any:
   - If you are using Windows, **hipaaconvert.cmd [-import] <map name> <code list path and filename>**

- If you are using UNIX, **hipaaconvert.sh [-import] <map name> <code list path and filename>**

  Do not specify the file extension for the map name when importing a code list—just indicate the base name of the map.

  The [-import] parameter is optional. You can convert the code list file without importing it. If you do not use the [-import] parameter during conversion, you can import the resulting XML file into the Sterling B2B Integrator using the import utility.

  The input files and maps for each code list are as follows:

Code List	Map	Input File and Directory
SWIFT_Addresses	SWIFTFIFIleToSWIFT_Addresses	FI file from BIC directory
SWIFT_BaseAddresses	SWIFTFIFIleToSWIFT_BaseAddresses	FI file from BIC directory
SWIFT_BICPlusIBAN	SWIFTBIFileToSWIFT_BICPlusIBAN	BI file from BICPlusIBAN directory
SWIFT_IBANFormats	SWIFTISFileToSWIFT_IBANFormats	IS file from BICPlusIBAN directory
SWIFT_Countries (this SWIFT code list contains countries and regions)	SWIFTCTFileToSWIFT_Countries (which contains countries and regions)	CT file from BIC directory
SWIFT_Currencies	SWIFTCUFileToSWIFT_Currencies	CU file from BIC directory

3. Once the utility completes, a translation report (hipaaconvert.rpt) and an input file (hipaaconvert.xml) are created. If no translation errors are reported, the code list was successfully generated (and imported if you used the [-import] parameter). A code list will not be imported if there are translation errors.

4. To modify the import file, change the codelist name from xxx to **SWIFT_xxx** so it can be used for SWIFT messages.

Disabling Automatic BICPlusIBAN Validation for the SWIFT_BICPlusIBAN Code List

The SWIFT_BICPlusIBAN code list is the only code list that uses BICPlusIBAN validation. If you want to disable the automatic BICPlusIBAN validation, access the **translator_swift_<year>.properties** file and change this property to NO to turn off the validation:

```
syntax.BICPlusIBAN=YES
```

Create Envelopes

About this task

Inbound envelopes define expected header and trailer information for inbound messages. This information helps IBM Sterling B2B Integrator route and process the messages. Outbound envelopes specify information about messages that enables them to be sent to and received by trading partners, and they gather and provide the appropriate data used to create the header.

To create an envelope:

Procedure

1. From the Admin Console, select **Trading Partner** > **Document Envelopes** > **Envelopes**.

2. Under Create, next to New Envelope, click **Go!**

3. On the Envelope Standards page, select **SWIFT** and click **Next**.
4. Select the level of envelope you want to create, inbound or outbound, and click **Next**.
5. On the Base Envelope page, do you want this envelope to inherit properties from a base envelope (if available)?
   - If Yes, select a base envelope and click **Next**.
   - If No (you want to create a new envelope), select **Not Applicable and** click **Next**.
6. On the Name page, type a unique name for the envelope, and a description or comments, then click **Next**.
7. Complete the properties for the envelope as necessary and click **Next** after each page until you reach the confirm page. Required fields are highlighted in blue. See "Inbound SWIFT Envelope" on page 21 or "Outbound SWIFT Envelope" on page 37.
8. Click **Finish** to add the envelope.

Use Base Envelopes

About this task

A *base envelope* is a regular envelope that you use as a starting point to create a new envelope. The base envelope maintains a link to the envelope that inherited its properties. If you modify the base envelope, all related envelopes (those that inherited the base envelope properties) are also changed.

When you create an envelope using a base envelope, everything in the new envelope is the same as in the base envelope, except the envelope name, description, and parameters such as unique identification numbers. If you plan to create many envelopes using base envelopes, do not use the base envelopes in production. You should also be sure to note the envelopes that are related to the base envelopes.

To use a base envelope:

Procedure

1. Create the base envelope, using "Create Envelopes" on page 19.
2. Specify the name and description of a new envelope.
3. Identify the base envelope that the new envelope uses.

Importing and Exporting Envelopes

The Import/Export feature enables you to save time and increase the accuracy of duplicating resources on different systems. This feature enables you to move resources and data between Sterling B2B Integrator environments of the same version. The Import/Export feature enables you to:

- Move from a test application environment to a production application environment.
- Move resources from one application system to another.

The ability to import and export envelopes means that you can configure resources on one system and then move or copy them to a different system, thereby avoiding having to recreate the resources on each system. Even if you have resources that are going to be slightly different from one system to another, you can export the

resources from one system and import them to a different system, and then make the necessary changes to the resource on the second system.

The Import/Export feature supports several different resource types, including envelopes.

Note: Importing an export file of envelopes always requires a passphrase, even if a passphrase was not required during the export. The passphrase is now required because of the addition of encrypted passwords that apply to some envelopes. When you are prompted for a passphrase for envelopes during the import of envelopes (when you did not use a passphrase when the envelopes were exported), you can supply any value for the passphrase.

Inbound SWIFT Envelope

You only need to create an Inbound SWIFT envelope if you are receiving inbound SWIFT messages. However, if you are receiving inbound SWIFT messages, you need to create a separate Inbound SWIFT envelope for each SWIFT message type you will be receiving. Additionally, if you are receiving SWIFT system messages, you may need to create an Inbound SWIFT envelope using a wildcard in the Message Type parameter, to ensure that all valid SWIFT messages are routed to a process whether the message has an envelope that is supported or is of a supported message type.

Note: An (*) asterisk indicates that a wildcard value can be used with that parameter (for mandatory fields, the wildcard value is an (*) asterisk and for optional fields, the wildcard value is leaving the field blank). For Inbound envelopes, a wildcard value in the envelope matches any value in the input document, while an empty value in the envelope matches only an empty value in the input document. For Outbound envelopes, a wildcard value is equivalent to an empty value in the envelope.

Field or Check Box
>   **Description**

* Sender ID
>   Coded identifier of the supplier number or data sender. Valid value is eight standard characters for BIC 8. Optional.
>
>   **Note:** This parameter enables you to type in a new ID or pick an ID that has already been used. When you start typing an ID, the Sterling B2B Integrator returns all matching IDs existing in the system and provides a combo-box from which you can select an ID by double-clicking it. There must be Sender ID Codes in the system for autocomplete to find matches and display a selection list.
>
>   **Note:** It is possible to leave the Sender ID parameter blank under some circumstances. If you leave this parameter blank, it will cause errors during the enveloping process. You must type a valid value in the Sender ID parameter.

Sender ID Type
>   Type of sender identifier. Valid values are * (wildcard), BIC8 (default), Nickname, Distinguished Name. Required.

Note: The full address for BIC8 is a BIC12, which includes the one-character logical terminal (which is specified for the sender and is always "X" for receiver), and the 3-character branch code. These are combined to form the BIC12.

* Receiver ID

Coded identifier of the customer number or data source number. Valid value is eight standard characters for BIC 8. Optional.

Note: This parameter enables you to type in a new ID or pick an ID that has already been used. When you start typing an ID, the Sterling B2B Integrator returns all matching IDs existing in the system and provides a combo-box from which you can select an ID by double-clicking it. There must be Receiver ID Codes in the system for autocomplete to find matches and display a selection list.

Note: It is possible to leave the Receiver ID parameter blank under some circumstances. If you leave this parameter blank, it will cause errors during the enveloping process. You must type a valid value in the Receiver ID parameter.

Receiver ID Type

Type of receiver identifier. Valid values are * (wildcard), BIC8 (default), Nickname, Distinguished Name. Required.

Note: The full address for BIC8 is a BIC12, which includes the one-character logical terminal (which is specified for the sender and is always "X" for receiver), and the 3-character branch code. These are combined to form the BIC12.

Envelope Format

The format of the envelope. Valid values are * (wildcard), FIN (default), and SAA XML Format 2. Required.

Message Category

The category of the message. Required. Valid values are:

- ACK/NAK (FIN only)
- Delivery Notification (XML Format 2)
- Delivery Report (XML Format 2)
- History Report (XML Format 2)
- Message (FIN or XML Format 2) (this is the default)
- Message Status (XML Format 2)
- Session Status (XML Format 2)
- Transmission Report (XML Format 2)

Enforce Message Size Limit

Specifies that the Sterling B2B Integrator will check the size of the message, and error out if the message exceeds it. If you select this check box, the Sterling B2B Integrator gives you the option to set the Maximum Message Size (which defaults to 10,000, the standard limit for SWIFT messages). Valid values are Yes (specify maximum size) - which is the default, Yes (use SWIFT-specified maximum size for message type) - which uses 2000 or 10000 as the limit, as specified in the SWIFT documentation, and No. Required.

Note: SWIFT maximum message size is only defined for MT messages. If you have defined your envelope for MX or XML messages, the SWIFT maximum message size selection is not available.

* Sender Branch Code

Three-character code that further refines the Sender ID. For example, if the SenderID is ROMANSFL, the branch code might be WAS to indicate the Washington branch. This parameter acts as the Sender ID qualifier for envelope matching. Required.

Note: Only displayed if Sender ID Type is set to BIC8. Sender Branch Code is not used for SAA XML v2.0 and does not apply when Distinguished Name and Nickname are used.

* Receiver Branch Code

Three-character code that further refines the Receiver ID. For example, if the ReceiverID is ROMANSFL, the branch code might be WAS to indicate the Washington branch. This parameter acts as the Receiver ID qualifier for envelope matching. Required.

Note: Only displayed if Receiver ID Type is set to BIC8. Receiver Branch Code is not used for SAA XML v2.0 and does not apply when Distinguished Name and Nickname are used.

Validate DataPDU Signature

Whether to validate the protocol data unit (PDU) for a signature. Optional. Valid values are Yes or No (default).

Note: Only displayed if Envelope Format is set to either wildcard or SAA XML Format 2 is selected. The signature is optional.

Key Part 1

First part of the validation signature. Each part of the key must contain at least one uppercase character, one lowercase character, and one digit. A character cannot be repeated more than seven times (across both parts of the key). Required only if you select **Yes** for **Validate DataPDU Signature**.

Key Part 2

Second part of the validation signature. Each part of the key must contain at least one uppercase character, one lowercase character, and one digit. A character cannot be repeated more than seven times (across both parts of the key). Required only if you select **Yes** for **Validate DataPDU Signature**.

Maximum Message Size

The maximum size of message that is allowed. The default is 10,000. This parameter only accepts numeric input.

Only displayed if you selected Yes for the **Enforce Message Size Limit** parameter. Required.

If you are using WebSphere Transformation Extender compliance checking, the maximum message size for MT messages will be based in the SWIFT defined maximum message size.

Note: When using WebSphere Transformation Extender compliance checking for an MT message format, the maximum message size selection is not available.

Message Format

The format of the message. You must select a Message Format of wildcard

(*), MT, MX, or Any XML. If you select a wildcard, the Message Type parameter is not displayed because it is assumed to be wildcard as well (since there is no reason to wildcard the format and then select a specific message). If you select MT as the Message Format, you will be given a list of MT messages for this Message Type parameter. If you choose MX, you receive a list of the "categories" (SWIFTNet Funds, SWIFTNet Alternative Funds, SWIFTNet Trade Services, SWIFTNet Exceptions and Investigations, SWIFTNet Cash Reporting) for this Message Type parameter. Once you select the subtype, you are presented with a list of the message types for that category, in the format camt.003.001.02 (Get Account).

MX Message Area

Transaction message type as determined by the information type in the header of the message group, which includes the message, or determined by the information type in the transaction message. Required.

Valid selections are:
- * (wildcard)
- Bulk Payments
- Cash Management
- Cash Reporting
- Exceptions and Investigations
- Funds
- Proxy Voting
- SCORE
- Trade Services
- Clearing
- Collateral Management
- Corporate Actions
- FpML
- Transaction Reporting
- Alternative Funds
- e-Mandate
- Funds Processing Passport

Note: The message area for MX messages.
The lists of Message Types are generated from two property files. The first property file (ui_swift_message_types.properties) will contain the base lists of the types provided with the Sterling B2B Integrator. The second property file (ui_swift_message_types_customer.properties) is user-modifiable, and can also contain additional messages that you want to add. If you add Message Types to the ui_swift_message_types_customer.properties property file, modify it in the customer_overrides.properties file. You are not required to restart Sterling B2B Integrator when you edit the ui_swift_message_types_customer.properties file.
You can also select the default, which is * (wildcard).

Message Type

Transaction message type as determined by the information type in the header of the message group, which includes the message, or determined by the information type in the transaction message. This list includes SWIFT system messages. Required.

Note: If you are receiving SWIFT system messages, you either need to create an Inbound SWIFT envelope using a wildcard in the Message Type parameter or create a separate Inbound SWIFT envelope for each SWIFTNet system message type you will receive, to ensure that all valid SWIFT messages are routed to a process whether the message has an envelope that is supported or is of a supported message type.

Note: The message type for MX messages are the full 4-component string (for example, camt.003.001.02). You are first allowed to select a Message Format of wildcard, MT, and MX. If you select a wildcard, this Message Type parameter is not displayed because it is assumed to be wildcard as well (since there is no reason to wildcard the format and then select a specific message). If you select MT as the Message Format, you will be given a list of MT messages for this Message Type parameter. If you choose MX, you receive a list of the "categories" (SWIFTNet Funds, SWIFTNet Alternative Funds, SWIFTNet Trade Services, SWIFTNet Exceptions and Investigations, SWIFTNet Cash Reporting) for this Message Type parameter. Once you select the subtype, you are presented with a list of the message types for that category, in the format camt.003.001.02 – Get Account Information. The envelope UI code will be modified to allow us to do this while still saving the value under a single envelope parameter. The lists of Message Types are generated from two property files. The first property file (ui_swift_message_types.properties) will contain the base lists of the types provided with the Sterling B2B Integrator. The second property file (ui_swift_message_types_customer.properties) is user-modifiable, and can also contain additional messages that you want to add. If you add Message Types to the ui_swift_message_types_customer.properties property file, modify it in the customer_overrides.properties file. You are not required to restart the Sterling B2B Integrator when you edit the ui_swift_message_types_customer.properties file.
You can also select the default, which is * (wildcard).

Scheme Instrument
Set the scheme instrument. Valid values are SCT (SEPA Credit Transfer), SDD (SEPA Direct Debit), or None. This parameter is only used for MX messages in the Bulk Payments and SCORE SWIFT Solutions.

Validation Flag (tag 119)
Specifies how to validate the message. Required. Default is * (wildcard).

Note: Be sure to configure your SWIFT envelopes appropriately for the version of SWIFT that you are using.

Reconcile Message Reference against a control number
Whether to reconcile the Message User Reference (MUR) against a control number. Valid values are Yes (default) and No.

The MUR is a message identifier separate from the one SWIFT assigns. Required.

Use global control number
Whether to use a global control number. Required. Valid values are:
- Yes (default)
- Yes (and generate name from data)

Note: If you select this option, you must select at least one parameter for Primary Name Format.

- No

Note: Only displayed if you selected Yes for **Reconcile message User Reference (MUR) against a control number**.

Perform Duplicate Control Number (MUR) Checking
> Determine control number/MUR duplications. Required. Valid values:
> - Yes
> - No (default)
>
> **Note:** Only displayed if you selected Yes for **Reconcile message User Reference (MUR) against a control number**.

Maximum age of Control Number History Records in days
> Maximum days that IBM Sterling B2B Integrator should retain a history of control numbers to use for duplication determinations. Valid value is nine numerics. Optional.
>
> **Note:** Only displayed if you selected Yes for **Reconcile message User Reference (MUR) against a control number**.

Assign control number
> Select a control number with this envelope. Optional.
>
> Displayed only if **Use Global Control Number** is set to Yes

Local control number
> Select a local control number to associate with this envelope. Default is 1. Required.
>
> Displayed only if **Use Global Control Number** is set to No.

Primary Name Format
> Check boxes to instruct what information to include when generating a name for a primary global control number and finding the correct number to assign based on that name. Optional.
>
> Select from:
> - Sender ID
> - Sender Branch Code
> - Receiver ID
> - Receiver Branch Code
> - Message Format (MT/MX/Any XML)
> - Message Type
> - Validation Flag (MT only)
> - Envelope Format (FIN/XML2)
>
> **Note:** If you are using a generated global control number (**Use global control number** is set to Yes (and generate name from data)), the Sterling B2B Integrator tries to generate and match the following control numbers:
> - First, it tries to generate and match the control number in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the control number SWIFT_100_Inbound_FIN.
> - Second, it tries to generate and match the First Backup Name.
> - Third, it tries to generate and match the Second Backup Name.

- Fourth, if there is not an existing control number that matches the names generated in steps 1-3, a control number with the name assigned in the first step is created.

First Backup Name Format

The first backup name format to use when generating a global control number. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated global control number (**Use global control number** is set to Yes (and generate name from data)), the Sterling B2B Integrator tries to generate and match the following control numbers:
- First, it tries to generate and match the control number in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the control number SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing control number that matches the names generated in steps 1-3, a control number with the name assigned in the first step is created.

Second Backup Name Format

The second backup name format to use when generating a global control number. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated global control number (**Use global control number** is set to Yes (and generate name from data)), the Sterling B2B Integrator tries to generate and match the following control numbers:

- First, it tries to generate and match the control number in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the control number SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing control number that matches the names generated in steps 1-3, a control number with the name assigned in the first step is created.

Compliance check message
Check the message body for compliance. Required. Valid values are Yes (default) and No.

MX Business Area
The MX business area. Valid values are * (wildcard), Funds, Trade Services, Exceptions and Investigations, Cash Reporting, Bulk Payments, Cash Management, Proxy Voting, SCORE, Transaction Reporting, Alternative Funds, e-Mandate, and Funds Processing Passport. Required.

Note: Only displayed if you set Message Format to MX.

Message Format
The format of the message. Valid values are * (wildcard), MT (default), MX, and Any XML. Required.

APC/FIN
For an MT message using XML Format 2, this indicates whether it is a system (APC) message or a user message (FIN). Required.

Valid values are * (wildcard—this is the default), APC, and FIN.

Scheme Instrument
Set the scheme instrument. Valid values are SCT (SEPA Credit Transfer), SDD (SEPA Direct Debit), or None. This parameter is only used for MX messages in the Bulk Payments and SCORE SWIFT Solutions.

Closed User Group
A specific set of trading partners defined within the SWIFT network. Optional.

Keep translated document after compliance check
Whether to keep the translated document after the compliance check.Valid values are Yes and No (default). Required.

Note: If you select Yes, the translated document replaces the primary document.

Validate translation output
Whether to validate the output. Valid values are Yes (default) and No.

Map Name Mode
How to determine which map to use to perform a compliance check. Required. Valid values are:

- Specify (default)

- Generate from data (this is useful if you want to share across envelopes)

  Note: If you select this option, you must select at least one parameter for Primary Name Format.

Treat message as a FileAct message for validation purposes (e.g. rule 163)
Whether to treat the message as a FileAct message for validation purposes (for example, for rule 163). Valid values are Yes (default) and No. Required.

Use WebSphere Transformation Extender compliance checking
Specifies the use of WebSphere Transformation Extender compliance checking. Valid values are Yes and No (default). This option displays only if WebSphere Transformation Extender is installed and is integrated with Sterling B2B Integrator.

For MT messages, WebSphere Transformation Extender compliance checking uses the Java Validation Component (JVC).

For MX messages, WebSphere Transformation Extender compliance checking uses the WebSphere Transformation Extender framework driver map.

If the value is Yes and the message format is MX, then the following options are available:
- Validate BIC addresses (option applied to MT and MX messages)
- Validate IBAN data (option applied to MX messages only)
- Validate currency codes (option applied to MX messages only)
- Validate amount and the number of decimal digits (option applied to MX messages only)
- Validate country codes (option applied to MX messages only)

If the value is Yes and the message format is MT, then the following option is available:
- Validate BIC addresses (option applied to MT and MX messages)

Map Name
Which map to use to perform a compliance check (if **Compliance Check message** set to Yes and **Map Name Mode** set to Yes). The map must already be checked in to the Sterling B2B Integrator. When using a Sterling B2B Integrator map, the map is either the compliance check map or the transformation map. However, when using the WebSphere Transformation Extender map, the only map available is the transformation map. Optional.

Primary Name Format
Check boxes to instruct what information to include when generating and matching a name for the map. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated map name (**Map Name Mode** is set to Generate from data), the Sterling B2B Integrator tries to generate and match the following maps:

- First, it tries to generate and match the map in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the map named SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing map that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no matching map is found** is set to Yes.

Use first backup name format
Whether to use the first backup name format if the primary format does not generate a match. Default is selected (use first backup name format).

First Backup Name Format
The first backup name format to use when generating a map name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated map name (**Map Name Mode** is set to Generate from data), the Sterling B2B Integrator tries to generate and match the following maps:

- First, it tries to generate and match the map in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the map named SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing map that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no matching map is found** is set to Yes.

Use second backup name format

Whether to use the second backup name format if the primary format does not generate a match. Default is selected (use second backup name format).

Second Backup Name Format

The second backup name format to use when generating a map name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated map name (**Map Name Mode** is set to Generate from data), the Sterling B2B Integrator tries to generate and match the following maps:
- First, it tries to generate and match the map in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the map named SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing map that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no matching map is found** is set to Yes.

Generate an error if no matching map is found

Specifies whether to generate an error if the Map Name selected is not found. Valid values are Yes (default) and No.

Note: Only displayed if **Map Name Mode** is set to Generate from data.

Determine Error Business Process Name By

How to determine the business process name to use if there were errors in the compliance check. Required. Valid values are:
- Specifying a Business Process name (default)
- Generating a Business Process name from the data (this is useful if you want to share across envelopes)

Note: If you select this option, you must select at least one parameter for Primary Name Format.

Primary Name Format

Check boxes to instruct what information to include when generating and matching a name for the error business process name. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated error business process name (**Determine Error Business Process Name By** is set to Generating a Business Process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:
- First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Use first backup name format
> Whether to use the first backup name format if the primary format does not generate a match. Default is selected (use first backup name format).

First Backup Name Format
> The first backup name format to use when generating an error business process name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

> Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

> **Note:** If you are using a generated error business process name (**Determine Error Business Process Name By** is set to Generating a Business Process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:

- First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Use second backup name format
Whether to use the second backup name format if the primary format does not generate a match. Default is selected (use second backup name format).

Second Backup Name Format
The second backup name format to use when generating an error business process name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated error business process name (**Determine Error Business Process Name By** is set to Generating a Business Process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:
- First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Generate an error if no generated business process name exists in the system
Specify whether to generate an error if there is no match to the generated business process name in the system. Valid values are Yes (default) and No.

Note: Only displayed if **Determine the Error Business Process By** is set to Generating the business process name from the data.

Business Process List
Select a previously created business process to associate with this envelope. Optional.

Displayed only if **Determine Error Business Process Name By** is set to Specifying the business process.

Determine the Business Process By
How to determine the business process name to use if there were no errors in the compliance check. Required.Valid values are:
- Specifying a business process
- Generating the business process name from the data

Note: If you select this option, you must select at least one parameter for Primary Name Format.

Primary Name Format
Check boxes to instruct what information to include when generating and matching a name for the business process. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated business process name (**Determine the Business Process By** is set to Generating the business process name from the data), Sterling B2B Integrator tries to generate and match the following business process:
- First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the first Backup Name.
- Third, it tries to generate and match the second Backup Name.
- Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Use first backup name format

Whether to use the first backup name format if the primary format does not generate a match. Default is selected (use first backup name format).

First Backup Name Format

The first backup name format to use when generating a business process name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated business process name (**Determine Business Process Name By** is set to Generating a Business Process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:
- First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Use second backup name format

Whether to use the second backup name format if the primary format does not generate a match. Default is selected (use second backup name format).

Second Backup Name Format

The second backup name format to use when generating a business process name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)

- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated business process name (**Determine Error Business Process Name By** is set to Generating a Business Process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:

- First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Inbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Generate an error if no generated business process name exists in the system
Specify whether to generate an error if there is no match to the generated business process name in the system. Valid values are Yes (default) and No.

Note: Only displayed if **Determine the Business Process By** is set to Generating the business process name from the data.

Business Process List
Select a previously created business process to associate with this envelope Optional.

Displayed only if **Determine the Business Process Name** is set to Specifying the business process.

Extraction Options
Business process data extraction. Required.

Valid values:

- Determined by business process (default)
- Extract to a file system directory
- Extract to a mailbox

Data Extraction Directory
Directory for data extraction. Displayed only if Extraction Options set to **Extract to a file system directory**. Optional

Data Extraction Filename
Filename for data extraction. Displayed only if Extraction Options set to **Extract to a file system directory**. Optional.

Data Extraction Mailbox
Mailbox for data extraction. Displayed only if Extraction Options set to **Extract to a mailbox**. Optional.

Data Extraction Mailbox Message Name
Mailbox message name for data extraction. Displayed only if Extraction Options set to **Extract to a mailbox**. Optional.

Outbound SWIFT Envelope

You only need to create an Outbound SWIFT envelope if you are sending outbound SWIFT messages. However, if you are sending outbound SWIFT messages, you need to create a separate Outbound SWIFT envelope for *each* SWIFT message type you will be sending. The following table describes Outbound SWIFT envelope properties:

Note: An (*) asterisk indicates that a wildcard value can be used with that parameter. For Inbound envelopes, a wildcard value in the envelope matches any value in the input document, while an empty value in the envelope matches only an empty value in the input document. For Outbound envelopes, a wildcard value is equivalent to an empty value in the envelope.

Field or Check Box
Description

Envelope Format
The format of the envelope. Required. Valid values are FIN (default) and SAA XML Format 2.

* Sender ID
Coded identifier of the supplier number or data sender. Valid value is eight standard characters for BIC 8. Required.

Note: This parameter enables you to type in a new ID or pick an ID that has already been used. When you start typing an ID, the Sterling B2B Integrator returns all matching IDs existing in the system and provides a combo-box from which you can select an ID by double-clicking it. There must be Sender ID Codes in the system for autocomplete to find matches and display a selection list.

It is possible to leave the Sender ID parameter blank under some circumstances. If you leave this parameter blank, it will cause errors during the enveloping process. You must type a valid value in the Sender ID parameter.

Sender ID Type
Type of sender identifier. Valid values are * (wildcard), BIC8 (default), and Distinguished Name. Required.

Note: The full address for BIC8 is a BIC12, which includes the one-character logical terminal (which is specified for the sender and is always "X" for receiver), and the 3-character branch code. These are combined to form the BIC12.

* Receiver ID
Coded identifier of the customer number or data source number. Valid value is eight standard characters for BIC 8. Required.

Note: This parameter enables you to type in a new ID or pick an ID that has already been used. When you start typing an ID, the Sterling B2B Integrator returns all matching IDs existing in the system and provides a

combo-box from which you can select an ID by double-clicking it. There must be Receiver ID Codes in the system for autocomplete to find matches and display a selection list.

It is possible to leave the Receiver ID parameter blank under some circumstances. If you leave this parameter blank, it will cause errors during the enveloping process. You must type a valid value in the Receiver ID parameter.

Receiver ID Type

Type of receiver identifier. Valid values are * (wildcard), BIC8 (default), and Distinguished Name. Required.

Note: The full address for BIC8 is a BIC12, which includes the one-character logical terminal (which is specified for the sender and is always "X" for receiver), and the 3-character branch code. These are combined to form the BIC12.

Accepter Lookup Alias

Identifying string used with the Sender ID and the Receiver ID to look up this envelope with the EDI Encoder service. This alias associates a message with the service it requires. Valid value must be at least one limited standard character. Required. Default is FIN.

Use Correlation Overrides

When to use correlation overrides (when a SWIFT Reviewer chooses not to validate a message on resend). Required. Valid values are:

- **Always** - If a correlation override value is provided, always use that value instead of the value in the envelope (except for the Sender ID, Receiver ID, and Sender/Receiver Qualifier fields, which must have a * in the envelope definition to use the correlation override value).
- **Only when the envelope field has a wildcard value (*)** - Only use correlation override values for fields in the envelope definition that are specified with a wildcard (*). This is the default.
- **Never** - Never use correlation overrides.
- **Use the default specified in enveloping.properties** - The enveloping.properties file defines when to use correlation overrides if this choice is selected. The properties file will either define Always, Never, or Only when the envelope field has a wildcard (*) value. The applicable properties file property is **enveloping.correl_override_default.[standard]**. Allowed values are No (which means never), WildcardOnly, or All (which means always, except for the Sender ID, Receiver ID, and Sender/Receiver Qualifier fields, which must have a * in the envelope definition to use the correlation override value).

Batching Options

When the EDI Encoder is used to prepare multiple messages for enveloping, you can either choose to process each message individually or concatenate them into a single file, with the messages separated by a '$'. Required.

Valid values are:

- Use batching ("$" between messages) (default)
- Do not batch messages

Enforce Message Size Limit

Specifies that the Sterling B2B Integrator will check the size of the message, and error out if the message exceeds it. If you select this check box, gives you the option to set the Maximum Message Size (which defaults to 10,000, the standard limit for SWIFT messages). Valid values are Yes (default) and No. Required.

Note: The option to use the SWIFT defined size will only be available for MT, wildcard, and File formats.

User Reference (MT Message User Reference/MX Message Reference)

Indicates how to process the message user reference. The message user reference block is an optional section in the SWIFT envelope that the SWIFT network tracks for acknowledgement purposes. Valid values are:

- Don't include a message user reference (default)
- Use control number
- Use value from process data

Note: You should only select **Use value from process data** if you are using immediate enveloping. With deferred enveloping, the process data values from the encoding step are not passed on to the enveloping step, so using a process data element will not work with deferred enveloping.

Note: Specify **Use control number** if you want to reconcile Delivery Notification (or Non-Delivery Warning message) with the Message Input Reference and the original input document.

Revision

The revision field. Values are 2.0.0. (default) and 2.0.1.

Note: This parameter is displayed on the SAA XML Format 2 Options screen when you select **Envelope Format** as **SAA XML Format 2** when creating an outbound envelope.

Message Format

The format of the message. Valid values are MT (default), MX, and Any XML. Required.

Include MX Application Header

Whether the MX Application Header should be included, and, if so, what the format is. Valid values are None (default), Standard Application Header, and ISO20022 Business Application Header. Optional.

From Type

The type of sender. Optional. Only displayed for SAA XML Format 2, Standard Application Header.

From ID

The sender's identifier. Optional. Only displayed for SAA XML Format 2, Standard Application Header.

To Type

The type of receiver. Optional. Only displayed for SAA XML Format 2, Standard Application Header.

To ID The receiver's identifier. Optional. Only displayed for SAA XML Format 2, Standard Application Header.

Service Name

The service name. Optional. Only displayed for SAA XML Format 2, Standard Application Header.

Message Name

The name of the message. Optional. Only displayed for SAA XML Format 2, Standard Application Header.

Duplicate Reference

Duplicate reference information, if any. Optional. Only displayed for SAA XML Format 2, Standard Application Header.

Duplicate Information

Information about a duplicate header, if applicable. Optional. Only displayed for SAA XML Format 2, Standard Application Header.

Character Set

The character set that will be used. Required.

Note: Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

Add Identity

If necessary, allows you to add an identity to associate with the header. Optional.

Note: Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

Name   Name of your organization. Required.

Reference URL

Internet address of your organization. Optional.

DUNS

Unique 9-13 character identification number of your organization, as issued by Dun & Bradstreet Corporation. Required.

Identifier

Primary ID of your organization. Use DUNS number. Required.

Address Line 1

Conventional mailing address of your organization. Required.

Address Line 2

Additional mailing address information of your organization. Optional.

City   City where your organization is located. Required.

State or Province

State or province where your organization is located. Required.

Zip or Postal Code

ZIP Code or postal code where your organization is located. Required.

E-mail Address

E-mail address of your organization. Required.

Telephone Number

Telephone number of your organization. Required.

Time Zone

Time zone where your organization is located. Optional.

Country or Region

Country or region where your organization is located. Optional.

VAT ID

The VAT transaction tax identifier. Optional.

Note: When you add VAT IDs to an identity, an associated permission ID is created, and the user is assigned that permission ID.

Internet Notification E-mail

The internal e-mail address to which you want error notifications to be sent. Optional.

Lookup ID

The identifier used to look up the agreement. Optional.

Lookup ID Comments

Any comments regarding the identifier used to look up the agreement. Optional.

Include ISO20022 Extensions

Whether to include ISO20022 extensions. Default is unchecked (do not included extensions). Optional.

ISO Type

The type of ISO. Valid selections are: Financial Institution (default), Organization, Branch, and Person. Optional.

Clearing System Member

The envelope for the clearing system member. Optional.

Other Another envelope to use. Optional.

Address Type

The type of address. Valid selections are: Postal (default), Business, DeliveryTo, Residential, MailTo, and POBox. Optional.

Country or Region of Residence

The country or region of residence. Optional.

Department

The name of the department. Optional.

Sub-Department

The name of the sub-department, if any. Optional.

Street Name

The name of the street in the address. Optional.

Building Number

The building number in the address. Optional.

Address Line 3

Another address line, if necessary. Optional.

Address Line 4

Another address line, if necessary. Optional.

Address Line 5

Another address line, if necessary. Optional.

Address Line 6

Another address line, if necessary. Optional.

Address Line 7
> Another address line, if necessary. Optional.

Issuer The ISO20022 issuer. Optional.

Code The code, if any. Optional.

Proprietary
> Proprietary information, if any. Optional.

Clearing System Member ID
> The clearing system member identifier. Optional.

From The sender of the message. Required.

> **Note:** Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

To The receiver of the message. Required.

> **Note:** Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

Business Message Identifier
> The business message identifier. Required.

> **Note:** Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

Message Definition Identifier
> The message definition identifier. Required.

> **Note:** Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

Business Service
> The business service. Optional.

> **Note:** Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

Copy Duplicate
> Indicates whether the header is a duplicate or copy. Valid selections are: Copy of a Duplicate, Copy, and Duplicate. Optional.

> **Note:** Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

Possible Duplicate
> Whether the header is a possible duplicate. Required. Valid values are Yes and No (default). Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

Priority
> The network priority. Optional.

> **Note:** Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

Related
> Enables you to specify a related envelope. Optional. Only displayed for SAA XML Format 2, ISO20022 Business Application Header.

Validate Sender and Receiver

Specify whether to validate the sender and receiver. Required. Valid values are Yes and No (default).

This parameter allows you to enable (Yes) or disable (No, which is the default) address verification. Address verification is performed using the SWIFT_Addresses and SWIFT_BaseAddresses code lists. See "Maintain the External Code Lists" on page 18 for more information on creating these code lists.

Note: This parameter is only available if the following selections are made:
- Message format is MT or File
- Use WebSphere Transformation Extender Compliance Checking is No
- Envelope is for an outbound message

MX Business Area

The MX business area. Valid values are * (wildcard), Funds, Trade Services, Exceptions and Investigations, Cash Reporting, Bulk Payments, Cash Management, Proxy Voting, SCORE, Transaction Reporting, Alternative Funds, e-Mandate, and Funds Processing Passport. Required.

Note: Only displayed if you set Message Format to MX.

FIN/APC

For an MT message using XML Format 2, this indicates whether it is a system (APC) message or a user message (FIN). Required.

Valid values are * (wildcard—this is the default), APC, and FIN.

MX Message Area

Transaction message type as determined by the information type in the header of the message group, which includes the message, or determined by the information type in the transaction message. Required.

Valid selections are:
- * (wildcard)
- Bulk Payments
- Cash Management
- Cash Reporting
- Clearing
- Collateral Management
- Corporate Actions
- e-Mandate
- Exceptions and Investigations
- Funds
- Alternative Funds
- Funds Processing Passport
- Proxy Voting
- SCORE
- Trade Services
- Transaction Reporting
- ISO 20022

Note: The message area for MX messages. The lists of Message Types are generated from two property files. The first property file (ui_swift_message_types.properties) will contain the base lists of the types provided with the Sterling B2B Integrator. The second property file (ui_swift_message_types_customer.properties) is user-modifiable, and can also contain additional messages that you want to add. If you add Message Types to the ui_swift_message_types_customer.properties property file, modify it in the customer_overrides.properties file. You are not required to restart the Sterling B2B Integrator when you edit the ui_swift_message_types_customer.properties file. You can also select the default, which is * (wildcard).

Message Type

Transaction message type as determined by the information type in the header of the message group, which includes the message, or determined by the information type in the transaction message. Required for both MT and MX messages.

Note: The message type for MX messages are the full 4-component string (for example, camt.003.001.02). You are first allowed to select a Message Format of wildcard, MT, and MX. If you select a wildcard, this Message Type parameter is not displayed because it is assumed to be wildcard as well (since there is no reason to wildcard the format and then select a specific message). If you select MT as the Message Format, you will be given a list of MT messages for this Message Type parameter. If you choose MX, you receive a list of the "categories" (SWIFT Funds, SWIFT Trade Services, SWIFT Exceptions and Investigations, SWIFT Cash Reporting) for this Message Type parameter. Once you select the subtype, you are presented with a list of the message types for that category. The lists of Message Types are generated from two property files. The first property file (ui_swift_message_types.properties) will contain the base lists of the types provided with the Sterling B2B Integrator. The second property file (ui_swift_message_types_customer.properties) is user-modifiable, and can also contain additional messages that you want to add. If you add Message Types to the ui_swift_message_types_customer.properties property file, modify it in the customer_overrides.properties file. You are not required to restart the Sterling B2B Integrator when you edit the ui_swift_message_types_customer.properties file.

Service Identifier

A two-character numeric field indicating the type of data. The default is 01 for all the Sterling B2B Integrator and user-to-user messages, 21 for acknowledgements, and 03 for SELECT commands. Required.

FIN/APC

For an MT message using XML Format 2, this indicates whether it is a system (APC) message or a user message (FIN). Required.

Valid values are * (wildcard—this is the default), APC, and FIN.

Closed User Group

A specific set of trading partners defined within the SWIFT network. Optional.

Include sender full name

Whether to include the full name of the sender. Valid values are Yes and No (default). Required.

X1 (Institution BIC11)
> The name of the sending institution. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

X2 (Department or Application Name)
> The name of the sending department or application name. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

X3 (Routing Information/Surname)
> The routing information or surname of the sender. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

X4 (Given Name)
> The given name of the sender. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

Financial Institution
> The financial institution for the sender. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

Branch Information
> The branch information for the sender. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

City Name
> The name of the city for the sender. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

Location
> The location code for the sender. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

Country or Region Code
> The country or region code for the sender. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

Include receiver full name
> Whether to include the full name of the receiver. Valid values are Yes and No (default). Required.

X1 (Institution BIC11)
> The name of the receiving institution. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

X2 (Department or Application Name)
> The name of the receiving department or application name. Optional.
>
> **Note:** Only displayed for SAA XML Format 2.

X3 (Routing Information/Surname)
> The routing information or surname of the receiver. Optional.

Note: Only displayed for SAA XML Format 2.

X4 (Given Name)
>The given name of the receiver. Optional.

>**Note:** Only displayed for SAA XML Format 2.

Financial Institution
>The financial institution for the receiver. Optional.

>**Note:** Only displayed for SAA XML Format 2.

Branch Information
>The branch information for the receiver. Optional.

>**Note:** Only displayed for SAA XML Format 2.

City Name
>The name of the city for the receiver. Optional.

>**Note:** Only displayed for SAA XML Format 2.

Location
>The location code for the receiver. Optional.

>**Note:** Only displayed for SAA XML Format 2.

Country or Region Code
>The country or region code for the receiver. Optional.

>**Note:** Only displayed for SAA XML Format 2.

Routing Code
>The routing code. Optional.

>**Note:** Only displayed for SAA XML Format 2.

Validation level
>The level of validation. Required. Valid values are None (use SAA default), Minimum, Intermediate, Maximum.

>**Note:** Only displayed for SAA XML Format 2.

Allow modification in SAA
>Whether to allow modification in SAA. Required. Valid values are SAA Default (this is the default), Yes, and No.

>**Note:** Only displayed for SAA XML Format 2.

Include routing instructions
>Whether to include routing instructions. Required. Valid values are Yes and No (default).

>**Note:** Only displayed for SAA XML Format 2.

Routing Function
>The routing function. Required. Valid values are Route (default), DisposeToRoutingPoint, and DisposeToRoutingStep.

>**Note:** Only displayed for SAA XML Format 2, if Include routing instructions is set to Yes.

Routing Point

The routing point. Optional.

Note: Only displayed for SAA XML Format 2, if Include routing instructions is set to Yes.

Routing Step

The routing step. Optional. Valid values are None (default), Verify, Authorise, Modify, and ReadyToSend.

Note: Only displayed for SAA XML Format 2, if Include routing instructions is set to Yes.

Network Priority

The network priority. Optional. Valid values are None (default), Normal, System, and Urgent.

Note: Only displayed for SAA XML Format 2.

Is Possible Duplicate

Whether the trailer is a possible duplicate. Required. Valid values are Don't include this tag (default), False, and True. Only displayed for SAA XML Format 2.

Request Notification

Whether delivery notification is requested. Required. Valid values are Don't Include (default), False, and True.

Note: Only displayed for SAA XML Format 2.

The Sterling B2B Integrator does not mark a message as accepted until the delivery notification or report has been received. Therefore, the Delivery Notification information is stored until the Transmission Report associating the SenderReference with the appropriate Reconciliation information is received.

Service

The network service used. Optional.

Note: Only displayed for SAA XML Format 2.

User Priority (FIN only; header field 113)

The user priority. Optional.

Note: Only displayed for SAA XML Format 2.

Value Added Service ID (FIN only)

The value-added network service used. Optional.

Note: Only displayed for SAA XML Format 2.

Standard Year (Version)

The SWIFT version (year of release). Optional.

Validation Identifier (FIN only; header field 119)

The validation identifier. Optional.

Note: Only displayed for SAA XML Format 2.

IsSigningRequested flag (ignored for FIN)

Whether or not a signature is requested as a security option. Optional. Valid values are None (use SAA emission profile configuration—this is the default), True, and False.

Note: Only displayed for SAA XML Format 2.

Request non-repudiation (SWIFTNet only)

Whether or not non-repudiation is requested as a security option. Optional. Valid values are None (use SAA emission profile configuration—this is the default), True, and False.

Note: Only displayed for SAA XML Format 2.

PAC Value

This the PAC value Optional

Note: Only displayed for SAA XML Format 2.

Sign the DataPDU

Whether to sign the protocol data unit (PDU) for a signature. Optional. Valid values are Yes or No (default).

Note: Only displayed if Envelope Format is set to either wildcard or SAA XML Format 2 is selected. The signature is optional.

Key Part 1

First part of the validation signature.

Key Part 2

Second part of the validation signature.

Message Priority

Specify the priority of the message delivery. Optional. Valid values are:

- Normal (default)
- Urgent
- System

Note: If you are requesting Delivery Notification andNon-Delivery Warning messages, you need to set the Priority and Delivery Options in the envelope: Normal indicates None and Delivery Notification and Urgent indicates Non-Delivery Warning and Both Non-Delivery Warning and Delivery Notification.

Delivery Monitoring

One digit that indicates how monitoring will be performed by the SWIFT network. Optional.

Valid values are:

- No Delivery Monitoring (default)
- 1 (Warning Message) indicates that a warning message will be given if the message is not delivered within a reasonable period
- 2 (Delivery Notification) indicates notification when the message is delivered
- 3 (Non-Delivery Warning and Delivery Notification) indicates that both the non-delivery warning and delivery notification will be given

Note: The allowable options are tied to the **Message Priority**: priority **Urgent** requires that the user select 1 or 3, priority **Normal** enables the user to select 2 or No Delivery Monitoring.

Note: If Delivery Monitoring is enabled, two Correlations are created when a message is sent. These Correlations are the same names used by the XML Format 2 envelope: MessageStatusIsSuccess and ReceiverDeliveryStatus. The values for both Correlations are initialized to WAITING. When the Delivery Notification or Non-Delivery Warning message is processed these Correlation values will be updated appropriately. If a Delivery Notification message is processed, that means the message has been delivered to the recipient. Therefore, **ReceiverDeliveryStatus** is set to RcvDelivered and **MessageStatusIsSuccess** is set to true. If a Non-Delivery Warning message is processed, then the **ReceiverDeliveryStatus** will be set to RcvDelayedNak, indicating that the a message has been rejected by FIN. For a Non-Delivery Warning message, **MessageStatusIsSuccess** is set to false. Likewise, if a NAK message is processed, the **ReceiverDeliveryStatus** is set to RcvDelayedNak because the message has been rejected by FIN and will not be delivered.

Obsolescence Period
Indicates the time after which a delayed message trailer will be added to the message by the SWIFT network if it has not yet been delivered. This is also the period after which a non-delivery warning will be generated by the SWIFT network, if the appropriate choice for the **Delivery Monitoring** parameter has been selected (that is, if you selected option **3** for **Delivery Monitoring**). Optional.

Note: This parameter must be three numerics. Each unit represents five minutes (so, for example, 003 equates to 15 minutes). SWIFT requires leading zeros, so the if the number of minutes is less than 3 digits, you must include leading zeroes.

Use MQHA Delivery Notification Reconciliation
Allows you to specify whether to reconcile Delivery Notification messages with the original message. Valid values are Yes and No (default - this specifies that just the ACK/NAK message will be reconciled). Required.

FIN Copy Service Code (tag 103):
A typical configuration requires that the FIN Copy Service Code tag be included in the envelope (usually set to **COP**). Optional.

The SWIFT network support the FIN Copy mode, in which a message is sent to an intermediary for approval before it goes to its final destination (or is just copied to the intermediary without requiring approval).

Banking Priority (tag 113):
A four-character optional tag indicating the banking priority. The allowed values are agreed on by you and your trading partner or partners. Optional.

Validation Flag (tag 119):
Specifies the validation flag. Required.

This is an optional part of the header that can contain a code word to indicate that certain types of validations should be performed on the enveloped message.

The valid values for this tag vary depending on the message type.

Note: Be sure to configure your SWIFT envelopes appropriately for the version of SWIFT that you are using.

Payment Release Information (tag 115):
Specifies the payment release information. Optional.

This is an envelope component used in FIN Copy that contains information from the central institution to the receiver of the payment message. The information from this parameter will be placed by the SWIFT network into the MT 097 FIN Copy Message Authorization/Refusal Notification in Y-copy mode.

Include Possible Duplicate Emission (PDE) Trailer
Indicates whether to include a trailer specifying that this message may be a duplicate. Required.

This is an optional component of the envelope that indicates that this message may already have been sent. For example, if the system crashes during the delivery of a message, another copy of the message could be sent, with this trailer included to indicate that it may be a duplicate.

Valid values are Yes and No (default).

Include Training (TNG) Trailer
Specifies whether to include a training (TNG) trailer. Required.

This is an optional component of the envelope that indicates the message contained is being sent for system testing purposes.

Valid values are Yes and No (default).

Sender Logical Terminal
Specifies the logical terminal for outbound FIN. This is a single alphanumeric character Required.

Note: This parameter is necessary to send messages through the WebSphere MQ adapter using the FileAct protocol.

* Sender Branch Code
Three-character code that further refines the Sender ID. For example, if the SenderID is ROMANSFL, the branch code might be WAS to indicate the Washington branch. This parameter acts as the Sender ID qualifier for envelope matching. Required.

* Receiver Branch Code
Three-character code that further refines the Receiver ID. For example, if the ReceiverID is ROMANSFL, the branch code might be WAS to indicate the Washington branch. This parameter acts as the Receiver ID qualifier for envelope matching. Required.

Maximum Message Size
The maximum size of message that is allowed. The default is 10,000. This parameter only accepts numeric input.

Only displayed if you selected Yes for the **Enforce Message Size Limit** parameter. Required.

Note: When using WebSphere Transformation Extender compliance checking for an MT message format, the maximum message size selection is not available.

Expect an acknowledgement for messages sent using this envelope
> Whether to expect an acknowledgement for messages that are sent using this envelope. Valid values are Yes and No (default). Required.

Acknowledgement overdue after this many hours
> Amount of time, in hours, within which you must receive an acknowledgement. Valid value is four numeric characters. Optional.

Acknowledgement overdue after this many minutes
> Amount of time, in minutes, within which you must receive an acknowledgement. Valid value is four numeric characters. Optional.

Use global control number
> Whether to use a global control number. Required. Valid values are:
> * Yes (default)
> * Yes (and generate from data)
>
> **Note:** If you select this option, you must select at least one parameter for Primary Name Format.
> * No
>
> Displayed only if **Message User Reference** is set to Use Control Number.

Primary Name Format
> Check boxes to instruct what information to include when generating a name for a primary global control number and finding the correct number to assign based on that name. Optional.
>
> Select from:
> * Sender ID
> * Sender Branch Code
> * Receiver ID
> * Receiver Branch Code
> * Message Format (MT/MX/Any XML)
> * Message Type
> * Validation Flag (MT only)
> * Envelope Format (FIN/XML2)
>
> **Note:** If you are using a generated global control number (**Use global control number** is set to Yes (and generate name from data)), the Sterling B2B Integrator tries to generate and match the following control numbers:
> * First, it tries to generate and match the control number in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the control number SWIFT_100_Outbound_FIN.
> * Second, it tries to generate and match the First Backup Name.
> * Third, it tries to generate and match the Second Backup Name.
> * Fourth, if there is not an existing control number that matches the names generated in steps 1-3, a control number with the name assigned in the first step is created.

Use first backup name format
> Whether to use the first backup name format if the primary format does not generate a match. Default is selected (use first backup name format).

First Backup Name Format

The first backup name format to use when generating a global control number. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated global control number (**Use global control number** is set to Yes (and generate name from data)), the Sterling B2B Integrator tries to generate and match the following control numbers:
- First, it tries to generate and match the control number in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the control number SWIFT_100_Outbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing control number that matches the names generated in steps 1-3, a control number with the name assigned in the first step is created.

Use second backup name format

Whether to use the second backup name format if the primary format does not generate a match. Default is selected (use second backup name format).

Second Backup Name Format

The second backup name format to use when generating a global control number. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated global control number (**Use global control number** is set to Yes (and generate name from data)), the Sterling B2B Integrator tries to generate and match the following control numbers:

- First, it tries to generate and match the control number in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the control number SWIFT_100_Outbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing control number that matches the names generated in steps 1-3, a control number with the name assigned in the first step is created.

Global Message User Reference (MUR) Control Number
Select the control number with which to start generating control numbers. Optional.

Displayed only if **Message User Reference** is set to Use Control Number.

Process Data element from which to set the Message User Reference
Specify the process data element from which the MUR will be set.

Displayed only if **Message User Reference** is set to Use value from process data. Required.

Local Message User Reference (MUR)
The message user reference block is an optional section in the SWIFT envelope that the SWIFT network tracks for acknowledgement purposes. If you choose to use a local control number as the user reference, type the starting value of that control number in this parameter.

Translate documents prior to enveloping
Whether to translate the documents prior to enveloping them. Valid values are Yes or No (default). Required.

Map Name Mode
How to determine which map to use to translate the message. Required. Valid values are:
- Specify (default)
- Generate from data (this is useful if you want to share across envelopes)

Note: If you select this option, you must select at least one parameter for Primary Name Format. This parameter is displayed only if **Translate document prior to enveloping** set to Yes.

Treat message as a FileAct message for validation purposes (e.g. for rule 163)
Whether to treat the message as a FileAct message for validation purposes (for example, for rule 163). Valid values are Yes (default) and No. Required.

Use WebSphere Transformation Extender compliance checking
Specifies the use of WebSphere Transformation Extender compliance checking. Valid values are Yes and No (default). This option displays only if WebSphere Transformation Extender is installed and is integrated with Sterling B2B Integrator.

For MT messages, WebSphere Transformation Extender compliance checking uses the Java Validation Component (JVC).

For MX messages, WebSphere Transformation Extender compliance checking uses the WebSphere Transformation Extender framework driver map.

If the value is Yes and the message format is MX, then the following options are available:

- Validate BIC addresses (option applied to MT and MX messages)
- Validate IBAN data (option applied to MX messages only)
- Validate currency codes (option applied to MX messages only)
- Validate amount and the number of decimal digits (option applied to MX messages only)
- Validate country codes (option applied to MX messages only)

If the value is Yes and the message format is MT, then the following option is available:

- Validate BIC addresses (option applied to MT and MX messages)

Map Name

Which map to use to perform a compliance check (if **Translate document prior to enveloping** set to Yes and **Map Name Mode** set to Yes). The map must already be checked in to the Sterling B2B Integrator. When using a Sterling B2B Integrator map, the map is either the compliance check map or the transformation map. However, when using the WebSphere Transformation Extender map, the only map available is the transformation map. Optional.

Primary Name Format

Check boxes to instruct what information to include when generating and matching a name for the map. Optional.

Select from:

- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated map name (**Map Name Mode** is set to Generate from data), the Sterling B2B Integrator tries to generate and match the following maps:

- First, it tries to generate and match the map in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the control number SWIFT_100_Outbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing map that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no matching map is found** is set to Yes.

Use first backup name format

Whether to use the first backup name format if the primary format does not generate a match. Default is selected (use first backup name format).

First Backup Name Format

The first backup name format to use when generating a map name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:

- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated map name (**Map Name Mode** is set to Generate from data), the Sterling B2B Integrator tries to generate and match the following maps:

- First, it tries to generate and match the map in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the control number SWIFT_100_Outbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing map that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no matching map is found** is set to Yes.

Use second backup name format

Whether to use the second backup name format if the primary format does not generate a match. Default is selected (use second backup name format).

Second Backup Name Format

The second backup name format to use when generating a map name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:

- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)

- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated map name (**Map Name Mode** is set to Generate from data), the Sterling B2B Integrator tries to generate and match the following maps:

- First, it tries to generate and match the map in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the control number SWIFT_100_Outbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing map that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no matching map is found** is set to Yes.

Generate an error if no matching map is found
Specifies whether to generate an error if the Map Name selected is not found. Valid values are Yes (default) and No.

Note: Only displayed if **Map Name Mode** is set to Generate from data.

On a translation error, determine the Business Process by
How to determine the business process name to use if there were errors in the translation process. Required. Valid values are:

- Specifying a Business Process name (default)
- Generating a Business Process name from the data (this is useful if you want to share across envelopes)

Note: If you select this option, you must select at least one parameter for Primary Name Format.

If you select a translation error business process, when a document with compliance errors is encountered an instance of the error business process is generated, using the non-compliant document as the primary document. The processing of that document within the enveloper is halted at that point, and the next document starts processing. If you do not specify a translation error business process, the enveloper continues to process the non-compliant document. In either case, the status report containing the translation errors is created.
If HALT_ON_TRANS_ERROR is set to Yes or True in ProcessData, no more documents are processed after a non-compliant document is encountered. If HALT_ON_TRANS_ERROR is set to No or False in ProcessData, the rest of the documents will be processed after a non-compliant document is encountered. If HALT_ON_TRANS_ERROR is not defined, the behavior depends on the enveloping mode; IMMEDIATE mode behaves as if HALT_ON_TRANS_ERROR is set to True, and DEFERRED mode behaves as if HALT_ON_TRANS_ERROR is set to False.

Primary Name Format

Check boxes to instruct what information to include when generating and matching a name for the error business process name. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated error business process name (**On a translation error, determine Error Business Process Name By** is set to Generating a Business Process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:

- First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Outbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Use first backup name format

Whether to use the first backup name format if the primary format does not generate a match. Default is selected (use first backup name format).

First Backup Name Format

The first backup name format to use when generating an error business process name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

Select from:
- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated error business process name (**On a translation error, determine Error Business Process Name By** is set to Generating a Business Process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:

- First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Outbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Use second backup name format
    Whether to use the second backup name format if the primary format does not generate a match. Default is selected (use second backup name format).

Second Backup Name Format
    The second backup name format to use when generating an error business process name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.

    Select from:
    - Sender ID
    - Sender Branch Code
    - Receiver ID
    - Receiver Branch Code
    - Message Format (MT/MX/Any XML)
    - Message Type
    - Validation Flag (MT only)
    - Envelope Format (FIN/XML2)

    Note: If you are using a generated error business process name (**On a translation error, determine Error Business Process Name By** is set to Generating a Business Process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:

    - First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Outbound_FIN.
    - Second, it tries to generate and match the First Backup Name.
    - Third, it tries to generate and match the Second Backup Name.
    - Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Generate an error if no generated business process name exists in the system

Specify whether to generate an error if there is no match to the generated business process name in the system. Valid values are Yes (default) and No.

Note: Only displayed if **On a translation error, determine the Error Business Process By** is set to Generating the business process name from the data.

Specifying a business process

Select a previously created business process to associate with this envelope.

Displayed only if **On a translation error, determine the Business Process by** is set to Specifying a business process names. Optional.

Determine the Business Process By

How to determine the business process name to use if there were no errors in the compliance check. Required. Valid values are:

- Specify a Business Process (default)
- Generate Business Process Name (from the data)

Note: If you select this option, you must select at least one parameter for Primary Name Format.

Primary Name Format

Check boxes to instruct what information to include when generating and matching a name for the business process. Optional.

Select from:

- Sender ID
- Sender Branch Code
- Receiver ID
- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated business process name (**Determine the Business Process By** is set to Generating the business process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:

- First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Outbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Use first backup name format
> Whether to use the first backup name format if the primary format does not generate a match. Default is selected (use first backup name format).

First Backup Name Format
> The first backup name format to use when generating a business process name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.
>
> Select from:
> - Sender ID
> - Sender Branch Code
> - Receiver ID
> - Receiver Branch Code
> - Message Format (MT/MX/Any XML)
> - Message Type
> - Validation Flag (MT only)
> - Envelope Format (FIN/XML2)
>
> **Note:** If you are using a generated business process name (**Determine the Business Process By** is set to Generating a Business Process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:
> - First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Outbound_FIN.
> - Second, it tries to generate and match the First Backup Name.
> - Third, it tries to generate and match the Second Backup Name.
> - Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.
>
> Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Use second backup name format
> Whether to use the second backup name format if the primary format does not generate a match. Default is selected (use second backup name format).

Second Backup Name Format
> The second backup name format to use when generating a business process name. The system allows for two alternatives if the Primary Name Format is not found—checking for the First Backup Name Format and then, if that is not found, checking for the Second Backup Name Format. Optional.
>
> Select from:
> - Sender ID
> - Sender Branch Code
> - Receiver ID

- Receiver Branch Code
- Message Format (MT/MX/Any XML)
- Message Type
- Validation Flag (MT only)
- Envelope Format (FIN/XML2)

Note: If you are using a generated business process name (**Determine the Business Process By** is set to Generating a Business Process name from the data), the Sterling B2B Integrator tries to generate and match the following business process:

- First, it tries to generate and match the business process in the primary name format (replacing the values that are selected for the corresponding value in the message). For example, if only message type is selected, and the message type is 100, the Sterling B2B Integrator checks for the business process SWIFT_100_Outbound_FIN.
- Second, it tries to generate and match the First Backup Name.
- Third, it tries to generate and match the Second Backup Name.
- Fourth, if there is not an existing business process that matches the names generated in steps 1-3, an error is generated.

Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Generate an error if no generated business process name exists in the system
Specify whether to generate an error if there is no match to the generated business process name in the system. Required. Valid values are Yes (default) and No.

Note: Only displayed if **Determine the Business Process By** is set to Generate Business Process Name.

Business Process List
Select a previously created business process to associate with this envelope. Optional.

Displayed only if **Determine the Business Process Name** is set to Specify Business Process.

Extraction Options
Business process data extraction. Required.

Valid values:
- Determined by business process (default)
- Extract to a file system directory
- Extract to a mailbox

Data Extraction Directory
Directory for data extraction. Displayed only if **Extraction Options** set to Extract to a file system directory. Optional.

Data Extraction Filename
Filename for data extraction. Displayed only if **Extraction Options** set to Extract to a file system directory. Optional.

Data Extraction Mailbox
Mailbox for data extraction. Displayed only if **Extraction Options** set to Extract to a mailbox. Optional.

Data Extraction Mailbox Message Name

Mailbox message name for data extraction. Displayed only if **Extraction Options** set to Extract to a mailbox. Optional.

Configuring the EDI Encoder Service for SWIFT Outbound Messages

The EDI Encoder service determines which envelope will be used on the document. If translations are specified in an envelope, the service determines which map to use. For SWIFT outbound messages you need to configure five parameters to allow the outbound message to be sent correctly. See *EDI Encoder Service* for more information.

To configure the EDI Encoder service, you must specify settings for the following fields in the GPM that match the values you configured in the outbound envelope:

- AccepterLookupAlias
- ReceiverID
- SenderID
- ReceiverIDQual
- SenderIDQual

You need to set the sender and receiver variables based on the desired source and destination for the outbound message. These variables also match the corresponding values in the envelope, except that the envelope allows wildcards (which will match documents encoded with any value for that variable), whereas the value used in the EDI encoder must always be the full eight-character identifier and three-character branch code of the desired source and destination.

Or, if you are editing the EDI Encoder BPML, you need to include these parameters and values, where for xxx you substitute the values set up in the outbound envelope:

```
<operation name="EDI Encode">
  <participant name="EDIEncoder"/>
  <output message="EDIEnc_In">
    <assign to="AccepterLookupAlias">xxx</assign>
    <assign to="ReceiverID">xxx</assign>
    <assign to="SenderID">xxx</assign>
    <assign to="ReceiverIDQual">xxx</assign>
    <assign to="SenderIDQual">xxx</assign>
    <assign to="." from="*"/>
  </output>
  <input message="EDIEnc_Out">
    <assign to="." from="*"/>
  </input>
</operation>
```

Correlation Names to Override SWIFT Envelope Values

Overrides are used (typically in conjunction with wildcard values in envelopes) to reduce the number of envelopes that you need to create. For example, if you send several types of messages to the same receiver, you can set up a single envelope for this receiver and use an override for the message type, rather than creating a different envelope for each message type you wish to send.

To use an override, you need to set the appropriate correlation on the document prior to running it through the EDI Encoder service. For example, you can set the correlation **Out_MessageType** to **999** if you want to override the message type.

For Inbound envelopes, a wildcard value in the envelope matches any value in the
input document, while an empty value in the envelope matches only an empty
value in the input document. For Outbound envelopes, a wildcard value is
equivalent to an empty value in the envelope. You can override wildcards with
specific EDI values in outbound processing. You must supply the EDI data to the
business process and the data must be in the correct format (that is, name/value
pairs).

When setting up your outbound processing, consider the following:
* If an envelope field contains a wildcard, you must supply a correlation value for
  it or the service halts with an error.
* You can override wildcard values in an envelope by using the Correlation
  service to pass name/value pairs from the primary document to the EDI
  Encoder service.
* If an envelope contains specific values in the Sender ID, Sender ID Qualifier,
  Receiver ID, or Receiver ID Qualifier fields (that is, values other than a
  wildcard), the values passed from the Correlation service to the EDI Encoder
  service do not overwrite the values in the fields.
* If you are using the same envelope for inbound processing and
  acknowledgements, supply the Sender ID, Receiver ID, and Qualifiers in the
  envelope so that they are not overwritten by the correlation values.

FIN Envelopes

This section contains the override values for FIN envelopes. The values are
organized by the envelope section in which they are located.

The basic header block overrides are as follows:
* Out_ServiceIdentifier
* Out_SenderID

  Note: This is the BIC8 of the sender.
* Out_SenderLogicalTerminal
* Out_SenderIDQual

  Note: This is the branch code of the sender.
* Out_SessionNumber
* Out_SequenceNumber

The Sterling B2B Integrator header overrides are as follows:
* Out_MessageType
* Out_ReceiverID

  Note: This is the BIC8 of the receiver.
* Out_ReceiverIDQual

  Note: This is the branch code of the receiver.
* Out_MessagePriority
* Out_DeliveryMonitoring
* Out_ObsolesencePeriod
* Out_ValidationFlag

The user header overrides are as follows:
- Out_ServiceCode: Block 103
- Out_BankingPriority: Block 113
- Out_ControlNumber: Block 108

   Note: This is the message user reference.
- Out_ValidationFlag: Block 119
- Out_PaymentReleaseInfo: Block 115

The TNG trailer override is as follows:
- Out_IncludeTNG

   Note: Set this to Yes to include the TNG trailer.

The PDE trailer overrides are as follows:
- Out_SWIFTPDEMessageInputTime
- Out_SWIFTPDEMessageInputDate
- Out_SWIFTPDEInputAddress
- Out_SWIFTPDEInputSessionNumber
- Out_SWIFTPDEInputSequenceNumber

XML Format 2 Envelopes

This section contains the override values for XML Format 2 envelopes. The values are organized by the envelope section in which they are located.

Out_MessageFormat: Value for the Format element within the Message element

The Sender element overrides are as follows:
- Out_SenderID

   Note: The BIC8 of the sender (for BIC12 senders), or the DN or Nickname for the sender.
- Out_SenderLogicalTerminal

   Note: This is the logical terminal (for BIC12 senders).
- Out_SenderIDQual

   Note: This is the branch code of the sender (for BIC12 senders).
- Out_SenderX1
- Out_SenderX2
- Out_SenderX3
- Out_SenderX4
- Out_SenderFinancialInstitution
- Out_SenderBranchInformation
- Out_SenderCityName
- Out_SenderLocation
- Out_SenderCountryCode

The Receiver element overrides are as follows:

- Out_ReceiverID
- This is the BIC8 of the receiver (for BIC12 receivers), or the DN or Nickname for the receiver.
- Out_ReceiverIDQual

  Note: This is the branch code of the receiver (for BIC12 receivers).
- Out_ReceiverX1
- Out_ReceiverX2
- Out_ReceiverX3
- Out_ReceiverX4
- Out_ReceiverFinancialInstitution
- Out_ReceiverBranchInformation
- Out_ReceiverCityName
- Out_ReceiverLocation
- Out_ReceiverCountryCode

The InterfaceInfo overrides are as follows:
- Out_ControlNumber
- Out_XML2RoutingCode
- Out_XML2ValidationLevel
- Out_XML2AllowModification
- Out_XML2RoutingFunction
- Out_XML2RoutingPoint
- Out_XML2RoutingStep

The NetworkInfo overrides are as follows:
- Out_XML2Priority
- Out_XML2IsPossibleDuplicate
- Out_XML2IsNotificationRequested
- Out_XML2Service
- Out_XML2UserPriority
- Out_XML2CopyService
- Out_ValidationFlag
- Out_XML2IsSigningRequested
- Out_XML2PACValue
- Out_XML2IsNRRequested

The Application Header (AppHdr) overrides are as follows:
- Out_AppHdrFromType
- Out_AppHdrFromID
- Out_AppHdrToType
- Out_AppHdrToID
- Out_AppHdrSvcName
- Out_AppHdrMsgName
- Out_AppHdrDupRef
- Out_AppHdrDupInfo

Chapter 3. SWIFT and SWIFTNet Business Processes

To help you accomplish your business goals, the Sterling B2B Integrator provides two predefined enveloping business processes, which are used by the Sterling B2B Integrator to implement SWIFT and SWIFTNet processing: SWIFTEnvelope and SWIFTDeenvelope. These predefined business processes are initiated by other processes/services during SWIFT processing and do not require you to modify them.

Sterling B2B Integrator also provides predefined business processes, which are used by the Sterling B2B Integrator to implement SWIFT transport, including the SWIFTNetClient business process (using for InterAct transport), and the SWIFTNet ClientFA (used for FileAct transport) business process. These predefined business processes are initiated by other processes/services during SWIFT processing but do require you to modify them.

The following table lists business goals for some of the predefined SWIFT business processes (please note that this is not an inclusive list of all predefined business process):

Business Process For SWIFTNet Version 6.x	Corresponding Business Process for SWIFTNet7 (if applicable)	Business Goals
SWIFTDeenvelope	SWIFTDeenvelope	Extracts SWIFT message types from a message and translates and processes them, according to the content of the envelopes.
SWIFTEnvelope	SWIFTEnvelope	Applies a SWIFT envelope to one or more SWIFT message types and then uses the envelope data to translate and process them.
SWIFTNetClient	SWIFTNet7Client	Used for InterAct processing. Contains the necessary parameters so the SWIFTNet Client or SWIFTNet7 Client service can prepare the request and send it (outbound) to the SWIFTNet Server. The client application on the SWIFTNet Server processes this request, performs the necessary communication exchange with the SWIFTNet SAG/SNL instance, and sends the request to the SWIFTNet Network.

Business Process For SWIFTNet Version 6.x	Corresponding Business Process for SWIFTNet7 (if applicable)	Business Goals
SWIFTNetClientFA	SWIFTNet7Client	Used for FileAct processing. Contains the necessary parameters so the SWIFTNet Client or SWIFTNet7 Client service can prepare the request and send it (outbound) to the SWIFTNet Server. The client application on the SWIFTNet Server processes this request, performs the necessary communication exchange with the SWIFTNet SAG/SNL instance, and sends the request to the SWIFTNet Network.
handleSWIFTNetServerRequest	handleSWIFTNet7InterActRequest	Used for InterAct processing. Enables the Sterling B2B Integrator to receive SWIFTNet messages. This is the bootstrap business process used by the inbound SWIFTNet request through the SWIFTNet Server. It is a system business process used by the SWIFTNet Server Adapter, which pre-processes the incoming request, searches the SWIFTNet Routing Rule table, and routes the request payload to the business process for processing.
handleSWIFTNetServerSnFRequest	handleSWIFTNet7InterActSnFRequest	Used for InterAct processing. Enables the Sterling B2B Integrator to receive SWIFTNet store-and-forward messages. This is a bootstrap business process used by the inbound SWIFTNet request that includes a store-and-forward option. For the store-and-forward option, an incoming request is not processed immediately, but instead is stored in the responder mailbox in the Sterling B2B Integrator to be responded to later. The Sterling B2B Integrator then sends an acknowledgement that the request has been successfully stored to the requestor through the SWIFTNet Server.

Business Process For SWIFTNet Version 6.x	Corresponding Business Process for SWIFTNet7 (if applicable)	Business Goals
handleSWIFTNetServerFARequest	handleSWIFTNet7FileActRequest	Used for FileAct processing. Enables the Sterling B2B Integrator to receive SWIFTNet messages. This is the bootstrap business process used by the inbound SWIFTNet request through the SWIFTNet Server. It is a system business process used by the SWIFTNet Server Adapter, which pre-processes the incoming request, searches the SWIFTNet Routing Rule table, and routes the request payload to the business process for processing.
handleSWIFTNetServerFASnFRequest	handleSWIFTNet7FileActSnFRequest	Used for FileAct processing. Enables the Sterling B2B Integrator to receive SWIFTNet store-and-forward messages. This is a bootstrap business process used by the inbound SWIFTNet request that includes a store-and-forward option. For the store-and-forward option, an incoming request is not processed immediately, but instead is stored in the responder mailbox in the Sterling B2B Integrator to be responded to later. The Sterling B2B Integrator then sends an acknowledgement that the request has been successfully stored to the requestor through the SWIFTNet Server.
handleSWIFTNetServerFASnFEvent	handleSWIFTNet7FileActEvent	The handleSWIFTNetServerFASnFEvent or handleSWIFTNet7FileActEvent business process is used by the SWIFTNet Server. It is a system business process called by the SWIFTNet Server Adapter or SWIFTNet7 Adapter that preprocesses the incoming FileAct Event with a COMPLETED status, searches the SWIFTNet routing rule table and bootstraps the business process for processing

Business Process For SWIFTNet Version 6.x	Corresponding Business Process for SWIFTNet7 (if applicable)	Business Goals
SWIFTMessageEntryOutbound	SWIFTMessageEntryOutbound	This is the default business process used to look up the correct envelope, based on information provided by the user while configuring the Send information in the SWIFT Message Entry Workstation. This business process uses the EDI Encoder service, Envelope service, and then invokes the business process specified in the SWIFT Outbound Envelope.

Writing a Business Process for a StandardsTranslation Operation

Assigning the SWIFT_ValidationFlag as a parameter to the StandardsTranslation operation within a business process does not perform the expected validation, and the validation is not performed by the translator because the workflow context returns the parameter as a string object instead of a NodeList. To define the SWIFT_ValidationFlag parameter, the assign statement should be made in process data outside of the StandardsTranslation operation. For example:

```
<process name="TEST_2008_574IRSLST">
  <sequence>
    <assign to="SWIFT_ValidationFlag">IRSLST</assign>
    <operation name="StandardsTranslation">
      <participant name="StandardsTranslation"/>
      <output message="Xout">
        <assign to="." from="*"></assign>
        <assign to="map_name">SWIFT_2012_574_IRSLST</assign>
      </output>
      <input message="Xin">
        <assign to="." from="*"></assign>
      </input>
    </operation>
  </sequence>
</process>
```

SWIFTDeenvelope Business Process

Before Using the SWIFTDeenvelope Business Process

Before you use the SWIFTDeenvelope business process, you must create a SWIFT inbound envelope for each SWIFT message type that you are receiving. See "Inbound SWIFT Envelope" on page 21.

SWIFTEnvelope Business Process

The SWIFTEnvelope business process is initiated when it is called by another business process. The SWIFTEnvelope business process envelopes the SWIFT messages contained in the business process context with outbound SWIFT envelopes that you have preconfigured. You must have created one outbound SWIFT envelope for each SWIFT message type that you are sending.

A typical scenario is one in which SWIFT data must be sent to a trading partner. To prepare for this, the data must be enveloped to provide identifying batch and interchange data. The SWIFTEnvelope business process helps to provide these enveloping services.

The SWIFTEnvelope business process is initiated as part of the following outbound process flow:

1. You create a business process that calls the EDI Encoder service or Document Extraction service.
2. The EDI Encoder service or Document Extraction service looks up the envelope to apply and prepares the document to be enveloped.
3. Either the business process calls the EDI Enveloping service or the Document Extraction service is configured to perform enveloping.
4. The EDI Enveloping service or Document Extraction service starts the SWIFTEnvelope business process (which runs the Generic Envelope service as a subprocess to extract the name of the business process).
5. The SWIFTEnvelope business process searches the envelope definition to retrieve information to envelope each message.

   Note: The Sender ID, Receiver ID, and Lookup Alias in your Outbound Envelope definition must match the parameters that you define for this outbound business process.

Before Using the SWIFTEnvelope Business Process
About this task

Before you use the SWIFTEnvelope business process, you must complete the following task:

Procedure

Create a SWIFT outbound envelope for each SWIFT message type that you are sending. See "Outbound SWIFT Envelope" on page 37.

SWIFTNetClient Business Process

The SWIFTNet Client service (for SWIFTNet 6.x) enables you to use Secure Sockets Layers (SSL), but to do so you must also upgrade the SWIFTNetClient business process if you have not already done so. The upgrade BPML differs based on whether you are using InterAct or FileAct. See the *SWIFTNet Client Service* documentation for more information on how to upgrade the SWIFTNetClient business process if you are using SSL.

Note: If you previously installed an earlier version of the Standards Library, you do not need to upgrade the SWIFTNetClient business process again. However, you will need to reinstall the SWIFTNet Server (see Chapter 7, "SWIFTNet Server," on page 113 for more information).

The SWIFTNetClient business process contains the necessary parameters so the SWIFTNet Client service can prepare the request and send it to the SWIFTNet Server. The client application on the SWIFTNet Server processes this request, performs the necessary communication exchange with the SWIFTNet SAG/SNL instance, and sends the request to the SWIFTNet Network.

Note: This business process is used for InterAct processing only. It takes the default parameters configured in the SWIFTNet Client service. If a parameter defined in the SWIFTNet Client Service is specified in this business process, the value in the business process overrides the parameter in the SWIFTNet Client service.

The SWIFTNetClient business process is initiated as part of the following outbound process flow:

1. The SWIFTNetClient business process invokes the SWIFTNet Client service and passes it all the necessary parameters to send a request.
2. The client application on the SWIFTNet Server processes the request.

Before Using the SWIFTNetClient Business Process
About this task

Before you use the SWIFTNetClient business process, you must complete the following tasks:

Procedure
1. Configure the SWIFTNet Client service (or create a new instance of it) to reflect your installation. See *SWIFTNet Client Service*.
2. Enable Document Tracking in the Business Process Manager.

   Then you can execute the SWIFTNetClient business process as part of your SWIFTNet processing.

SWIFTNet7Client Business Process

The SWIFTNet7 Client service (for SWIFTNet 7.x) enables you to use Secure Sockets Layers (SSL), but to do so you must also upgrade the SWIFTNet7Client business process if you have not already done so. The upgrade BPML differs based on whether you are using InterAct or FileAct. See the *SWIFTNet7 Client Service* documentation for more information on how to upgrade the SWIFTNet7Client business process if you are using SSL.

Note: If you previously installed an earlier version of the Standards Library, you do not need to upgrade the SWIFTNet7Client business process again. However, you will need to reinstall the SWIFTNet7 Servers.

The SWIFTNet7Client business process contains the necessary parameters so the SWIFTNet7Client service can prepare the request and send it to the SWIFTNet Servers. The client application on the SWIFTNet Servers processes this request, performs the necessary communication exchange with the SWIFTNet SAG/SNL instance, and sends the request to the SWIFTNet Network.

The SWIFTNet7Client business process is initiated as part of the following outbound process flow:

1. The SWIFTNet7Client business process invokes the SWIFTNet7Client service and passes it all the necessary parameters to send a request.
2. The client application on the SWIFTNet Servers processes the request.

Before Using the SWIFTNet7Client Business Process
About this task

Before you use the SWIFTNet7Client business process, you must complete the following tasks:

Procedure

1. Configure the SWIFTNet7 Client service (or create a new instance of it) to reflect your installation. See *SWIFTNet7 Client Service*.
2. Enable Document Tracking in the Business Process Manager.

   Then you can execute the SWIFTNet7Client business process as part of your SWIFTNet processing.

SWIFTNetClientFA Business Process

This is a client business process to send requests to SWIFTNet using FileAct.

Note: This business process is used with FileAct processing only. It takes the default parameters configured in the SWIFTNet Client service. If a parameter defined in the SWIFTNet Client Service is specified in this business process, the value in the business process overrides the parameter in the SWIFTNet Client service.

The SWIFTNetClientFA business process is initiated as part of the following outbound process flow:

1. The SWIFTNetClientFA business process invokes the SWIFTNet Client service and passes it all the necessary parameters to send a request.
2. The client application on the SWIFTNet Server processes the request.

Before Using the SWIFTNetClientFA Business Process
About this task

Before you use the SWIFTNetClientFA business process, you must complete the following tasks:

Procedure

1. Configure the SWIFTNet Client service (or create a new instance of it) to reflect your installation. See *SWIFTNet Client Service*.
2. Enable Document Tracking in the Business Process Manager.

   Then you can execute the SWIFTNetClientFA business process as part of your SWIFTNet processing.

SWIFTNet7ClientFA Business Process

This is a client business process to send requests to SWIFTNet using FileAct.

The SWIFTNet7ClientFA business process is initiated as part of the following outbound process flow:

1. The SWIFTNet7ClientFA business process invokes the SWIFTNet7 Client service and passes it all the necessary parameters to send a request.
2. The client application on the SWIFTNet Servers process the request.

Before Using the SWIFTNet7ClientFA Business Process
About this task

Before you use the SWIFTNet7ClientFA business process, you must complete the following tasks:

Procedure

1. Configure the SWIFTNet7 Client service (or create a new instance of it) to reflect your installation. See *SWIFTNet7 Client Service*.
2. Enable Document Tracking in the Business Process Manager.

   Then you can execute the SWIFTNet7ClientFA business process as part of your SWIFTNet processing.

handleSWIFTNetServerRequest and handleSWIFTNet7InterActRequest Business Processes

The handleSWIFTNetServerRequest business process (handleSWIFTNet7InterActRequest business process for SWIFTNet7) enable the Sterling B2B Integrator to receive SWIFTNet messages. This is the bootstrap business process used by the inbound SWIFTNet request through the SWIFTNet Server. It is a system business process used by the SWIFTNet Server Adapter or SWIFTNet7 Adapter, which pre-processes the incoming request, search the SWIFTNet Routing Rule table, executes the matching routing rule, and route the request payload to the business process for processing.

Note: These business processes are used with InterAct processing only.

The handleSWIFTNetServerRequest or handleSWIFTNet7InterActRequest business process is initiated as part of the following inbound process flow:

1. The SWIFTNet Server Adapter or SWIFTNet7 Adapter invokes the handleSWIFTNetServerRequest or handleSWIFTNet7InterActRequest business process and passes it all the necessary parameters to send a request.
2. The handleSWIFTNetServerRequest or handleSWIFTNet7InterActRequest business process invokes the SOAP Inbound service.
3. The SOAP Inbound service:
   - Pre-processes the incoming request.
   - Searches the SWIFTNet Routing Rule table.
   - Routes the request payload to the business process for processing.
4. When a message is received, the message is parsed and then the routing rule will be evaluated.
   - If Action = Store to Mailbox, the Routing Rule calls the Add Mailbox service. If the mailbox does not exist, it will be automatically created.
   - If Action = Store to File System, the Routing Rule calls the File System Adapter. If the directory does not exist, it will be automatically created.
   - If Action = Invoke BP, the specified business process is invoked.
5. The business process also evaluates the routing rule and since the responder need to respond immediately, the system will (if necessary) retrieve the Response Template File. This file becomes the response payload. The Response Template File can be configured inside the Routing Rule as well.
6. Then the SOAP Outbound service invokes the HTTP Response service.
7. The HTTP Response service sends a response to the request.

Before Using the handleSWIFTNetServerRequest or handleSWIFTNet7InterActRequest Business Process

About this task

Before you use the handleSWIFTNetServerSnFRequest or handleSWIFTNet7InterActSnFRequest business process, you must complete the following tasks:

Procedure

1. Configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to reflect your installation. See *SWIFTNet Server Adapter* or *SWIFTNet7 Adapter*.
2. Enable Document Tracking in the Business Process Manager for the handleSWIFTNetServerRequest or handleSWIFTNet7InterActRequest business process.

handleSWIFTNetServerSnFRequest and handleSWIFTNet7InterActSnFRequest Business Processes

The handleSWIFTNetServerSnFRequest or handleSWIFTNet7InterActSnFRequest business process enable the Sterling B2B Integrator to receive SWIFTNet store-and-forward messages. This is a bootstrap business process used by the inbound SWIFTNet request that includes a store-and-forward option. For Y Copy Services (regardless of InterAct or FileAct), an incoming request is not processed immediately, but instead is stored in the responder mailbox in the Sterling B2B Integrator to be responded to later. The Sterling B2B Integrator then sends an acknowledgement that the request has been successfully stored to the requestor through the SWIFTNet Server.

When a message is received, the message is parsed and then the routing rule will be evaluated.

- If Action = Store to Mailbox, the Routing Rule calls the Add Mailbox service. If the mailbox does not exist, it will be automatically created.
- If Action = Store to File System, the Routing Rule calls the File System Adapter. If the directory does not exist, it will be automatically created.
- If Action = Invoke BP, the specified business process is invoked

Since the responder need to respond immediately, the system will (if necessary) retrieve the Response Template File. This file is the response payload. The Response Template File can be configured inside the Routing Rule as well. For Store and Forward, acknowledgment (positive or negative) will be sent.

Note: These business processes are used with InterAct processing only.

The handleSWIFTNetServerSnFRequest or handleSWIFTNet7InterActSnFRequest business process is initiated as part of the following inbound process flow:

1. The SWIFTNet Server Adapter or SWIFTNet7 Adapter invokes the handleSWIFTNetServerSnFRequest or handleSWIFTNet7InterActSnFRequest business process and passes it all the necessary parameters to send a request.
2. The handleSWIFTNetServerSnFRequest or handleSWIFTNet7InterActSnFRequest business process invokes the SOAP Inbound service.
3. The SOAP Inbound service:

- Pre-processes the incoming request
- Searches the SWIFTNet Routing Rule table.

4. The Mailbox Add service Routes the request payload to the responder mailbox to be responded to later.

5. Then the SOAP Outbound service invokes the HTTP Response service.

6. The HTTP Response service sends an acknowledgement to the requestor that the request has been successfully stored.

Before Using the handleSWIFTNetServerSnFRequest or handleSWIFTNet7InterActSnFRequest Business Process

About this task

Before you use the handleSWIFTNetServerSnFRequest or handleSWIFTNet7InterActSnFRequest business process, you must complete the following tasks:

Procedure

1. Configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to reflect your installation. See *SWIFTNet Server Adapter* or *SWIFTNet7 Adapter*.

2. Enable Document Tracking in the Business Process Manager for the handleSWIFTNetServerSnFRequest or handleSWIFTNet7InterActSnFRequest business process.

3. For Y Copy Services only, create a Responder Mailbox (/Responder DN) and Requestor Mailbox (/Responder DN/Requestor DN) where the request payload can be saved.

handleSWIFTNetServerFARequest and handleSWIFTNet7FileActRequest Business Processes

The handleSWIFTNetServerFARequest or handleSWIFTNet7FileActRequest business process enables the Sterling B2B Integrator to receive SWIFTNet messages. This is the bootstrap business process used by the inbound SWIFTNet request through the SWIFTNet Server. It is a system business process used by the SWIFTNet Server Adapter or SWIFTNet7 Adapter, which pre-processes the incoming request, search the SWIFTNet Routing Rule table, executes the matching routing rule, and route the request payload to the business process for processing.

Note: These business processes are used with FileAct processing only.

The handleSWIFTNetServerFARequest or handleSWIFTNet7FileActRequest business process is initiated as part of the following inbound process flow:

1. The SWIFTNet Server Adapter or SWIFTNet7 Adapter invokes the handleSWIFTNetServerFARequest or handleSWIFTNet7FileActRequest business process and passes it all the necessary parameters to send a request.

2. The handleSWIFTNetServerFARequest or handleSWIFTNet7FileActRequest business process invokes the SOAP Inbound service.

3. The SOAP Inbound service:
   - Pre-processes the incoming request.
   - Searches the SWIFTNet Routing Rule table.
   - Routes the request payload to the business process for processing.

4. Then the SOAP Outbound service invokes the HTTP Response service.

5. The HTTP Response service sends a response to the request.

Before Using the handleSWIFTNetServerFARequest or handleSWIFTNet7FileActRequest Business Process

About this task

Before you use the handleSWIFTNetServerFARequest or handleSWIFTNet7FileActRequest business process, you must complete the following tasks:

Procedure

1. Configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to reflect your installation. See *SWIFTNet Server Adapter* or *SWIFTNet7 Adapter*.
2. Enable Document Tracking in the Business Process Manager for the handleSWIFTNetServerFARequest or handleSWIFTNet7FileActRequest business process.

handleSWIFTNetServerFASnFRequest and handleSWIFTNet7FileActSnFRequest Business Processes

The handleSWIFTNetServerFASnFRequest or handleSWIFTNet7FileActSnFRequest business process enable the Sterling B2B Integrator to receive SWIFTNet store-and-forward messages. This is a bootstrap business process used by the inbound SWIFTNet request that includes a store-and-forward option. For Y Copy Services (regardless of InterAct or FileAct), an incoming request is not processed immediately, but instead is stored in the responder mailbox to be responded to later. Then Sterling B2B Integrator an acknowledgement that the request has been successfully stored to the requestor through the SWIFTNet Server.

Note: These business processes are used for FileAct processing only.

The handleSWIFTNetServerFASnFRequest or handleSWIFTNet7FileActSnFRequest business process is initiated as part of the following inbound process flow:

1. The SWIFTNet Server Adapter or SWIFTNet7 Adapter invokes the handleSWIFTNetServerFASnFRequest or handleSWIFTNet7FileActSnFRequest business process and passes it all the necessary parameters to send a request.
2. The handleSWIFTNetServerFASnFRequest or handleSWIFTNet7FileActSnFRequest business process invokes the SOAP Inbound service.
3. The SOAP Inbound service:
   - Pre-processes the incoming request.
   - Searches the SWIFTNet Routing Rule table.
4. The Mailbox Add service Routes the request payload to the responder mailbox to be responded to later.
5. Then the SOAP Outbound service invokes the HTTP Response service.
6. The HTTP Response service sends an acknowledgement to the requestor that the request has been successfully stored.

Before Using the handleSWIFTNetServerFASnFRequest or handleSWIFTNetServerFASnFRequest Business Process

About this task

Before you use the handleSWIFTNetServerFASnFRequest or handleSWIFTNet7FileActSnFRequest business process, you must complete the following tasks:

Procedure

1. Configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to reflect your installation. See *SWIFTNet Server Adapter* or *SWIFTNet7 Adapter*.
2. Enable Document Tracking in the Business Process Manager for the handleSWIFTNetServerFASnFRequest or handleSWIFTNet7FileActSnFRequest business process.

handleSWIFTNetServerFASnFEvent and handleSWIFTNet7FileActEvent Business Processes

The handleSWIFTNetServerFASnFEvent or handleSWIFTNet7FileActEvent business process is used by the SWIFTNet Server. It is a system business process called by the SWIFTNet Server Adapter or SWIFTNet7 Adapter that preprocesses the incoming FileAct Event with a COMPLETED status, searches the SWIFTNet routing rule table and bootstraps the business process for processing.

When a message is received, the message is parsed and then the routing rule will be evaluated.

- If Action = Store to Mailbox, the Routing Rule calls the Add Mailbox service. If the mailbox does not exist, it will be automatically created.
- If Action = Store to File System, the Routing Rule calls the File System Adapter. If the directory does not exist, it will be automatically created.
- If Action = Invoke BP, the specified business process is invoked.
-

Note: These business processes are used for FileAct processing only.

The handleSWIFTNetServerFASnFEvent or handleSWIFTNet7FileActEvent business process is initiated as part of the following inbound process flow:

1. The SWIFTNet Server Adapter or SWIFTNet7 Adapter invokes the handleSWIFTNetServerFASnFEvent or handleSWIFTNet7FileActEvent business process and passes it all the necessary parameters to send a request.
2. The handleSWIFTNetServerFASnFEvent or handleSWIFTNet7FileActEvent business process preprocesses the incoming FileAct event.
3. The handleSWIFTNetServerFASnFEvent or handleSWIFTNet7FileActEvent business process searches the SWIFTNet Routing Rule table.
4. Then the business process bootstraps the appropriate business process for processing.

Before Using the handleSWIFTNetServerFASnFEvent or handleSWIFTNet7FileActEvent Business Process

About this task

Before you use the handleSWIFTNetServerFASnFEvent or handleSWIFTNet7FileActEvent business process, you must complete the following tasks:

Procedure

1. Configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to reflect your installation. See *SWIFTNet Server Adapter* or *SWIFTNet7 Adapter*.
2. Enable Document Tracking in the Business Process Manager for the handleSWIFTNetServerFASnFEvent or handleSWIFTNet7FileActEvent business process.

SWIFTMessageEntryOutbound

The SWIFTMessageEntryOutbound business process is the default business process used to look up the correct envelope, based on the information provided by the user while configuring the Send information in the SWIFT Message Entry Workstation.

The SWIFTMessageEntryOutbound business process is initiated as part of the following outbound process flow:

1. Create a SWIFT message using the SWIFT Message Entry Workstation.
2. Validate the message and mark it **Ready to Send**.
3. Do not change the default business process and send the SWIFT message you created.
4. The SWIFTMessageEntryOutbound business process invokes the EDI Encoder service.
5. The EDI Encoder service includes the Generic Envelope service.
6. The Generic Envelope service invokes the business process configured in the SWIFT Outbound Envelope to send the SWIFT message.

Before Using the SWIFTMessageEntryOutbound Business Process

Before you use the SWIFTMessageEntryOutbound business process, you must enable Document Tracking in the Business Process Manager for the SWIFTMessageEntryOutbound business process.

Chapter 4. Creating SWIFT Maps

The Sterling B2B Integrator Map Editor enables you to map SWIFT documents. If WebSphere Transformation Extender is installed and integrated with Sterling B2B Integrator, you also have the option of using WebSphere Transformation Extender to create SWIFT maps. For details on creating maps with WebSphere Transformation Extender see the *WebSphere Transformation Extender Information Center and Pack for SWIFT*.

The Sterling B2B Integrator Map Editor generates a file layout for you using the components and message types that you select. The Sterling B2B Integrator Map Editor-generated SWIFT map consists of groups, records, composites, and fields that are comparable to parameters that are defined by SWIFT. See "How SWIFT Terminology Correlates with the Sterling B2B Integrator Map Editor" on page 82.

Note: You need to download the Sterling B2B Integrator Map Editor component from Sterling B2B Integrator to use it.

Sterling B2B Integrator Map Editor allows you to modify the map components by using the Deactivate, Promote, Split, Copy, Cut, and Paste functions.

You can create a map for all SWIFT Standards Release messages loaded into the standards database. All messages are validated by the Sterling B2B Integrator for syntax (for example, field types, field lengths, and so forth). See Chapter 1, "SWIFT and SWIFTNet with Sterling B2B Integrator," on page 1 for information on which messages are validated by the Sterling B2B Integrator for syntax *and* semantics. If you create a map for a message that is not currently validated by the Sterling B2B Integrator, when the Sterling B2B Integrator Map Editor is capable of updating your existing map with message validation rules in a future release, any extended rules you have created will exist outside the validation block (and thus will always run regardless of whether validation is enabled).

When you create a map, for both the input and output sides, you choose whether you want to use a preloaded standard, an existing file format, or wants to create a new file format for that side of the map (including selecting from standards that you have previously downloaded to the standards database. The preloaded standards are automatically loaded when you download the Sterling B2B Integrator Map Editor to your machine. You must install the SWIFT Financial Services XML Standards to be able to select SWIFTNet Bulk Payments, SWIFTNet Transaction Reporting, SWIFTNet Funds, SWIFTNet Trade Services, SWIFTNet Exceptions and Investigations, SWIFTNet Proxy Voting, SWIFTNet SCORE, SWIFTNet FpML, SWIFTNet Cash Management, SWIFTNet Cash Reporting, SWIFTNet Corporate Actions, SWIFTNet Clearing, or ISO 20022.

For version 8.0 and forward, all SWIFT and SWIFTNet maps have group parsing shortcuts enabled for the Input side of the map during translation to skip group parsing if the first non-repeating mandatory record of a group is not present in the input.

However, if you want to use a specialized version of a message that is not available in the Sterling B2B Integrator SWIFT data dictionary, it may be appropriate for you to define the SWIFT map yourself. Alternatively, you could use an MT or MX that is similar to what is required and customize it yourself.

The Sterling B2B Integrator also performs SWIFT MX validations. Validations are defined in the **translator_swift_mx.properties** property file and generated maps define the parameters used to look up validates in this property file. Changes to validations are also applied to SWIFT MX messages for currencies, BICPlusIBAN, BIC/BEI addresses, and International Bank Account Number checksum validation (IBAN). You can define code lists for currencies, countries or regions, BIC, or BEI addresses (that are validated against the SWIFT_Addresses code list) BICPlusIBAN, and SEPARouting. IBAN data contains a country or region code that is validated against the SWIFT_Countries code list (this SWIFT code list contains countries and regions) in the Sterling B2B Integrator, and additional IBAN validation is handled internally by the translator.

The Sterling B2B Integrator supplies you with the following statements to be used with SWIFT maps and any other data format:

- cerror
- exist
- occurrencetotal
- resetoccurrencetotal
- sum
- sumtotal

See *Alphabetic Language Reference* for all the available extended rules.

To ensure that extended rules for SWIFT maps are only run when validation is enabled, place your custom extended rule code in a validation block. See "Creating Extended Rules for SWIFT Maps" on page 94.

How SWIFT Terminology Correlates with the Sterling B2B Integrator Map Editor

The terminology used by SWIFT differs from that used in the Sterling B2B Integrator Map Editor. This table lists the SWIFT terms and how they correspond to map components in the Sterling B2B Integrator Map Editor:

SWIFT Terminology
> **Corresponding Sterling B2B Integrator Map Editor Terminology**

MT or MX (File)
> SWIFT file format (the top map component on the input and output sides of a map)

Sequence or an implicit group of repeating field tags
> Group

Field tag
> SWIFT Record

Subfield that is an "OR" option or a group of related subfields that occur in a sequence, and they are also groups of related subfields that occur in a sequence (for example, each SWIFTBIC address is defined as a "composite" consisting of a number of subfields such as branch code, location, and so forth)
> Composite

Subfield
> Field

SWIFT Components in the Sterling B2B Integrator Map Editor

The following table lists the components that make up the SWIFT layout in the Sterling B2B Integrator Map Editor, the icons that represent the components, and descriptions of the components.

Component	Icon	Description
SWIFT root element		The *SWIFT root element* represents the MT or MX that the Sterling B2B Integrator is mapping. At the SWIFT file root element, you define the message type and encoding. It is a group and can contain groups and SWIFT records.
Group		A *group* is a looping structure that contains a sequence or an implicit group of repeating field tags (in Sterling B2B Integrator Map Editor a group is related records and groups that repeat in sequence until either the group data ends, or the maximum number of times that the loop is permitted to repeat is exhausted). A group that is subordinate to another group is a subgroup (and corresponds to a nested looping structure, a loop within a loop). When a group contains an extended rule or a standard rule, an asterisk appears to the right of the group icon.
SWIFT Record		A *SWIFT record* contains a field tag (in Sterling B2B Integrator Map Editor a SWIFT record is a group of related fields or composite data elements that combine to communicate useful data). A SWIFT record can occur once or can repeat multiple times. **Note:** If a SWIFT record occurs more than once in a map, it is identified by its name <ID>. The second and subsequent occurrences are identified by <ID>:*n*, where *n* is the number of the occurrence in the map.
Composite		A *composite* is a subfield that is an "OR" option or a group of related subfields that occur in a sequence (in Sterling B2B Integrator Map Editor a composite is a data element that contains two or more component data elements or subelements). They are also groups of related subfields that occur in a sequence. For example, each SWIFTBIC address is defined as a "composite" consisting of a number of subfields (e.g. branch code, location). A composite can occur once or repeat multiple times. For example, each SWIFTBIC address is defined in the Sterling B2B Integrator Map Editor as a composite that consists of a number of fields (SWIFT subfields) such as branch code, location, and so forth. **Note:** If a composite occurs more than once in a map, it is identified by its name <ID>. The second and subsequent occurrences are identified by <ID>:*n*, where *n* is the number of the occurrence in the map. A *repeating composite* is a related group of fields that have the ability to loop as a whole (occur more than once) within a particular SWIFT record. To enable a composite to repeat multiple times within a SWIFT record, each occurrence of the composite must be separated by a start and end delimiter. **Note:** SWIFT does not use the repeating option, although it is available to you.

Component	Icon	Description
Field		A *field* is a subfield or a group of SWIFT components (the smallest piece of information defined by the SWIFT standard) that define a SWIFT subfield. A field can have different meanings depending on the context. In other data formats in the Sterling B2B Integrator Map Editor, a field is not considered to have useful meaning except in the larger context of the record that contains it. However, fields used in the Sterling B2B Integrator Map Editor to represent SWIFT subfields and components contain useful and discrete information. **Note:** If a field occurs more than once in a map it is identified by its name <ID>. The second and subsequent occurrences are identified by <ID>:*n*, where *n* is the number of the occurrence in the map. A *repeating field* is a field with the ability to loop (occur more than once) within a particular SWIFT record. To enable a single field to repeat multiple times within a SWIFT record, each occurrence of the field must be separated by a start and end delimiter. The use of start and end delimiters help the translator determine where subfields and components are defined within a field tag. When a field has a link performed against it, a red check mark appears over the field icon. When a field contains an extended rule or a standard rule, an asterisk appears to the right of the field icon.

Download and Install the Sterling Standards Database for SWIFT

About this task

Before you install the Sterling Standards Database for SWIFT on your desktop, consider these guidelines:

- Download the Sterling B2B Integrator Map Editor.
- For the most current version of the SWIFT standard, contact IBM Customer Support.
- Be sure your desktop meets the Windows Client requirements listed in the *System Requirements*.

To download and install the Sterling Standards Database for SWIFT:

Procedure

1. From the Sterling B2B Integrator **Deployment** menu, select **Standards**.

   Note: You must download and install both the Sterling B2B Integrator Map Editor and the desired Sterling Standards Database for SWIFT from the *same* Sterling B2B Integrator release.

2. In the Download and Install section next to **Download SWIFT Standards**, click **Go!**

3. Click **Save** to save the executable file and then double-click the file and click **Run** to start the install.

4. You are prompted to choose the setup language (that is, the language in which the installation instructions will be given). Select the setup language and click **Next**.

   Note: The Standards installation program is available in the following languages: English, Français (France), Deutsche (Germany), Italiano (Italy),

Japanese (Japan), Korean (Korea), Português (Brazil), Español (Spain), Simplified Chinese (China), and Traditional Chinese (Taiwan).

5. In the Welcome window, click **Next**.

6. In the Choose Destination Location window, select where you want to install the standards database:

   - If you accept the default location, click **Next**.
   - If you want to specify a different location, click **Browse**, specify the path to the folder, click **OK**, and click **Next**.

     If you specify a folder name that does not exist, the installation program displays a message asking if you want to create that folder.

7. In the Select Components window, verify that **SWIFTStandardDatabase**, **SWIFTMarketPractice**, and **ExtendedRuleLibrary** are selected and click **Next**.

   The download wizard installs the Sterling Standards Database for SWIFT database.

8. In the Setup Complete window, click **Finish**.

Create a File Layout from an MT or Market Practice

About this task

When you create a new map, you typically use the map wizard that creates a layout for you based on an MT or Market Practice from the standards database. The wizard saves you the time and effort to create the SWIFT side of the map yourself, and minimizes the risk of having an invalid standard format for a message.

When you create a map, for both the input and output sides, you choose whether you want to use a preloaded standard, an existing file format, or wants to create a new file format for that side of the map (including selecting from standards that you have previously downloaded to the standards database.

Note: You can create a map for all SWIFT messages through the Sterling B2B Integrator Map Editor, and all messages are validated by the Sterling B2B Integrator for syntax (that is, field types, field lengths, and so forth).

To create a file layout from an MT, MX, or Market Practice:

Procedure

1. From the Sterling B2B Integrator Map Editor **File** menu, select **New**.

2. In the **New Map Wizard**, complete the questions in the first window and click **Next**.

   Note: Be sure that Sterling B2B Integrator is selected in the **What type of map are you creating** list.

3. If you are translating from SWIFT, in the Input Format window, Create a new data format using this standards, select **SWIFT (Society for Worldwide Interbank Financial Telecommunications)** and click **Messages**.

   Note: The **Create a new data format using this** selection allows you to use a preloaded standard (that is, you do not have to download the standard to the Sterling B2B Integrator database) or use the standards that were downloaded through the standards database.

4. Click **Next**.

5. Select the type of message to auto-generate:
   * Standard SWIFT Message Type
   * Market Practice
   * Custom Market Practice

   Note: If you select custom Market Practice, you must have already created a custom Market Practice to have it listed as an available option.
6. If you are translating an MT, select the **SI SWIFT Standard** ODBC data source (which contains the Sterling Standards Database for SWIFT) and click **Next**.
7. If you are translating a Market Practice, select the **SI SWIFTMP Customer** ODBC data source (which contains the SWIFT preloaded market practices) and click **Next**.
8. If you are translating from an MT, select the MT that you want to use and select the version, and click **Next**.
9. If you are translating from a preloaded Market Practice, select the MT you want to modify and click **Next**.
10. If you are translating from SWIFT, click **Finish** to load the MT you selected.
11. Click **Next**.
12. If you are translating to SWIFT, in the Output Format window, select **SWIFT (Society for Worldwide Interbank Financial Telecommunications)** and click **Customize**. If you are translating from another format, select that format and continue to the next screen.
13. Click **Next**.
14. If you are translating to SWIFT, select the **SWIFT Standard** ODBC data source (which contains the Sterling Standards Database for SWIFT) and click **Next**.
15. If you are translating to SWIFT, select the MT that you want to use and select the version, and click **Next**.
16. If you are translating to SWIFT, click **Finish** to load the MT you selected.
17. Click **Next**.
18. If you are translating to SWIFT, click **Finish**. The Sterling B2B Integrator Map Editor displays the new map in the Sterling B2B Integrator Map Editor window.
19. Use Autolink to link your fields prior to using the Link function or creating standard or extended rules.
20. Use the Rule Library Manager to add a reusable extended rule library that can then be added to any map so you do not have to redefine extended rules for each map you create.

   Note: The SWIFT extended rules libraries that are installed with the Sterling B2B Integrator, contain all the extended rules necessary to carry out the business logic for SWIFT messages. Extended Rule Libraries from 2008 and forward are delivered in the SWIFT Standards Install.

Create a Custom Market Practice or Fund Message

About this task

You have the capability to create a custom market practice or fund message by using the Sterling B2B Integrator Map Editor to create a message of the same message type you want, modifying it accordingly, and using the SWIFT Rule Importer to import the map into the Sterling B2B Integrator.

To create a custom Market Practice or Fund message:

Procedure

1. From the Sterling B2B Integrator Map Editor **File** menu, select **New**.
2. In the **New Map Wizard**, complete the questions in the first window and click **Next**.

   Note: Be sure that Sterling B2B Integrator is selected in the **What type of map are you creating** list.
3. In the Input Format window, Create a new data format using this standards, select **SWIFT (Society for Worldwide Interbank Financial Telecommunications)** and click **Messages**.
4. Click **Next**.
5. Select **Standard SWIFT Message Type** as the type of message to auto-generate.
6. Select the **SI SWIFT Standard** ODBC data source (which contains the Sterling Standards Database for SWIFT) and click **Next**.
7. Select the MT that you want to use and select the version, and click **Next**.
8. If you are translating from SWIFT and have selected a common group MT, select the MT for copy fields and click **Next**.
9. If you are translating from SWIFT, click **Finish** to load the MT you selected.
10. Click **Next**.
11. In the Output Format window, select **SWIFT (Society for Worldwide Interbank Financial Telecommunications)** and click **Customize**. If you are translating from another format, select that format and continue to the next screen.
12. Click **Next**.
13. Select the **SI SWIFT Standard** ODBC data source (which contains the Sterling Standards Database for SWIFT) and click **Next**.
14. Select the *same MT that you selected for the Input Format* and select the version, and click **Next**.

    Note: By selecting the same MT on both the input and output sides of the map, you are creating a "passthrough" map.
15. If you are translating to SWIFT and have selected a common group MT, select the MT for copy fields and click **Next**.
16. Click **Finish** to load the MT you selected.
17. Click **Next**.
18. Click **Finish**. The Sterling B2B Integrator Map Editor displays the new map in the Sterling B2B Integrator Map Editor window.
19. On the Output side of the map, right-click the SWIFT Properties icon and select **Properties**.
20. Select the **Message Type** tab.
21. In **Market Practice ID**, type the unique identifier for the market practice. This identifier distinguishes the map from the standard SWIFT message type.

    Note: This identifier must be 6 upper-case alphanumerics with no spaces. You create the unique Market Practice ID.
22. Use the **Save As** function to save the modified map under a new name.

23. Open a command window and change to the directory where you installed the SWIFT Standards Database.
24. Then, from the Sterling B2B Integrator Map Editor**File** menu, select **New** so you can use the customer Market Practice or Fund message to create a new map.
25. In the **New Map Wizard**, complete the questions in the first window and click **Next**.

    Note: Be sure that Sterling B2B Integrator is selected in the **What type of map are you creating** list.
26. In the Input Format window, Create a new data format using this standards, select **SWIFT (Society for Worldwide Interbank Financial Telecommunications)** and click **Messages**.
27. Click **Next**.
28. Select **Custom Market Practice or Fund Template** as the type of message to auto-generate.
29. Select the **SWIFT Standard** ODBC data source (which contains the Sterling Standards Database for SWIFT) and click **Next**.
30. Select the custom Market Practice or Fund message and click **Next**.
31. Follow the prompts to complete the map selection.
32. Click **Finish**. The Sterling B2B Integrator Map Editor displays the new map in the Sterling B2B Integrator Map Editor window.
33. Use Autolink to link your fields prior to using the Link function or creating standard or extended rules.
34. Modify the map to implement the additional restrictions related to the desired Market Practice or Fund message.
35. Use the Rule Library Manager to add a reusable extended rule library that can then be added to any map so you do not have to redefine extended rules for each map you create.

    Note: The SWIFT extended rule libraries that are installed with the Sterling B2B Integrator contain all the extended rules necessary to carry out the business logic for SWIFT messages. Extended Rule Libraries from 2008 and forward are delivered in the SWIFT Standards Install.
36. For the SWIFT Rule Importer command script, set the Java_home variable.
37. Run the SWIFT Rule Importer from the command prompt using the following syntax:

    `SWIFTRuleImporter MapName.mxl`
38. Save your map using the following naming convention:

    `SWIFT_<version>_<messageType>[_<messageExtension>]_<marketPracticeID>.mxl`

    Note: In this syntax, version is synonymous with year. For Market Practices, you do not need a messageExtension since all market practice maps are in the 5xx series. The map must be saved in MXL format.
39. Compile the map.
40. If necessary, update your property files with code word and qualifier validations that are specific to the Market Practice.

    Note: The properties file to update is **translator_swift_mp_customer_year.properties**, located in the **<install_dir>/properties** directory.

Create a SWIFTSolutions (MX) Map

About this task

When you create a SWIFTSolutions (MX) map, for both the input and output sides, you choose whether you want to use a preloaded standard, an existing file format, or want to create a new file format for that side of the map (including selecting from standards that you have previously downloaded to the standards database. The preloaded standards are downloaded when you download the Sterling B2B Integrator Map Editor to your machine and include SWIFTNet Funds, SWIFT eMandate, SWIFTNet Funds Passport, SWIFTNet Trade Services, SWIFTNet Exceptions and Investigations, SWIFTNet Cash Reporting, Bulk Payments, Cash Management, Cash Reporting Proxy Voting, Bank Account Management, FpML, SCORE, and Transaction Reporting.

For a complete list of the preloaded standards consult the Sterling B2B Integrator Map Editor New Map Wizard.

If WebSphere Transformation Extender is installed and integrated with Sterling B2B Integrator, you also have the option of using WebSphere Transformation Extender to create a SWIFTSolutions (MX) map. For details on creating maps with WebSphere Transformation Extender see the *WebSphere Transformation Extender Information Center and Pack for SWIFT*.

Note: If the map you are creating contains greater than 20,000 objects, you will receive a message noting that this map contains a very large number of objects. For best performance, it is recommended that you consider whether any unnecessary objects in the map can be removed, do not expand the entire object tree—expand only the section of the tree you are currently mapping, consider using the "Show links to or from the currently selected element" option instead of the "Show links to or from all visible elements" option, and save the map using the.MAP file format (using the Save As function).

Note: For the ContractCreated and ContractNovated SWIFTNet FpML messages, all conditional elements are disabled by default but a tree list is displayed to allow you to select the elements that you want to be generated.

To create a map using a preloaded SWIFTNet Fund message:

Procedure

1. From the Sterling B2B Integrator Map Editor **File** menu, select **New**.
2. In the **New Map Wizard**, complete the questions on the first screen.

   Note: To use XML schemas, the map type must be Sterling B2B Integrator
3. If you are translating from a SWIFTNet Fund message, from the **Create a new format using this standard** list, select **SWIFTNet Funds (or** SWIFTNet Bulk Payments,SWIFTNet Trade Services, SWIFTNet Exceptions and Investigations, SWIFTNet Funds Passport, SWIFTNet Transaction Reporting, SWIFT eMandate, SWIFTNet Proxy Voting, SWIFTNet SCORE, SWIFTNet FpML, SWIFTNet Cash Management, SWIFTNet Cash Reporting, SWIFTNet Corporate Actions, SWIFTNet Clearing, SWIFTNet Bank Account Management, or ISO 20022) and click **Messages**. If you are translating from another format, select that format and continue to the next screen.

4. Complete the Map Wizard by selecting the version of the standard, the desired message, the maximum length of the data elements, and whether to build codelists for enumerated attributes, and click **Next**.

   Note: The default for the **Build code lists for enumerations** check box is cleared because using code lists may be redundant and thus your map is smaller and more efficient without them.

5. Click **Next**.

6. If you chose to import an XML schema, do not modify any of the subelements selected in the display. *This list is included for your information only.* Click **Next**.

   If you are using an XML schema and the XML parser detects any errors, the messages are displayed in an error window.

7. Click **Finish**.

8. If you are translating to a SWIFTNet Fund message, on the Output screen (from the **Create a new format using this standard** list), select **SWIFTNet Funds (or** SWIFTNet Bulk Payments,SWIFTNet Trade Services, SWIFTNet Exceptions and Investigations, SWIFTNet Funds Passport, SWIFTNet Transaction Reporting, SWIFT eMandate, SWIFTNet Proxy Voting, SWIFTNet SCORE, SWIFTNet FpML, SWIFTNet Cash Management, SWIFTNet Cash Reporting, SWIFTNet Corporate Actions, SWIFTNet Clearing, SWIFTNet Bank Account Management, or ISO 20022) and click **Messages**. If you are translating from another format, select that format and continue to the next screen.

9. Complete the Map Wizard by selecting the version of the standard, the desired message, the maximum length of the data elements, and whether to build codelists for enumerated attributes, and click **Next**.

   Note: The default for the **Build code lists for enumerations** check box is cleared because using code lists may be redundant and thus your map is smaller and more efficient without them.

10. Click **Next**.

11. If you chose to import an XML schema, do not modify any of the subelements selected in the display. *This list is included for your information only.* Click **Next**.

    If you are using an XML schema and the XML parser detects any errors, the messages are displayed in an error window.

12. Click **Finish**.

    If the Sterling B2B Integrator Map Editor made changes to the DTD to make it compliant with the Sterling B2B Integrator, it indicates the changes. Click **OK**.

13. Continue with the **New Map Wizard** as directed. When you click **Finish**, the Sterling B2B Integrator Map Editor displays the new map in the **SWIFT Editor**.

    Note: After you click **Finish** in the Confirm page and the SWIFT Editor is displayed, you can click **View Text** to access an XML display of the message.

Convert Your Map to Another SWIFT Standards Version

About this task

The Update SWIFT Version features enable you to convert the SWIFT MT format side of your map to the most recent standards version from the standards database, and to save it to a new map.

Note: In some previous versions of the Sterling B2B Integrator Map Editor, SWIFT numeric fields were generated with a maximum of nine decimal places (R9 format). However, SWIFT special functions allow for greater precision. For example, AMOUNT<15> can have a maximum of 13 decimal positions, and Sterling B2B Integrator Map Editor supports numeric formats greater than R9. Therefore, the numeric fields have been upgraded for SWIFT and any previously generated map that is updated through the SWIFT Update feature will have the numeric fields updated to support the greater precision. These changes will be reported as "standard fixes" in the report.

The Update Version feature:
- Migrates the old map to the new SWIFT standard version.
- Identifies the differences between the required (mandatory) map components in the old version as compared with the new version and reports the following changes:
    - **STANDARD Change**: A change from the old map that was made by the new SWIFT standard.
    - **STANDARD Fix**: A change from the old map that was made by a fix to the Sterling B2B Integrator Map Editor.
    - **CUSTOM Change**: A change from the old map that was made by you (the customer).
    - **CONFLICT Change**: A change from the old map that was made by you (the customer), but the change conflicts with a change made by the new SWIFT Standard.
- Inserts the new required map components into the newly created version of the map.
- Ignores any "temporary" map components you have created to use as work areas for calculations and so forth. You can use temporary records and fields when you cannot use a simple link or if you must extract only specific occurrences of a record from your data file. A simple link enables you to join data from the Input and Output sides of the map in either a one-to-one relationship (map components that both do not repeat) or a many-to-many relationship (map components that repeat the same number of times).
- Provides you with a report that denotes the differences between the two standard versions and lists the segments and elements that are added as part of new standard version and were not in the old map. It will also list custom changes carried forward to the new version of the map.

    Note: If a field and record is no longer used in the new standard, it is deleted from the map and a note of the deletion is included in the report. If a link is broken as a result of a deletion, that is also noted in the report.

To upgrade to a newer SWIFT standards version:

Procedure
1. In Sterling B2B Integrator Map Editor, open the map.
2. Select a map component on the side of the map you want to upgrade.
3. Select **Functions** > **Update** > **SWIFT Version**.
4. Select the ODBC data source that contains the SWIFT standards database or the ODBC data source that contains the customer Market Practices. Click **Next**.
5. Type the name of the new map or accept the default, and click **Next**.

6. Click **Finish** to start the standard version upgrade process and create the new map.

The upgrade process reads in the new version of the SWIFT standard and the old version of the standard specified by the map. The upgrade procedure consists of a three-way comparison between the new SWIFT standard, the customer map, and the old SWIFT standard (that is, the version defined by your map).

SWIFTNet Syntax Validation

SWIFT field tag syntax validation is defined in the Sterling B2B Integrator Map Editor and stored in the **translator_swift_<year>.properties** file. This properties file contains the validation syntax for the field tags, as defined by the SWIFT standards (for example, field 73 is 35x['CRLF'35x]0-5), and may not be edited. The validation syntax for any given record is displayed in the Field Syntax parameter on the SWIFT Record Properties dialog box (SWIFT Validation tab). This parameter is automatically populated when you create a map using the SWIFTNet standards data dictionary. The convention of the syntax conforms exactly to the SWIFT documentation.

SWIFT also uses special validation functions such as <CUR>, <SWIFTBIC>, and <NON-SWIFTBIC>, which are components of a field tag. Each of these special functions has an expanded syntax that is validated by the translator. When these special validation functions are used, the special function syntax is displayed in the **Field Syntax** text box on the SWIFT Record Properties dialog box. The currency (<CUR>) and sender/receiver addresses (<SWIFTBIC>, <NON-SWIFTBIC>) must also be validated against legal code word lists. These code lists are defined in the SWIFT standards data dictionary and used by the Sterling B2B Integrator.

If WebSphere Transformation Extender is installed and integrated with Sterling B2B Integrator, you also have the option of using WebSphere Transformation Extender for the translation and compliance checking of WebSphere Transformation Extender maps.

Note: You will need to maintain these code lists in the Sterling B2B Integrator (not in the Sterling B2B Integrator Map Editor).

SWIFT MX Validation

Sterling B2B Integrator performs SWIFT MX validations. Validations are defined in the **translator_swift_mx.properties** property file and generated maps define the parameters used to lookup validates in this property file.

If WebSphere Transformation Extender is installed and integrated with Sterling B2B Integrator, you also have the option of using WebSphere Transformation Extender for the translation and compliance checking of WebSphere Transformation Extender maps.

The following is an example entry from the **translator_swift_mx.properties** property file for Currency validation. In this example, currency validation is performed on the XML element **Amt**, attribute **Ccy**.

```
swiftMX.tsmt.009.001.02.Amt.Ccy=CUR
```

Note: There are also currency validation functions: CURACTIVE (active currency) and CURACTHIST (active or historic currency).

The following is an example entry from the **translator_swift_mx.properties** property file for Country or Region validation. In this example, country or region validation is performed on the XML element **PdctOrgn**.

`swiftMX.tsmt.009.001.02.PdctOrgn=CC`

The following is an example entry from the **translator_swift_mx.properties** property file for IBAN validation. In this example, IBAN validation is performed on the XML element **IBAN**.

`swiftMX.tsmt.009.001.02.IBAN=IBAN`

The following is an example entry from the **translator_swift_mx.properties** property file for BIC validation. In this example, BIC validation is performed on the XML element **BIC**.

`swiftMX.tsmt.009.001.02.BIC=BIC`

When you create a SWIFT MX map, the parameters used to look up validations in the property file are defined on the Map Details dialog box. This table describes the Map Details parameters and their function in MX validation:

Map Details Parameter
> **Description of MX Validation Function**

Agency (description field)
> **SWIFT-MX** identifies the map as requiring SWIFT MX validation.

Version
> The version of the SWIFT solution (for example, 1-0).

Transaction (description field)
> The schema name is used as a unique key to lookup validation entries. For example, tsmt.009.001.02 is a unique key for a Trade Services Baseline Amendment Request in which:
> - tsmt is the SWIFT solution (for example, Trading Services)
> - 009.001.02 is the schema name

Required Error Codes for MX Validation

Sterling Standards Library supports Generic, Cash Management, Cash Reporting, and Funds rule book validation error codes. The error codes are logged when MX validation errors occur. Errors are logged in the translator report and in the SWIFT Message Editor when you are using the Document Repair function.

Note: For a complete listing of MX validation error codes, see the `translator_swift_error_codes.properties` file.

Table 1. Generic rule book validation error codes

Error code	Description
Sw.Stds.D00001	Invalid BIC code.
Sw.Stds.D00002	Invalid BEI code.
Sw.Stds.D00003	Invalid IBAN format or invalid check digits. A valid IBAN consists of all 3 of the following components: # Country Code, check digits, and BBAN.

Table 1. Generic rule book validation error codes (continued)

Error code	Description
Sw.Stds.D00004	Invalid Country Code. The code is checked against the list of country names obtained from the United Nations (ISO 3166, Alpha-2 code).
Sw.Stds.D00005	Invalid active currency code.
Sw.Stds.D00006	Invalid active or historic currency code.
Sw.Stds.D00007	Invalid Currency Amount. Also, the number of fractional digits (or minor unit of currency) must comply with ISO 4217.
Sw.Stds.D00008	Invalid BIC or BEI code.

Creating Extended Rules for SWIFT Maps

In the next release of SWIFT there will be support to update existing maps to include or add any message rule validation changes. This future update feature will impact you only if you want custom code in the extended rules (in existing maps) included in the validation block that only runs when validation is enabled. The update feature places custom extended rule code outside of the validation block. Thus, if you want to create custom rules used only during validation, then you need to follow the example below when you write extended rules for SWIFT maps to ensure they are only run when validation is enabled:

```
if validation = 1 then
begin
    {type your custom validation extended rule code}
end
```

Extended Rules Used with SWIFT Maps

The following new extended rules are used with SWIFT maps, though they can also be used with any other data format:

- cerror
- occurrencetotal
- resetoccurrencetotal
- sum
- sumtotal

See *Alphabetic Language Reference* for information on all the available extended rules.

cerror

The cerror function raises a compliance error in the translation report. This extended rule has been updated for use with SWIFTNet FIN to allow it to be called with only a code and description string (instead of code, field reference, option description string). You typically specify this function as an action to be performed if a condition is false. This function is valid on the input side of a map only. There is also an optional third parameter you can supply—a string which is written to the translator report as part of the entry for the compliance error you are raising.

```
cerror(code,$GROUPNAME[0][1][1].#FIELDNAME, "Optional string with error information
can be supplied here");
```

Example1: An example of this function follows:

```
resetoccurrencetotal(%myRec);
//This returns the current sum of the field debitAmount (on which the sum function
//was used, and returns the current sum of that field into the debitTotal field.
```

Syntax:

```
cerror(code, "String with error information supplied here");
```

Example2: An example of this function in this syntax follows:

```
cerror(100, "Number not valid");
//This raises compliance error 100 with error text "Number not valid" in the
//translator report.
```

occurrencetotal

The occurrencetotal function allows an extended rule to check the total occurrences
of a particular record thus far in the processing (this is a running total of the
number of times the record has occurred at the specific point in time that the
occurrencetotal rule is called). This function returns an integer value containing the
number of occurrences of the specified record that have been processed.

Note: To reset OccurrenceTotal to zero so you can determine, for example, if a
particular block occurred in the current iteration of a group, use the
resetoccurrencetotal function.

Syntax

```
occurrencetotal(%recordname);
```

Note: The syntax %recordname indicates a record that is referenced by name.

Example An example of this function follows:

```
If occurrencetotal(%myRec) > 2 then
//This specifies that if the myRec record has occurred (been processed) more than
//twice, then the logic that follows this function will be performed by the
//translator.
```

resetoccurrencetotal

The reset occurrencetotal function allows you to reset OccurrenceTotal to zero so
you can determine if a particular block occurred in the current iteration of a group,
as opposed to learning how many times the translator has encountered a specified
block in the course of processing the data on either the input or output side of a
map.

Syntax

```
resetoccurrencetotal(%recordname);
The syntax %recordname indicates a record that is referenced by name
```

Example

An example of this function follows:

```
resetoccurrencetotal(%myRec);
//This specifies that the Occurrencetotal for the myRec record will be set to zero.
```

sum

The sum function is used (along with the sumtotal function) to maintain and validate the sum of a specified numeric field. This function is called with no parameters in the extended rule of the field for which you need to keep a running sum. Optionally, you can add a string parameter—the first character in the string indicates whether the value of the field is treated as a negative number before it is added to the running sum of the field. If the first character of the string begins with N or n, the value from the field is treated as a negative and added to the *sumtotal*.

Syntax 1

```
sum();
```

Example1

An example of this function follows:

```
sum();
//This maintains a running sum for the field on which the extended rule is called.
```

Syntax 2

```
sum("nString");
```

Example 2

An example of this function follows:

```
sum("n");
//This adds the value of the field as a negative number to the sumtotal to maintains
//a running sum for the field on which the extended rule is called.
```

sumtotal

The sumtotal function is used (along with the sum function) to maintain and validate the sum of a specified numeric field. This function returns a real numeric value and is used to obtain the current sum of a specified field wherever the value is needed. For example, if want to verify the sum of a specific field (debitAmount), add the sum function to the extended rule of this field. Then, to access this running sum in a later field (debitTotal), use the sumtotal function on the debitTotal field.

Syntax

```
sumtotal(#fieldname);
```

Example

An example of this function follows:

```
sumtotal(debitTotal);
//This returns the current sum of the field debitAmount (on which the sum function
//was used, and returns the current sum of that field into the debitTotal field.
```

Editing Syntax

Procedure

1. In the Sterling B2B Integrator Map Editor, double-click an existing field or create a new one.

2. In the **appropriate Properties** dialog box, click the **Validation** tab.
3. From the data-type list, select **String**.
4. From the data format list, select **Free Format**. The drop-down list becomes an editable text box. Free Format indicates that any characters are acceptable in the field. The translator does not check the characters for compliance.
5. In the text box, type the complex syntax for this field. For example, this enables you to define a complex syntax such as **3a[2!n]**, which indicates that the String should consist of zero to three uppercase letters followed by an optional two digits.
6. Click **OK** to save your selections.

Using Autolink and Link

The Autolink function automatically creates links between input and output fields that have the same name or which contain logically equivalent business data. This functionality can be used regardless of which format you have selected for the input and output sides of your map. You can choose to link by either Field Name or Business Name.

Note: To increase the likelihood that the links in your maps are valid, in the Preferences dialog box Confirmations tab, select the link objects at different levels and link objects with different maximum usages confirmations.

Just like with the Link function, the link between two map components is represented visually with a connecting line.

Note: An erroneous message is displayed when you deactivate a group that contains linked fields in the Sterling B2B Integrator Map Editor. The message displayed is: "The object is part of one or more links. Deleting the object will remove the links. Do you want to continue?" This message should read "The object is part of one or more links. Deactivating the object will remove the links..." If you receive this message when deactivating a linked map component, be assured that the group object is not deleted, and you should ignore the erroneous message.

Using Extended Rule Libraries with SWIFT

This section describes how to use the extended rule library and the properties of the dialog boxes that comprise its functionality. A rules library (used with SWIFT and any other data format) contains a list of rules in a separate file outside of the Sterling B2B Integrator Map Editor source. Sterling B2B Integrator Map Editor stores the name of the library in its source file, so when you open a map the library is also loaded. When you compile a map, the library rules that are referenced in the map are also compiled. This enables you to create a library of extended rules and then add it to any other map, so you do not have to recreate those extended rules after the first time. You can use this functionality with any data format.

Note: The SWIFT extended rules libraries from 2008 and forward are delivered in the SWIFT Standards install. These contain all the extended rules necessary to carry out the business logic for SWIFT messages.

Note: Please note that for all Sterling B2B Integrator versions 4.3 and higher, changes have been made to the extended rule library for SWIFT_2007 to report field errors more accurately in the Message Entry Workstation. For this reason, many SWIFT_2007.erl methods were modified and are no longer compatible with

maps that were created prior to this change. If you encounter this issue, you can recreate your map using the latest version of the standard.

Semantic validation rules for Market Practices are stored in an extended rule library separate from the library that implements the SWIFT standard semantic validation rule

Only the extended rule libraries referenced by a map are compiled into the TXO translation object.

This functionality minimizes the impact to users when, for example, SWIFT updates their messages—without the rule library you would need to update the extended rules for each updated map (correlating to the updated messages), but using the extended rule library you just update the library and then use the library with all the applicable maps.

When you view the checked in libraries through the Extended Rule Library check in interface, you are also able to obtain a list of all the maps that use each library.

The extended rules library can contain many rules. An extended rule consists of a declarations section followed by a statements section. The *declarations* section is required only if you use additional variables. The declarations section is where you declare the names and types of any variables you use either in the extended rule. The *statements* section is where you define the actions that you want the extended rule to run.

When calling a rule library function, you can pass parameters.

You must declare any variables that are not already defined as part of the input or output specification of the map before you use those variables in an extended rule. For the extended rule libraries, you typically use global variables that are passed as parameters.

Rule libraries are versioned resources. When you create a new rule library you need to check it in to the Sterling B2B Integrator just like you need to check in maps. This also enables you to check out, version, and delete extended rule libraries. Furthermore, when you view the checked in libraries through the Extended Rule Library check in interface, you can also see all the maps that use each library. This is very important because it enables you to easily view a list of the maps that will need to be recompiled if you change an extended rule in a library (you would recompile all the maps that use that particular library).

Additionally, you can import and export extended rule libraries into the Sterling B2B Integrator using the Resource Manager.

You can call an extended rule from a library in any extended rule in a map.

See *Extended Rule Libraries* for more information on this functionality.

The extended rule libraries that are preloaded for use with SWIFT are as follows:

Library	ERL File Name	Description
SWIFT_2002	SWIFT_2002.erl	Contains rules to validate SWIFT 2002 MT and MX messages

Library	ERL File Name	Description
SWIFT_2005	SWIFT_2005.erl	Contains rules to validate SWIFT 2005 MT and MX messages
SWIFT_2006	SWIFT_2006.erl	Contains rules to validate SWIFT 2006 MT and MX messages.
SWIFT_2007	SWIFT_2007.erl	Contains rules to validate SWIFT 2007 MT and MX messages.
SWIFT_2008	SWIFT_2008.erl	Contains rules to validate SWIFT 2008 MT and MX messages.
SWIFT_2009	SWIFT_2009.erl	Contains rules to validate SWIFT 2009 MT and MX messages.
SWIFT_2010	SWIFT_2010.erl	Contains rules to validate SWIFT 2010 MT and MX messages.
SWIFT 2011	SWIFT_2011.erl	Contains rules to validate SWIFT 2011 MT and MX messages.
SWIFT 2012	SWIFT_2012.erl	Contains rules to validate SWIFT 2012 MT and MX messages.
SWIFT 2013	SWIFT_2013.erl	Contains rules to validate SWIFT 2013 MT and MX messages.
SWIFT_IOA001_2005	SWIFT_IOA001_2005.erl	Contains rules to validate 2005 Market Practice U.S. MT536
SWIFT_IOA001_2006	SWIFT_IOA001_2006.erl	Contains rules to validate 2006 Market Practice U.S. MT536
SWIFT_IOA001_2007	SWIFT_IOA001_2007.erl	Contains rules to validate 2007 Market Practice U.S. MT536
SWIFT_IOA001_2008	SWIFT_IOA001_2008.erl	Contains rules to validate 2008 Market Practice U.S. MT536
SWIFT_DeenvelopeLib_2008	SWIFT_DeenvelopeLib_2008.erl	Contains rules to deenvelope 2008 SWIFT messages.
SWIFTMX	SWIFTMX_v1-0.erl (generic rule library for MX)	Contains rules to validate MX messages.
SWIFTMX_BulkPayments_v2-0	SWIFTMX_BulkPayments_v2-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_BulkPayments_v2-1	SWIFTMX_BulkPayments_v2-1.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_CashManagement_v4-0	SWIFTMX_CashManagement_v4-0.erl	Contains rules to validate SWIFT Cash Management 4.0 messages.
SWIFTMX_CashManagement_v2009	SWIFTMX_CashManagement_v2009.erl	Contains rules to validate SWIFT Cash Management 2009 messages.
SWIFTMX_CashManagement_v2013	SWIFTMX_CashManagement_v2013.erl	Contains rules to validate SWIFT Cash Management 2013 messages.

Library	ERL File Name	Description
SWIFTMX_CashReporting_v1-0	SWIFTMX_CashReporting_v1-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_CashReporting_v3-0	SWIFTMX_CashReporting_v3-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_CashReporting_v3-1	SWIFTMX_CashReporting_v3-1.erl	Contains rules to validate SWIFT Cash Reporting 3.1 messages.
SWIFTMX_CashReporting_v3-2	SWIFTMX_CashReporting_v3-2.erl	Contains rules to validate SWIFT Cash Reporting 3.2 messages.
SWIFTMX_CashReporting_v4-0	SWIFTMX_CashReporting_v4-0.erl	Contains rules to validate SWIFT Cash Reporting 4.0 messages.
SWIFTMX_CashReporting_v5-0	SWIFTMX_CashReporting_v5-0.erl	Contains rules to validate SWIFT Cash Reporting 5.0 messages.
SWIFTMX_EI_v1-0	SWIFTMX_EI_v1-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_EI_v1-1	SWIFTMX_EI_v1-1.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_EI_v1-2	SWIFTMX_EI_v1-2.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_e-Mandate_v1-0	SWIFTMX_e-Mandate_v1-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_FpML_v1-0	SWIFTMX_FpML_v1-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_Funds_v2-0	SWIFTMX_Funds_v2-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_Funds_v2-1	SWIFTMX_Funds_v2-1.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.

Library	ERL File Name	Description
SWIFTMX_Funds_v2-2	SWIFTMX_Funds_v2-2.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_Funds_v3-0	SWIFTMX_Funds_v3-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_Funds_v3-1	SWIFTMX_Funds_v3-1.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_Funds_v4-0	SWIFTMX_Funds_v4-0.erl	Contains rules to validate SWIFT Funds 4.0 messages.
SWIFTMX_Funds_v4-1	SWIFTMX_Funds_v4-1.erl	Contains rules to validate SWIFT Funds 4.1 messages.
SWIFTMX_Funds_v4-2	SWIFTMX_Funds_v4-2.erl	Contains rules to validate SWIFTNet Alternative Funds 4.2 messages.
SWIFTMX_Funds_v4-4	SWIFTMX_Funds_v4-4.erl	Contains rules to validate SWIFTNet Alternative Funds 4.4 messages.
SWIFTMX_Funds_v4-5	SWIFTMX_Funds_v4-5.erl	Contains rules to validate SWIFTNet Alternative Funds 4.5 messages.
SWIFTMX_ProxyVoting_v1-0	SWIFTMX_ProxyVoting_v1-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_ProxyVoting_v1-1	SWIFTMX_ProxyVoting_v1-1.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_ProxyVoting_v1-2	SWIFTMX_ProxyVoting_v1-2.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_TradeServices_v1-0	SWIFTMX_TradeServices_v1-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_TransactionReporting_v1-0	SWIFTMX_TransactionReporting_v1-0.erl	Contains rules to validate SWIFT Transaction Report 1.0 messages.
SWIFTMX_Clearing_v1-0	SWIFTMX_Clearing_v1-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.

Library	ERL File Name	Description
SWIFTMX_Clearing_v2-0	SWIFTMX_Clearing_v2-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_CorporateActions_v1-0	SWIFTMX_CorporateActions_v1-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_CorporateActions_v1-1	SWIFTMX_CorporateActions_v1-1.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_CorporateActions_v1-2	SWIFTMX_CorporateActions_v1-2.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_CorporateActions_v1-3	SWIFTMX_CorporateActions_v1-3.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_CollateralManagement_v1-0	SWIFTMX_CollateralManagement_v1-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_CollateralManagement_v1-2	SWIFTMX_CollateralManagement_v1-2.erl	Contains rules to validate SWIFT Collateral Management 1.2 messages.
SWIFTMX_SCORE_v2-5	SWIFT_SCORE_v2-5.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_FundsProcessingPassport_v4-2	SWIFTMX_FundsProcessingPassport_v4-2.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_FundsProcessingPassport_v4-3	SWIFTMX_FundsProcessingPassport_v4-3.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_AlternativeFunds_v4-1	SWIFTMX_AlternativeFunds_v4-1.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.
SWIFTMX_BankAccountManagement_v2-0	SWIFTMX_BankAccountManagement_v2-0.erl	This rule library contains no rules, but you can implement and store rules for the SWIFTSolution indicated in the library.

Calling a Rule from an Extended Rule Library in a Map

You can call a rule from any extended rule library (that is currently checked in to the Sterling B2B Integrator) in any extended rule in your map. The syntax you use to call a rule from a library is:

```
call library_name.rule_name(parameter1, parameter2, parameter3)
```

In this syntax, **library_name** is the name of the extended rule library. For example, if the library is SWIFT_2012.erl, the library_name is **SWIFT**. The **rule_name** is the name of the rule that you defined in the Library Rule dialog box. There is no limit on the number of parameters you can use.

Note: You can have multiple rule libraries with the same name and different version numbers, but you can only use one rule library of the same name in a map (the last version of that rule library that was checked in to the system).

The syntax you use to call a rule with a return value set is:

```
integer i;
i = call library_name.rule_name parameter1
```

In this syntax, **i** is the return value set.

Truncate Number Fields When Converting Strings to Numbers

About this task

By default, the translator trims all trailing zeros from output values. For example, 80000.0100 will be output as 8000.01. This behavior is not desirable for financial standards such as SWIFT, for which trailing zeros represent a specific amount for a particular currency.

Therefore, the **storage.keepTrailingZeros** property in the **translator.properties** file enables you to specify whether or not trailing zeros are kept on any numeric field on the output side of the map (numeric fields that have been converted to strings).

By default trailing zeros are trimmed (storage.keepTrailingZeros=false). If you set this property to True, this will preserve trailing zeroes. Therefore, by default the value 3.142000 is truncated to 3.142, but if you set the **storage.keepTrailingZeros** property to True, the resulting string value is 3.142000.

To set the **storage.keepTrailingZeros** property to True:

Procedure

1. In the *install_dir*/properties directory, locate (or create, if necessary) the **customer_overrides.properties** file.
2. Open the **customer_overrides.properties** file in a text editor.
3. Add the property that you want to override, using the following format:
   ```
   translator.storage.keepTrailingZeros=True
   ```
4. Save and close the **customer_overrides.properties** file.
5. Restart the Sterling B2B Integrator to use the new values.

Using BigDecimal

The Sterling B2B Integrator allows you to use either Java™ double primitive or BigDecimal data types for real numbers. BigDecimal can be used regardless of the standard you are using, but if you are using a financial standard like SWIFT we highly recommend that you use BigDecimal. See *BigDecimal Support for Real Numbers* for more information on using this functionality. When processing SWIFT messages with BigDecimal mode turned off, real values (such as amounts) may be rounded when those messages are processed through the SWIFT Message Editor. When rounding occurs, the changed values are not reported.

If you are using BigDecimal mode on a per map basis for the SWIFT standard, you also need to enable BigDecimal (turn it on) for any validation pass-through maps you are using. It is also possible to use double primitive mode by default, but to enable BigDecimal mode on a per-map basis. Validation pass-through maps should have BigDecimal mode enabled (turned on).

Chapter 5. SWIFTNet Correlations

Correlation data, often called *correlations,* is defined as specific pieces of data that you may need to review in the process of monitoring, tracking, and troubleshooting your activities. These data items are defined by type and value. These type-value pairs are known as *name-value pairs*, and are a powerful tool you can use to record and search for business process- and document-specific data. The name-value pair records are stored in the correlation table in the Sterling B2B Integrator.

Defining correlations can be part of configuring your business process models and maps. Taking the time to determine the name-value pairs for monitoring and troubleshooting, and configuring them in maps and business process models, can save you time later.

Correlations can be predefined (provided in certain services), defined by you, and configured at the map level:

- You can assign name-value pairs to a service in a process model using the Service Editor in the GPM.
- You can create unique name-value pairs for a service using the Service Editor in the GPM.
- You can define name-value pairs and include the service in a business process model. This enables you to search for the data items you need using advanced search options in the Sterling B2B Integrator interface.
- You can configure correlation data using standard rules in a map, to specify that data is extracted from a document when it is translated. The Update standard rule *correlation data* function enables you to record document-specific correlation parameters during translation. These correlation parameters are attached to the translated document. You can then use the Correlation Search user interface to locate the translated document using the criteria you specified in the map through the Update standard rule. This function may save you effort, because you would otherwise need to locate the translated document by reviewing the results from a Central Search query.

The Correlation Search page provides details about each SWIFTNet transport transaction. Each transaction is a tracking point for any workflow IDs and documents involved in the transaction. The SWIFTNet Correlation search feature offers the following additional benefits:

- You can receive the results of search queries more quickly.
- You can further refine correlation searches by specifying a start and end date/time range.
- You can further refine SWIFT correlation searches by specifying SWIFT-specific criteria.

Search for SWIFT Messages Using Correlations

About this task

To perform an advanced search for documents:

Procedure

1. From the **Administration** menu, select **Business Process** > **Monitor** > **Advanced Search** > **SWIFTNet Correlation**.
2. In the **Search SWIFTNet Correlation** section, specify any combination of the following search criteria, as appropriate:

Field	Description	Action
Location	The location of the tables you wish to search.	Select Live Tables or Restore Tables.
Requestor DN	Distinguished name of the requestor.	Type the name of the requestor. **Note:** This DN must be registered with the SAG instance using SWIFTNet Alliance Webstation.
Responder DN	Distinguished name of the responder.	**Note:** This DN must be registered with the SAG instance using SWIFTNet Alliance Webstation.
Service Name	Name of the service to which both SWIFT correspondents have subscribed.	Type the service name. **Note:** This must be a SWIFTNet service to which you are subscribed.
Request Type	Request type supported by the message exchange.	Type the request type.
Interface	SWIFTNet message type.	Select either InterAct or FileAct.
Operation	The SWIFTNet operation to send an InterAct or FileAct message.	Select from the following: • get (FileAct) • put (FileAct) • sync (Interact) • async (Interact)
Reference	The message reference number.	Select from the following: • SWIFT Reference • Transfer Reference • SnF Reference
SnF	Indicates if the file transfer is done using the store-and-forward method.	Select either True (use Store-and-Forward) or False (do not use Store-and-Forward).
Direction	Indicates the direction of the messages.	Select either Inbound or Outbound.
Status	The status of the messages.	Select one of the following formats: • Success • Failed • In Process
Activity From	Documents in progress or completed after the specified start date and time.	Using the following formats, type a starting date and time range and select **AM or PM**: • Date – MM/DD/YY • Time – HH:MM

Field	Description	Action
Activity To	Documents in progress or completed before the specified end date and time.	Using the following formats, type an end date and time range and select **AM or PM.**: • Date – MM/DD/YY • Time – HH:MM
File Name (FileAct Only)	Search for SWIFTNet records that are associated with a specific file.	Type the name of the file for which you want to search the associated records. **Note:** This search is valid for FileAct messages only.
File Size From (FileAct Only)	Search for SWIFTNet records that are associated to files with a specified size range.	Type the beginning size range of the files.
File Size To (FileAct Only)	Search for SWIFTNet records that are associated to files with a specified size range.	Type the endpoint of the range for the file size.
File Size (FileAct Only)	Measurement of file size	Select from the following: • bytes • KB • MB
Results per page	Number of documents you want to display on the results page.	Select the value to indicate the number of documents to display. Valid values are: • 5 • 10 • 15

3. Click **Go!** and the Search Results page opens, listing the SWIFTNet Transaction and SWIFTNet Message details that match your search criteria.

Field   Description

Status Whether the transaction was a success or failure.

Requestor DN - Responder DN
>   The Requestor/Responder pair for the transaction.

Interface/Operation
>   The transport mode for the transaction.

Service Name
>   The service name for the transaction.

Type   The request type for the transaction.

Ref   The reference for the transaction.

File Name (bytes size)
>   The size of the Request/Response payload for InterAct transactions or the Transfer file name with size for FileAct transactions.

Msg Type
>   The type of message. Valid values are Request, Response, Delivery Notification, and Transfer Event.

Status The status of the message. Valid values are Accepted, Failed, In Progress, and Completed.

WF ID
　　A link to the initiating workflow identifier of the transaction.

Start Time
　　The start time for the transaction.

Document
　　A link to the document corresponding to the transaction.

Signature List
　　A link to the signature list of the document for the transaction (for FileAct transactions only).

Delivery Notification
　　A link to show the delivery notification status, either from the responder or the third party.

Migrating Correlation Details to Version 5.3

If you upgrade to the IBM Sterling Standards Library Version 5.3, you must run the **SwnetCorrMigrate.cmd** migration script because the SWIFTNet Correlations details are now stored in the SWNET_MSG_INFO and SWNET_MSG database tables instead of the CORRELATION_SET table.

The Migration script transfers the SWIFTNet Correlation data from the CORRELATION_SET table to the SWNET_MSG table (where TYPE = SWIFTNET MESSAGE) and the SWNET_MSG_INFO table (where TYPE = SWIFTNET TRANSACTION).

The Migration script is available for both Live and Restore records, and you run the SwnetCorrMigrate.cmd script with the argument either live or restore, depending on which type of records you want to migrate. If you do not specify an argument, the live records are migrated.

Chapter 6. Overview of SWIFTNet Transport

Sterling B2B Integrator enables you to send messages to SWIFTNet using either the InterAct or FileAct protocol. Sterling B2B Integrator enables you to connect to SWIFTNet through the InterAct protocol for real-time messaging, store-and-forward messaging and real-time query and response. It supports delivery notification, non-repudiation, and message priority.

InterAct and FileAct Protocol

When you use the InterAct or FileAct protocol to transport messages to and from SWIFTNet, the SWIFTNet Server serves requests and receives messages to and from SWIFTNet, through a client application and a server application that communicate with the SWIFTNet network through the InterAct protocol. The SWIFTNet Server operates independently from the Sterling B2B Integrator.

The SWIFTNet Server Adapter or SWIFTNet7 Adapter is responsible for receiving request messages from SWIFTNet through the SWIFTNet Server and sending responses back. The SWIFTNet Server Adapter or SWIFTNet7 Adapter is comprised of two parts: the service part and the adapter part. The service part is used in a business process that does not require configuration except for enabling it for document tracking. The adapter part is configured through the Admin Console or the GPM, and this adapter is responsible for starting and stopping the SWIFTNet Server from Sterling B2B Integration using the Command Line Adapter 2 (CLA2), which is built into the SWIFTNet Server Adapter or SWIFTNet7 Adapter. A Command Line 2 client operates in remote installations (with the SWIFTNet Server) to enable Sterling B2B Integration to run a program from a command line in a business process.

The SWIFTNet Client service is responsible for sending SWIFTNet InterAct and FileAct messages (both requests and responses) to SWIFTNet, which are initiated by the Sterling B2B Integrator. The SWIFTNet Client service enables you to use InterAct and FileAct messaging with a store-and-forward option. The benefits of using store-and-forward include:

- The sender and receiver do not need to be online at the same time, as is necessary for real-time messaging.
- The sender is notified if a message cannot be delivered (and can optionally be notified when messages are delivered).

The SWIFTNet Client service enables you to use either synchronous or asynchronous messaging using InterAct and Put or Get mode in FileAct.

The SWIFTNet Routing Rule is used by the SWIFTNet Client or SWIFTNet7 Client service to route incoming request to a business process for processing. It uses four parameters:

- RequesterDN
- ResponderDN
- Service name
- Request type

These parameters are used to map an incoming request to a business process. The SWIFTNet Routing Rule page enables you to assign any business process to a set of routing parameters. You need to create a SWIFTNet routing rule and associate it with an appropriate business process to process incoming SWIFTNet requests.

Additional FileAct Protocol Options

To send and receive SWIFTNet messages through the FileAct protocol, you will either use the SWIFTNet Server or use one of the following two methods to connect the Sterling B2B Integrator to SWIFT:

- **WebSphere MQ Adapter or WebSphere MQ Suite** (see Chapter 17, "Configuring the WebSphere MQ Adapter/Suite to Communicate with SWIFT," on page 249)—enables you to configure the Sterling B2B Integrator to send and receive InterAct and FileAct files to/from SWIFTNet through the WebSphere MQ Interface for SWIFTAlliance Access (MQSA). Also enables you to send and receive messages to/from SWIFTNet though SWIFTNet Alliance Access (SAA) and the SWIFTNet Remote API Host Adapter (RAHA).
- **IBM Sterling Connect:Direct® for SWIFTNet**—enables you to send and receive FileAct files to/from SWIFTNet through the SWIFTNet Remote API (SWIFTNet RA) and the SWIFTNet Remote API Host Adapter (RAHA).

Secure Sockets Layer (SSL)

Secure Sockets Layer (SSL) is a cryptographic protocol that provides secure communications on the Internet for activities such as web browsing, e-mail, Internet faxing, instant messaging, and other data transfers.

SSL provides endpoint authentication and communications privacy over the Internet using cryptography. When you use SSL, usually only the server is authenticated (the client is not authenticated). The authentication of the server ensures that the end user (which may be a person or an application such as a web browser) knows exactly with whom he or she is communicating. Once the server is authenticated, all the data communicated is encrypted and secured between the client and server only.

The Sterling B2B Integrator provides you with the ability to set up SSL in a loopback between the SWIFTNet HTTP Server and the SWIFTNet Server, so both ends of the communication are secure. You can also initiate or receive an InterAct or FileAct request using SSL (another form of loopback).

CHIPS Message Transport Using SWIFTNet

The Sterling B2B Integrator enables you to send CHIPS messages to CHIPS, using either the SWIFTNet network (optionally using IBM Websphere MQ) or The Clearing House Frame Relay Network (a proprietary network that uses IBM Websphere MQ). Acknowledgements are sent to CHIPS using the SWIFTNet Server adapter. Using the SWIFTNet transport available in the Sterling B2B Integrator, you can also receive all response messages from CHIPS, including heartbeat messages, and send supervisory STATUS messages to CHIPS to test the connection.

The CHIPS adapter works with the SWIFTNet Server adapter, SWIFTNet HTTP Server adapter, and the SWIFTNet Server to communicate with CHIPS. When the CHIPS adapter is used with the SWIFTNet network, it receives acknowledgement messages from CHIPS in the SWIFTNet Response within sixty seconds, and any incoming messages (for example, heartbeat message, resolver notification, and so

forth) are received by SWIFTNet Server adapter. The return acknowledgement of the incoming messages is performed by the Receive Handler and Acknowledgement Handler within the CHIPS adapter (the business process is bootstrapped using the SWIFTNet Routing Rule).

Chapter 7. SWIFTNet Server

The SWIFTNet Server serves requests and receives messages to and from SWIFTNet, through a client application and a server application that communicate with the SWIFTNet network through the InterAct or FileAct protocol. The SWIFTNet Server operates independently from the Sterling B2B Integrator and includes all the APIs necessary to communicate with the SWIFTNet network.

Sterling B2B Integrator enables you to use either InterAct or FileAct messaging with a store-and-forward option. The benefits of using store-and-forward include:

- The sender and receiver do not need to be online at the same time, as is required for real-time messaging.
- The sender is notified in the event delivery fails (and can optionally be notified upon delivery of the message).

Sterling B2B Integrator also has a feature that provides you with failover support from real-time messages to store-and-forward (if there is a failure in real-time messaging, you can configure the Sterling B2B Integrator to automatically switch to store-and-forward messaging to increase your messaging success).

Note: This feature requires subscription to both real-time and store-and-forward services.

The SWIFTNet client application sends requests to the SWIFTNet network through the SWIFTNet Alliance Gateway/SWIFTNet Net Link (SAG/SNL) instance. The client application listens for requests from the SWIFTNet Client or SWIFTNet7 Client service, and interacts with SWIFTNet to obtain responses.

The SWIFTNet Server application receives requests from SWIFTNet. The server application listens for requests from SWIFTNet and interacts with the Sterling B2B Integrator to obtain responses. A request from the server application to the Sterling B2B Integrator calls the SWIFTNet Server adapter or SWIFTNet7 Adapter to process the request.

The SWIFTNet Server server application is started by enabling (and stopped by disabling) the SWIFTNet Server Adapter or SWIFTNet7 Adapter. The starting and stopping of the server application is handled through the Command Line Adapter 2, which is built into the SWIFTNet Server Adapter or SWIFTNet7 Adapter.

The administration of the SWIFTNet Server is through SWIFTNet Server Adapter or SWIFTNet7 Adapter, including enabling and disabling the SWIFTNet Server.

If you use the SWIFTNet HTTP Server adapter in conjunction with the SWIFTNet Server Adapter or SWIFTNet7 Adapter to use Secure Sockets Layer (SSL), the SWIFTNet HTTP Server adapter accepts the forwarded request from the SWIFTNet Server and provides secure authentication.

The SWIFTNet Client service (in conjunction with the SWIFTNet HTTP Server adapter) enables you to use Secure Sockets Layer (SSL) to provide secure authentication, using the SWIFTNet HTTP Server adapter to accept the forwarded request from the SWIFTNet Server. When you use SSL with the Sterling B2B

Integrator, two channels are secured: an Outbound channel (the Sterling B2B Integrator acting as the Requestor) and an Inbound channel (the Sterling B2B Integrator acting as the Responder).

You will need 2 pairs of certificates. The first pair belongs to the SWIFTNet Server and is used to secure the outbound channel. The second pair of certificates belongs to the Sterling B2B Integrator and is used to secure the inbound channel. In the above diagram, the callouts signify the following:

- A public key certificate file belongs to the SWIFTNet Server that is configured on the SWIFTNet Client or SWIFTNet7 Adapter (the certificate is specified for the CA Certificate parameter).
- A private key certificate file that is stored on the SWIFTNet Server as a key file (which you configure through the SSL Configuration utility named sslUtil.jar in the SWIFTNet Server installation bin subdirectory). The sslUtil.jar file is located in the bin subdirectory of the SWIFTNet Server installation directory.
- A private key certificate file that is configured on the SWIFTNet HTTP Server adapter or SWIFTNet7 HTTP Server adapter (the certificate is specified for the System Cert parameter).
- A public key file that belongs to Sterling B2B Integration and is stored for the SWIFTNet Server as a CA Cert file or trusted list (that you configure through the SSL Configuration utility named sslUtil.jar in the SWIFTNet Server installation directory).

Exchanging Messages in the Client Application

The client application can exchange messages in synchronous or asynchronous mode for InterAct processing or can exchange messages in Put or Get mode for FileAct processing.

Synchronous Message Exchange

When the client application is communicating in synchronous mode messaging, the SWIFTNet Client or SWIFTNet7 Client service prepares the request and sends it to the SWIFTNet Server. Then, the client application on the SWIFTNet Server processes the request, performs the necessary communication exchange with the SWIFTNet SAG/SNL instance, and sends the request to the SWIFTNet network.

In synchronous mode, the client application is blocked until a response is received from the responder through the SAG/SNL instance. Once a response is received, it is sent to the Sterling B2B Integrator by the client application, and the response is placed in the primary document.

Asynchronous Message Exchange

In asynchronous mode, the SWIFTNet Client or SWIFTNet7 Client service prepares the request and sends it to the SWIFTNet Server. The client application on the SWIFTNet Server processes the request, performs the necessary communication exchange with the SWIFTNet SAG/SNL instance, and sends the request to the SWIFTNet network.

In asynchronous mode, the client application receives a response handle from the SAG/SNL instance. Using this response handle, the client application periodically checks with the SWIFTNet network to determine if a response is available . Once a response is received by the Sterling B2B Integrator, it is placed in the primary document.

Configuring the Client Application

You must appropriately configure the SWIFTNet Client or SWIFTNet7 Client service to use the client application. See *SWIFTNet Client* or *SWIFTNet7 Client Service*.

Server Application

When a request from the SWIFTNet network arrives, the SWIFTNet SAG/SNL sends it to the SWIFTNet Server server application. The server application processes the request and forwards the request to the Sterling B2B Integrator. When the Sterling B2B Integrator receives the request, it invokes the SWIFTNet Server Adapter or SWIFTNet7 Adapter to process the request.

Configuring the Server Application

You must appropriately configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to use the server application. See *SWIFTNet Server Adapter* or *SWIFTNet7 Adapter*.

Sterling B2B Integrator Acting as Third Party

In SWIFT 6.1 and higher, the Sterling B2B Integrator can act as a Third Party either in T-Copy mode or Y-Copy mode. As a Third Party, SWIFT forwards the copied request messages to the Sterling B2B Integrator. When the copied request is received, based on the Service Name specified in the message, the Sterling B2B Integrator searches the SWIFTNet Copy Service Profile to find out whether the Sterling B2B Integrator must act in T-Copy or Y-Copy mode.

For SWIFTNet 6.x, if Sterling B2B Integration must act in Y-Copy mode, Enhanced Header Information is extracted from the copied request, and is inserted into the mailbox. The message name for the mailbox is in the following format:

ThirdParty_[CopySnFRef]

In SWIFTNet 7, SWIFT supports 2 Copy Type: Header or Full. So, it depends on the Copy Type. If Copy Type = Header, the Enhanced Header Information is extracted and put into the mailbox. However, if Copy Type = Full, the whole request XML is put in the mailbox (because if Copy Type = Full, the Enhanced Header Info does not need to exist).

You can create a mailbox routing rule to evaluate this mailbox and bootstrap a custom business process to make the decision whether to authorize as a third party (**Third Party Authorised** option) or refuse (**Refuse** option) the copied request based on the Enhanced Header Information or the full payload that has been received (depends on the Copy Type Header or Full).

For SWIFTNet 6.x, by default, Sterling B2B Integrationsupplies a predefined business process for this notification process named SWIFTNet3rdPartyClientNotification. This is a sample business process for SWIFTNet 6.x. In SWIFTNet 7.0, the application does not provide a sample approval/refusal process but you may use SWIFTNet Client Profile feature to approve/refuse manually.

For SWIFTNet 6.x, this predefined business process is based on Transaction Count Enhanced Header Info. In this business process , the third party authorizes if the Transaction Count is less than five, and refuses if the Transaction Count is greater

than or equal to five. The decision is then made by the translator using the
SWIFTNet3rdPartyClientTxsCntr map. This is the predefined business process:

By default, Sterling B2B Integrator supplies a predefined business process for this
notification process named **SWIFTNet3rdPartyClientNotification**. This predefined
business process is based on Transaction Count Enhanced Header Info. In this
business process , the third party authorizes if the Transaction Count is less than
five, and refuses if the Transaction Count is greater than or equal to five. The
decision is then made by the translator using the **SWIFTNet3rdPartyClientTxsCntr**
map. This is the predefined business process:

Note: The notification message is a system message, and it should be sent using
Service Name **swift.snf.system**. To test this business process, the service name that
you should use is **swift.snf.system!x**. You must change this name in the business
process depending on whether you are testing it.

```
<process name="SWIFTNet3rdPartyClientNotification">
  <rule name="NoMoreMessage">
    <condition>number(msgCount/text()) &gt; number(msgTotal/text())</condition>
  </rule>
  <sequence name="NotificationFlow">
    <assign to="msgCount">1</assign>
    <assign to="msgTotal" from="count(RoutingRequest/RoutingRequest/MessageId)"/>
    <choice name="ForEachMessage">
      <select>
        <case ref="NoMoreMessage" negative="true" activity="DoNotification"/>
      </select>
      <sequence name="DoNotification">
        <assign to="msgId"
from="RoutingRequest/RoutingRequest/MessageId[number(//ProcessData/msgCount/text())]
/text()"/>
        <operation name="Extract HeaderInfo from Mailbox">
          <participant name="MailboxExtractBegin"/>
          <output message="MailboxExtractBegin_In">
            <assign to="CommitNow">YES</assign>
            <assign to="MessageId" from="msgId/text()"/>
            <assign to="." from="*"/>
          </output>
          <input message="MailboxExtractBegin_Out">
            <assign to="." from="*"/>
          </input>
        </operation>
        <operation name="Using Map">
          <participant name="Translation"/>
          <output message="Xlate_In">
            <assign to="map_name">SWIFTNet3rdPartyClientTxsCntr</assign>
            <assign to="validate_input">NO</assign>
            <assign to="validate_output">NO</assign>
            <assign to="output_to_process_data">YES</assign>
            <assign to="." from="*"/>
          </output>
          <input message="Xlate_Out">
            <assign to="." from="OUTPUT/node()"/>
          </input>
        </operation>
        <operation name="set user token">
          <participant name="SetUserToken"/>
          <output message="SetUserTokenMessage">
            <assign to="USER_TOKEN">admin</assign>
            <assign to="." from="*"/>
          </output>
          <input message="inmsg">
            <assign to="." from="*"/>
          </input>
        </operation>
```

```
          <operation name="SendNotification">
            <participant name="SWIFTNetClientService"/>
            <output message="handleClientRequest">
              <assign to="." from="*"/>
              <assign to="serviceName">swift.snf.system!x</assign>
              <assign to="thirdPartyAuth">TRUE</assign>
              <assign to="MessageName" from="/ProcessData/MessageName/text()"/>
            </output>
            <input message="testing">
              <assign to="." from="*"/>
            </input>
          </operation>
          <onFault>
            <sequence name="SWIFTClientOnFault">
              <operation name="set user token">
                <participant name="SetUserToken"/>
                <output message="SetUserTokenMessage">
                  <assign to="USER_TOKEN">admin</assign>
                  <assign to="." from="*"/>
                </output>
                <input message="inmsg">
                  <assign to="." from="*"/>
                </input>
              </operation>
              <operation name="RefuseDueToTechnicalIssue">
                <participant name="SWIFTNetClientService"/>
                <output message="handleClientRequest">
                  <assign to="." from="*"/>
                  <assign to="serviceName">swift.snf.system!x</assign>
                  <assign to="thirdPartyAuth">TRUE</assign>
                  <assign to="AuthDecision">Refused</assign>
                  <assign to="RefuseReason">Error sending original
notification.</assign>
                  <assign to="MessageName" from="/ProcessData/MessageName/text()"/>
                </output>
                <input message="testing">
                  <assign to="." from="*"/>
                </input>
              </operation>
            </sequence>
          </onFault>
          <onFault code="Mailbox Message Is Not Extractable">
            <sequence name="MoveOnToNextMessage">
              <assign to="msgCount" from="msgCount+1"/>
              <repeat ref="ForEachMessage"/>
            </sequence>
          </onFault>
          <assign to="msgCount" from="msgCount+1"/>
          <repeat ref="ForEachMessage"/>
        </sequence>
      </choice>
    </sequence>
</process>
```

When you send the notification message, the third party must also use the
SWIFTNet Client or SWIFTNet7 Client service and must specify at least three
parameters as follows:

- **thirdPartyAuth** must to be set to TRUE to send notification message
- **AuthDecision** must contain a valid value (either Authorised or Refused)
- **MessageName** is the message name for the mailbox

  Note: When you call the Mailbox Extract service to extract the message, this
  name should be available in ProcessData. If the extraction of the message from
  mailbox and sending the notification are configured by the same business

process (as was done in this predefined business process), you do not need to include the MessageName. However, if you use a more complex business process that invokes other Business Process, you must be careful not to lose the original MessageName.

When you send the notification message, the third party may also specify three optional parameters as follows:

- **ToSndrInfo** - When the decision is Authorised, this parameter contains the information that the third party wants to convey to the original sender of the request.
- **ToRcvrInfo** - When the decision is Authorised, this parameter contains the information that the third party wants to convey to the original receiver of the request.
- **RefuseReason** - When the decision is Refused, this parameter contains the refusal reason that the third party wants to convey to the original sender of the request.

SWIFTNet Server Installation

The SWIFTNet Server installation consists of a sequence of related tasks. This table outlines the process flow you must follow to install the SWIFTNet Server:

Task Number	Description	For more information
1	Configure SAG/SNL.	"Configure SAG/SNL" on page 119
2	Install and configure the SWIFTNet remote API.	"Install the SWIFTNet Remote API (RA)" on page 121
3	Install OpenSSL software on the SWIFTNet Server host. **Note:** Optional—this is only required if you wish to enable SSL support between the Sterling B2B Integrator and SWIFTNet Server.	OpenSSL libraries might be included with your operating system, although they may be optionally installed. Please check the appropriate documentation for your operating system to ensure the OpenSSL software is installed.
4	Install the SWIFTNet Server.	"Install the SWIFTNet Server" on page 127
5	Configure SSL between the Sterling B2B Integrator and the SWIFTNet Server. **Note:** Optional—this is only required if you wish to enable SSL support between the Sterling B2B Integrator and SWIFTNet Server.	"Configure SSL Between the Sterling B2B Integrator and the SWIFTNet Server" on page 123
6	Install the Command Line Adapter 2 Client.	"Start the Command Line Adapter 2 Client" on page 129
7	Configure and enable the SWIFTNet Server Adapter or SWIFTNet7 Adapter and start the SWIFTNet Server. **Note:** To monitor the status of the SWIFTNet Server, you need to select **Show Advanced Status** when you configure the SWIFTNet Server adapter or SWIFTNet7 Adapter.	*SWIFTNet Server Adapter* or *SWIFTNet7 Adapter*

Task		
Number	Description	For more information
8	Configure and enable the SWIFTNet Client Service.	*SWIFTNet Client Service*
9	Configure the SWIFTNet routing rule.	SWIFTNet Routing Rule
10	Configure Mailboxes (only if you are executing store-and-forward).	Using Mailboxes
11	Configure the SWIFTNetClient business process.	SWIFTNetClient Business Process

Prerequisites for using SWIFTNet Server

The following prerequisites must be met for the SWIFTNet Server to operate:

- The Command Line Adapter 2 client (CLA2Client) must be running to receive commands from the Sterling B2B Integrator to start and stop the SWIFTNet Server.
- The Command Line Adapter 2 client (CLA2Client) must be deployed on the same machine as the SWIFTNet Server.
- SWIFTNet Remote API (RA) must be installed on the same machine as the SWIFTNet Server.
- The SAG/SNL installed and configured with appropriate message partners and endpoints. See "Configure SAG/SNL."
- You must have a SWIFTNet Subscription for the InterAct and/or FileAct protocols.
- You use the same account to install the SWIFTNet Server as you used for the SAG/SNL installation.
- You have the correct Java JDK version installed.
- You installed the SWIFTNet Server on the appropriate operating system and version, according to the Sterling B2B IntegratorSystem Requirements.

Configure SAG/SNL
About this task

Complete the following steps to configure SAG/SNL for use with the SWIFTNet Server:

Procedure
1. Log in as an administrator to the SWIFTAlliance Workstation.
2. Go to **Gateway Admin - Application Interface** and create the client and server message partners.

   Note: The client (type = Client) and server (type = Server) message partner names must match the names in the SWIFTNet Server configuration (**<SagMessagePartnerClientName>** and **<SagMessagePartnerServerName>**).
3. In the Application Interface module, for the server message partner, configure the parameters as follows:

   Parameter

   Configuration

Name Name from the SWIFTNet Server configuration (<SagMessagePartnerServerName>).

Type Server

Status Enabled

Unit None

Host Adapter
Remote API Host Adapter

Supported Message Formats
Select **Strict SNL Format**.

Additional Processing
Select **Remote API Host Adapter**.

4. In the Application Interface module, for the client message partner, configure the parameters as follows:

Note: The Application Interface must be started.

Parameter
Configuration

Name Name from the SWIFTNet Server configuration (<SagMessagePartnerClientName>).

Type Client

Status Enabled

Unit None

Default Message Format for Emission (from Message Partner)
Strict SNL Format

Note: Strict SNL Format is required by the API.

Supported Message Formats
Select **Strict SNL Format**.

Note: Strict SNL Format is required by the API.

Additional Processing

Note: Do not select any additional processing options.

5. In the Endpoints module, for the server message partner, configure the endpoint parameters as follows to define where to route the messages:

Parameter
Configuration

Name Name from the SWIFTNet Server configuration.

Destination
Application Interface:<Name from the SWIFTNet Server configuration>

Status Enabled

6. In the Endpoints module, for the server message partner, configure the routing detail criteria appropriately for your installation:

Parameter
Configuration

From   SWIFTNet Interface

Sequence

>   **Note:** This is the sequence number.

Name   Name for this Endpoint configuration.

Status   Enabled

SNL Endpoint
>   None

Service Name
>   None

Request Type
>   None

Requestor DN
>   None

Responder DN
>   None

Traffic Type
>   None

Delivery Mode
>   None

Priority
>   None

7. In the Endpoints module, for the server message partner, configure the destination detail parameters as follows:

Parameter
>   **Configuration**

Interface
>   Application Interface

Application
>   Name of the Server Message Partner specified in SWIFTNet Server configuration.

Mode   Strict

Install the SWIFTNet Remote API (RA)
About this task

You need to install the SWIFTNet Remote API on the machine on which the SWIFTNet Server will be installed. This is the software distributed by SWIFT, the API that the SWIFTNet Server uses to connect to the SWIFTNet SAG/SNL instance to link into a SAG. Remote API is software supplied by SWIFT that establishes a communication link with the RA Host adapter component of the SWIFTAlliance Gateway (SAG), either from a SWIFTNet application existing on a remote machine or from a SWIFTNet application existing on the machine where SAG is installed. RA offers two sets of APIs: SWIFTNet Link-specific APIs and SWIFTAlliance Gateway-specific APIs.

Before you install the SWIFTNet Remote API (RA), verify that you have performed the following:

- The SWIFTAlliance Gateway software is installed on the machine you wish to connect, with the RAHA option checked in the licensing screen. Otherwise you will only be able to install the RA locally on the SAG system.
- You know the host name or IP address of the machine on which the SWIFTAlliance Gateway software is installed.
- You know the port number used by SAG to communicate with its remote applications. This is the port number specified during the SAG installation, and its default value is 48002.
- You have 300MB of free disk space as working space is available on the machine where you will install the RA software.
- The patch level requirements in the Release Letter from SWIFT are respected. The same patch requirements apply for both the machines on which RA and/or SAG are installed. This is particularly important for Java patches.

The following requirements apply to UNIX systems only:
- The Remote API installer is graphical and should be run using a dedicated UNIX X display terminal. If this is not possible or desirable, run the installer using a PC based X server such as Exceed, PC-Xware, XManager or Reflection-X. It is important that the PC X server is configured to run in single window mode and uses XDM to start a UNIX desktop session such as CDE. Failure to comply with these requirements may cause the installer not to display correctly. The method used to configure it in single window depends on your product. If you are using Exceed software, select Window Mode in the Xconfig module and toggle the mode from Multiple to Single.
- The home directory must have at least 60 MB of free space.
- On AIX and Solaris operating systems, if the disk space requirements for the temporary files for the install program cannot be satisfied, the installer option **-is:tempdir <TMPDIR>** can be used to specify an alternate temporary directory.

Complete the following steps to install the RA:

Note: See the *SWIFT Remote Developers Toolkit Installation Guide* for complete installation instructions.

Procedure

1. Install the remote API on the machine where you will install SWIFTNet Server.
2. Configure the RA to point to the SAG instance you will be accessing, or if the RA is already configured, verify that it points to the correct SAG instance.
3. If you are installing on the Windows operating system, add **SYSTEM** to the Security settings, allow SYSTEM **Full Control** and select **Allow inheritable permissions from parent to propagate to this object**. See the documentation for the SWIFTNet Remote API for more information.
   a. Right-click and select **Properties** on <SWIFT RA API installdir>.
   b. Select the **Security** tab.
   c. Click **Add** to and select **SYSTEM**.
   d. Allow **Full Control** to **SYSTEM**.
   e. Select the **Allow inheritable permissions from parent to propagate this object** option.
   f. To confirm, navigate to the <applicationinstalldir\SWIFTAlliance\RA\lib> directory, and right-click and select **Properties**.
   g. Select the **Security** tab.

h. Verify that SYSTEM has Full Control permissions for the <applicationinstalldir\SWIFTAlliance\RA\lib> directory.

4. If you are installing on a Windows operating system, following the SWIFT RA installation, you will need to set the PATH system environment variables:

```
PATH : append <Swift RA API installdir>\bin;<Swift RA API installdir>\lib
SWNET_HOME : <Swift RA API installdir>
SWNET_CFG_PATH : <Swift RA API installdir>\Ra1\cfg;
```

Configure SSL Between the Sterling B2B Integrator and the SWIFTNet Server

About this task

Note: Optional—this is only required if you wish to enable SSL support between the Sterling B2B Integrator and SWIFTNet Server.

To configure SSL for the SWIFTNet Server, you must complete the following tasks:

Task Number
> **Description**

1   Prepare the SSL certificates for the SWIFTNet Server.

2   Prepare the SSL certificates for the Sterling B2B Integrator.

3   Configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter.

4   Configure the SWIFTNet HTTP Server adapter.

5   Configure the SWIFTNetClient business process. This step is not required for SWIFTNet7 configurations.

6   Configuring the SSL Setup on the SWIFTNet Server

Prepare the SSL Certificates for the SWIFTNet Server
About this task

To prepare the SSL certificates for use with the SWIFTNet Server, complete the following:

Procedure
1. Create the keyfile that contains the private key for MEFGCommServer.

   You can use OpenSSL to generate the keyfile and certificate request file. Then, you can use this certificate request to ask the CA to generate the certificate and sign it for you.

   Note: When you use OpenSSL to generate the keyfile, you will be prompted to type in the password to protect the keyfile.

   Note: Please take note of this password for later use. (step 8 of *Configuring the SSL Setup on the SWIFTNet Server*).

2. Import the signed certificates into the Sterling B2B Integrator CA repository.

3. Note the CA Certificate ID and CA Certificate Name because you will need to use them in the SWIFTNetClient business process.

4. Create the CA Certificate that contains the cert (public key) for the Sterling B2B Integrator side of this configuration.

Note: You can only do this after you have completed step 3 of *Preparing the SSL Certificates for the Sterling B2B Integrator.*

Prepare the SSL Certificates for the Sterling B2B Integrator
About this task

To prepare the SSL certificates for use with the Sterling B2B Integrator, complete the following:

Procedure

1. Create a self-signed certificate on the Sterling B2B Integrator for the System certificate, including the following:
   - Select **Set Certificate Signing Bit**.
   - The name of this certificate must be the name of your server/domain so that Open SSL can properly validate the certificate.

     Note: Alternatively, you can generate a certificate signing request using the Certificate Wizard, and ask a CA to sign your certificate. If you choose this option, include the following:
     - Ensure your common name for the certificate matches the correct server name/domain name.
     - Check in the key and the certificate (after the certificate is signed and returned by CA) to System Certificate.

2. When you create and check in the certificate, note the certificate name. This name will be used when you configure the SWIFTNet HTTP Server adapter (step 4 below).

3. Export the public key of the certificate you generated above. This public key will be used by the SWIFTNet Server (that you created in "Prepare the SSL Certificates for the SWIFTNet Server" on page 123) as CA certificate file (trusted list).

4. Configure the SWIFTNet HTTP Server adapter to use SSL and choose the System certificate you generated in step 1 of this procedure.

   Note: Note the port number of the SWIFTNet HTTP Server adapter because this number must match the port you configure for the SWIFTNet Server Adapter or SWIFTNet7 Adapter (below).

Configuring the SWIFTNet Server Adapter or SWIFTNet7 Adapter for SSL
About this task

To configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter for SSL, complete the following:

Procedure

1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet Server adapter or SWIFTNet7 Adapter or select it from the list and click **Go!**.
3. Click **Edit**.
4. For **SI HTTP Server Adapter Port**, use the SSL port configured for the SWIFTNet HTTP Server adapter (above).

5. For **SI Server IP**, type your exact server name/domain name. This must match with the server/domain name and also match the system certificate name you created in "Prepare the SSL Certificates for the Sterling B2B Integrator" on page 124.

6. Ensure that **Use SSL** is set to **True**.

7. For CA Certificate, select the appropriate CA certificate. This is the certificate you imported in step 2 of *Preparing the SSL Certificates for the SWIFTNet Server*.

Configure the SWIFTNet HTTP Server Adapter or SWIFTNet7 HTTP Server Adapter for SSL

About this task

To configure the SWIFTNet HTTP Server adapter or SWIFTNet7 HTTP Server adapter for SSL, complete the following:

Procedure

1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet HTTP Server adapter or SWIFTNet7 HTTP Server adapter, or select it from the list and click **Go!**.
3. Click **Edit**.
4. Ensure that **Use SSL** is set to **Must**.
5. For **System Certificate**, select the appropriate system certificate.

Configure the SWIFTNet Client Service or Business Process for SSL

About this task

Note: These steps are not required for SWIFTNet7 configurations.

You must either configure the SWIFT Client service through the Sterling B2B Integrator user interface or through the business process you create for the service.

To configure the SWIFTNet Client service for SSL, complete the following:

Procedure

1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet Client service or select it from the list and click **Go!**.
3. Click **Edit**.
4. Ensure that **Use SSL** is set to **Must**.
5. For **CA Certificate**, select the appropriate CA certificate. This is the certificate you imported in step 2 of "Prepare the SSL Certificates for the SWIFTNet Server" on page 123.

   Alternatively, to configure the SWIFTNetClient business process, add the following to the BPML to ensure the SSL configuration for the SWIFTNet HTTP Client adapter is included:

   Note: The **bold** lines indicate information that you need to modify to match your installation.

   ```
   <<operation>
        <participant name="SWIFTNetClientService"/>
        <output message="handleClientRequest">
   ```

```
            <assign to="." from="*"></assign>
            <assign to="interfaceMode">interact</assign>
            <assign to="swiftOp">sync</assign>
            <assign to="requestorDN">o=ptscfrnn,o=swift</assign>
            <assign to="responderDN">o=ptscfrnn,o=swift</assign>
            <assign to="serviceName">swift.generic.ia!x</assign>
            <assign to="SnF">FALSE</assign>
            <assign to="nonRepudiation">FALSE</assign>
            <assign to="possibleDuplicate">FALSE</assign>
            <assign to="deliveryNotification">FALSE</assign>
            <assign to="UseSSL">TRUE</assign>
            <assign to="CipherStrength">All</assign>
      <assign to="CACertId">000.00.000.00:00000:10f3202f455:4337</assign>
        </output>
        <input message="testing">
          <assign to="." from="*"></assign>
        </input>
      </operation>
```

Note: The **CACertId** must match the CA Cert ID you configured for the
SWIFTNet Server (step 2 of "Prepare the SSL Certificates for the SWIFTNet
Server" on page 123).

Configure the SSL Setup on the SWIFTNet Server
About this task

Prior to completing the next steps, ensure that you have completed the tasks in
"Configure SSL Between the Sterling B2B Integrator and the SWIFTNet Server" on
page 123.

Procedure

1. Type the following command to change to the directory where the SWIFTNet
   Server SSL Utilityjar is located:

   ```
   cd <SWIFTNet Server installdir>/bin
   ```

2. Type the following command to invoke the configuration script:

   ```
   dir *.jar
   ```

   You will see the sslUtil.jar file.

3. Type the following command:

   ```
   java -jar sslUtil.jar
   ```

4. Type the full path for the SWIFTNet Server home directory (the directory in
   which you just installed the SWIFTNet Server) and press **Enter**. You are
   prompted to confirm the directory.

5. Type **Yes** and press **Enter** to confirm. The configuration script verifies the
   structure of the directory to ensure that it is the valid installation directory for
   the SWIFTNet Server. If the directory is not valid (for example, the bin
   directory is missing from the path), you are prompted to retype the valid
   directory. Once a valid directory is entered, you proceed with the SSL
   configuration.

6. Type the full path to the private key location and press **Enter**. This key file
   belongs to the SWIFTNet Server and will used during the SSL "handshake."

   Note: The key file contains your private key.

7. Type **Yes** and press **Enter** to confirm the path.

8. Type the correct password to access the key file and press **Enter**.

9. Type the password again to confirm it and press **Enter**.

10. Type the full path to the CA Certificate location and press **Enter**. This CA Certificate file contains the trusted certificates that are used during the SSL "handshake."

11. Type **Yes** and press **Enter** to confirm the path. The configuration completes and displays a message that the SSL configuration process has finished.

Install the SWIFTNet Server
About this task

Complete the following steps to install the SWIFTNet Server:

Note: The installation script and the binary install are located on the Sterling B2B Integrator installation DVD.

Procedure

1. After installing the Sterling B2B Integrator, log on to your UNIX or Windows system using the same account as the one you used to install and configure SWIFTNet RA.

2. If you are installing on UNIX system with X-Windows, please ensure that you configure the X-Windows so the installation user interface can be displayed correctly. If X-Windows environment is not available, the installation reverts to the command line interface.

3. Download the SCIInstallWizard_swiftnet.jar and swiftnet_[version].jar files from the installation DVD to the server where the SWIFTNet Server will be installed.

   Note: Replace [version] in the command above with the value from the list that matches the patch you are installing.

4. Type the following command to invoke the installation script:

   ```
   java -jar SCIInstallWizard_swiftnet.jar
   ```

5. Press **Enter**.

6. On the SWIFTNet Server Installer page, click **Next**.

7. Type the jdk directory which you want to use or click **Select Folder** to choose the jdk directory.

8. Type the destination directory where you want to install the SWIFTNet Server or click **Select Folder to choose the installation directory.**

9. If you are installing on a Windows operating system, select the **Install on Windows** check box.

10. Click **Next**. If the directory does not exist, you are prompted to click **Yes** to create it.

11. If you are installing on the Windows operating system, you are required to enter the account user name in the format **DomainName\Username**. If it is a local user, type **.\Administrator**.

12. On the Install Jar page, click **Select Files** to specify the path to the swiftnet_[version].jar installation jar.

13. If you want to install SSL, select the **Install SSL** check box.

14. Click **Next**.

15. If you chose to install SSL, the SSL configuration page is displayed. Complete the necessary parameters and click **Next**.

Note: The key file contains the private key to be used during the SSL
handshake.

16. On the Install SWIFTNet Adapter Components page, you can view a
    summary of the installation steps that will be performed.

17. Click **Next**.

    The installation script copies files from the installation jar to the destination
    directory and verifies that the correct number of files and blocks are copied. If
    the destination directory does not have enough free disk space, the script
    suggests you delete enough files to provide the necessary disk space and then
    exits the installation. During the installation process, you can click **Show
    Details** to view detailed information about the progress of the install.

    You are notified that the installation is complete with the following message:
    Installation Finished.

    On a Windows system, this installs the following service instances for
    SWIFTNet 6.x configurations (you can verify this by checking **Control Panel** >
    Administrative tools > **Services**):

    - MEFGCommServerService1
    - MEFGSwiftnetServerService1_Instance1
    - MEFGSwiftnetServerService1_Instance2
    - MEFGCommSSLServerService1
    - MEFGSwiftnetSSLServerService1_Instance1
    - MEFGSwiftnetSSLServerService1_Instance2
    - MEFGMonServerService1

    Note: On a Windows system, this installs the following service instances for
    SWIFTNet7 configurations (you can verify this by checking Control Panel >
    Administrative tools > Services): MEFGCommServer7Service1,
    MEFGCommSSLServer7Service1, MEFGMonServerService1,
    MEFGSwiftnetServer7Service1_Instance1,
    MEFGSwiftnetServer7Service1_Instance2,
    MEFGSwiftnetSSLServer7Service1_Instance1, and
    MEFGSwiftnetSSLServer7Service1_Instance2.

    Note: If you have already installed any of these services, you will be
    notified through an error message containing Error Code 1073 that the
    service or services are already installed. If this occurs, you can unregister
    the services as follows and then re-install the SWIFTNet Server again:

    – Go to <MEFG Server directory>\bin.

    – Double-click **unregister_svcs.cmd** to unregister all the related services.
      Or, alternatively, you can unregister each of the services manually by
      typing the following and pressing **Enter** after each line:

      To uninstall the MEFG Comm Server Service, type `MEFGCommServer.exe`
      `-u`

      To uninstall the MEFG SWIFTNet Server Service Instance 1, type
      `MEFGSwiftnetServer.exe -u s1`

      To uninstall the MEFG SWIFTNet Server Service Instance 2, type
      `MEFGSwiftnetServer.exe -u s2`

      To uninstall the MEFG Comm SSL Server Service, type
      `MEFGCommSSLServer.exe -u`

      To uninstall the MEFG SWIFTNet SSL Server Service Instance 1, type
      `MEFGSWIFTNetSSLServer.exe -u s1`

To uninstall the MEFG SWIFTNet SSL Server Service Instance 2, type `MEFGSWIFTNetSSLServer.exe -u s2`

For SWIFTNet 7, to uninstall the MEFG Comm Server Service, type MEFGCommServer7.exe -u

For SWIFTNet 7, to uninstall the MEFG SWIFTNet Server Service Instance 1, type MEFGSwiftnetServer7.exe -u s1

For SWIFTNet 7, to uninstall the MEFG SWIFTNet Server Service Instance 2, type MEFGSwiftnetServer7.exe -u s2

For SWIFTNet 7, to uninstall the MEFG Comm SSL Server Service, type `MEFGCommSSLServer7.exe -u`

For SWIFTNet 7, to uninstall the MEFG SWIFTNet SSL Server Service Instance 1, type `MEFGSWIFTNetSSLServer7.exe -u s1`

For SWIFTNet 7, to uninstall the MEFG SWIFTNet SSL Server Service Instance 2, type `MEFGSWIFTNetSSLServer7.exe -u s2`

18. If you are installing on a Windows operating system, you must add the following under the System variables (add the <MEFG installdir>\bin directory to the PATH environment variable just before the <SWIFT RA API installdir>\bin entry):

   - PATH : append <MEFG installdir>\bin;

     Note: Insert this PATH variable *before* the SWIFT RA API installdir entries you added in step 4 of "Install the SWIFTNet Remote API (RA)" on page 121 (the PATH variable must appear ahead of the library references in the System variable list).

   - Allow the defined user to start MEFGSwiftnetServer and MEFGCommServer through the following steps:
     – Select **Control Panel** > **Administrative Tools** > **Local Security Settings**.
     – Select **Local Policies** > **User Rights Assignment**.
   - Double-click **Log on as a service** and assign the account user (that you entered during the installation process) to this setting.

Configuring Fail-over Processing Using the SWIFTNet Server

To set up the SWIFTNet Server in a dual-active SAG configuration for fail-over processing, specify the following application interface definitions for the SWIFTNet Server Adapter or SWIFTNet7 Adapter:

- active-active Configuration (or failover-SAG for the SWIFTNet7 Adapter)
- RA1 definitions for primary SAG (s1)
- RA2 definitions for alternate SAG (s2)

Note: Certificates and profiles must be available on the SAG where they are used. For fail-over processing, Puts and Gets try to connect to the first SAG specified for s1 and s2. If the connection fails, the Put and Get try to connect to the next SAG. If this connection also fails, the cycle is repeated if retry has been enabled.

Start the Command Line Adapter 2 Client
About this task

The Command Line Adapter 2 client (CLA2Client) must be installed and run on a remote server. Complete the following steps to start the remote adapter implementation version of the Command Line Adapter 2:

Procedure

1. Locate the client jar (CLA2Client.jar) in your Sterling B2B Integrator installation that contains the necessary classes.
2. Move the client jar to the machine that will be running the remote Command Line Adapter 2 client.

    Note: This is the machine on which the SWIFTNet Server is installed.
3. Start the remote adapter implementation using the following command:

    `[path to java bin]/java -jar [path to CLA2 Client jar file]/CLA2Client.jar <port> [debug]`

    Note: The port (above) will be used when you configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter.

    This is an example of the command to start the remote adapter implementation:

    `jdk1.5.0_11/bin/java -jar CLA2Client.jar 15699 debug`

    Note: The [debug] option is not required, but you may find it helpful. When you upgrade the Sterling B2B Integrator, you will also need to obtain the corresponding new CLA2Client.jar file to avoid receiving a ClassConflict error.

Monitor the Status of the SWIFTNet Server

About this task

To monitor the status of the SWIFTNet Server, you need to select **Show Advanced Status** when you configure the SWIFTNet Server Adapter SWIFTNet7 Adapter:

Procedure

1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet Server Adapter SWIFTNet7 Adapter, or select it from the list and click **Go!**.

    Note: When you select the SWIFTNet Server Adapter SWIFTNet7 Adapter, make sure you also select the **Show Advanced Status** check box prior to clicking **Go!**. This enables you to view the Advanced Status column on the Services Configuration page to see whether the SWIFTNet Server is stopped or started.
3. Click **Edit**.
4. Specify field settings in the Admin Console. See *SWIFTNet Server Adapter* or *SWIFTNet7 Adapter*.
5. On the Confirm page, verify that the **Enable Service for Business Processes** check box is selected. This enables the adapter instance.

Start and Stop the SWIFTNet Server

About this task

To start and stop the SWIFTNet Server:

Procedure

1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet Server Adapter or SWIFTNet7 Adapter, or select it from the list and click **Go!**.

Note: When you select the SWIFTNet Server Adapter or SWIFTNet7 Adapter, make sure you also select the **Show Advanced Status** check box prior to clicking **Go!**. This enables you to view the Advanced Status column on the Services Configuration page to see whether the SWIFTNet Server is stopped or started.

3. Once the SWIFTNet Client adapter is configured and saved, click the **Enabled** check box on the Services Configuration page. This starts the SWIFTNet Server.

   Note: To stop the SWIFTNet Server, clear the **Enabled** check box on the Services Configuration page

Monitor the SWIFTNet Server Queues

About this task

To monitor the status of the SWIFTNet Server:

Procedure

1. Select **Operations** > **SWIFTNet Monitor**.
2. To automatically refresh the display, ensure that the **Automatically refresh every 15 seconds** option is selected.
3. View the status of the SWIFTNet Server queues.

   Note: The status is only populated after you configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter and start it.

   Note: Click **Monitor Input Channels** if you want to monitor the input channels.

Chapter 8. SWIFTNet Input Channel

The SWIFTNet 6.1 release introduced the concept of an input channel. Currently, the Sterling B2B Integrator only supports the use of input channel for InterAct messages in store-and-forward (SnF) mode. An input channel can be used by the messaging interface to establish an input session with SWIFT. The input session starts when the messaging interface opens the input channel and ends when the messaging interface closes the input channel. The input channel also supports sender-to-receiver first-in-first-out (FIFO), which means that each message is delivered only one time and minimizes the number of possible duplicates.

During such an input session, each message that is sent using this input channel is assigned a sequence number. This numbering is used by SWIFT to detect messages that are out of sequence, gaps in a sequence, and duplicate messages.

So messages are delivered in the same sequence that they are sent, SWIFT only delivers messages when all messages with a lower sequence number have been acknowledged by SWIFT. When the acknowledgement on a message is lost, the messaging interface must resend the message with the original sequence number. This allows duplicate messages to be more easily identified and avoided by SWIFT.

SWIFT automatically provides a default input channel for each user (BIC8) who is subscribed to at least one service working in store-and-forward mode. You can optionally create additional input channels when you have two or more messaging interfaces that operate independently at the same time, because the input channel can only be used by one messaging interface at a time.

How the Sterling B2B Integrator Supports the Use of Input Channels and Queues

To support the use of input channel, the Sterling B2B Integrator includes the following functionality:

- Ability to create a new input channel
- Ability to delete an existing input channel
- Ability to open an input channel
- Ability to close an input channel
- **Ability to monitor input channel status**. The input channel status resides in an application database table (SWNET_CHANNEL) to allow the message traffic to be synchronized with SWIFT. Each operation related to the input channel checks this status before proceeding to the next step.
- **Support for local processing**. If a message was sent when the input channel is closed or down, the message is stored locally in a database. When the input channel is open again, it is resent to SWIFT. This local processing support is only available when the message is sent using the input channel.
- **An automatic process to resend messages.** The Sterling B2B Integrator provides a predefined scheduler business process to automate the resend process. This business process checks both unsent messages (stored in local processing) as well as unacknowledged messages to be resent when the input channel is available again.

- **An automatic process to resolve sequence gaps.** SWIFT only acknowledges messages when all messages that have a lower sequence number have been acknowledged by SWIFT. However, when there is a persistent error, it creates a gap because there is not an acknowledgement for a subsequent message number, and this error will eventually become a bottleneck. The only way to resolve this gap is by sending a Resolve Gap request to SWIFT. So, when the Sterling B2B Integrator resend attempts exceed the maximum resend attempts setting that you have configured, Sterling B2B Integrator automatically sends the Resolve Gap request to SWIFT to resolve the gap, so SWIFT will then continue processing with the subsequent sequence number.
- **Ability to automatic open the input channel when the SWIFTNet Server Adapter or SWIFTNet7 Adapter is started and closing the input channel when the SWIFTNet Server Adapter or SWIFTNet7 Adapter is stopped.** When the SWIFTNet Server is started, it triggers a callback to the Sterling B2B Integrator that bootstraps a predefined business process to open the input channel (if you have configured it to use input channel). During the shutdown process, the SWIFTNet Server is not stopped immediately. The request to shut down is delayed to ensure that there is no gap in the input channel and all requests has been acknowledged. Once all gaps are resolved, the Sterling B2B Integrator automatically triggers a predefined business process to close the input channel and then automatically disables the SWIFTNet Server Adapter or SWIFTNet7 Adapter and shuts down the SWIFTNet Server.
- **Automatic synchronization after the input channel is opened (in force mode) and before the input channel is closed, to ensure all gaps are resolved and each sequence number is acknowledged.** This also ensures that the messaging interface assigns the correct sequence number for new messages each time a new input session begins.
- **Automatic traffic throttling, which is used in conjunction with the input channel status monitoring to provide an automatic switch to local processing mode when the message traffic has exceeded SWIFT capacity**. The resend scheduler business process handles the resend operation of the unsent messages when the capacity resume to normal.

Support for Local Processing

If a message is sent when the input channel is closed, down, or during synchronization, the message is temporarily stored locally in the Sterling B2B Integrator database. When the input channel is open, the message is resent to SWIFT.

Note: This local processing support is only available when the message is sent using the input channel. Input processing is not supported in a clustered environment.

The SWIFTNet Client or SWIFTNet7 Client service indicates if a message is stored locally in the Advance Status column when you view the service instance.

Automatic Process to Resend a Message

The resend scheduler automatically handles the resend process for both unacknowledged messages and unsent messages (from local processing) when the input channel status allows it. The resend scheduler invokes the predefined SWIFTNetClientResend or SWIFTNet7ClientResend business process to perform the resend of the message.

You do not need to supply any parameters to the predefined
SWIFTNetClientResend or SWIFTNet7ClientResend business process because all
necessary information is passed dynamically by the resend scheduler. Therefore, all
resend processes are handled automatically.

Automatic Process to Resolve a Sequence Gap

SWIFT only acknowledges messages when all messages with a lower sequence
number have already been accounted for (acknowledged) by SWIFT. However,
when there is a persistent error, this creates a gap (no acknowledgement for the
subsequent message number) and may eventually become a bottleneck. The only
way to resolve this gap is to send a Resolve Gap request to SWIFT.

When this occurs, the resend scheduler automatically sends a Resolve Gap request
when the retry count has exceeded the maximum retry attempts configured in the
SWIFTNet Server Adapter or SWIFTNet7 Adapter. The resend scheduler invokes
the predefined SWIFTNetClientResend or SWIFTNet7ClientResend business
process to send the Resolve Gap request to SWIFT.

You can track the Resolve Gap request in the SWIFTNet Correlation Search
Interface if you wish.

Input Channel Status

Input channel operation relies heavily on the input channel status. Certain statuses
trigger automatic operation while other statuses may prevent an operation from
occurring. The statuses maintain message traffic and synchronize with the SWIFT
queue mailbox. The input channel status is stored in the Sterling B2B Integrator
database table (SWNET_CHANNEL) and most input channel functions (all
functions except the create and delete operations) check the current value for input
channel status before proceeding. You can use the SWIFTNet Monitor to view the
status of the input channels. The following are the possible input channel status
values and their description:

Input Channel Status
> **Description**

OPENING
> This status occurs when the SWIFTNet Client or SWIFTNet7 service
> attempts to open the input channel. When this status appears, no other
> opening requests (for the same input channel name) are allowed. If the
> client is attempting to send the message, the message is saved locally. If
> there is a gap, the status will be updated to OPEN_SYNC. If there is no
> gap but there are any local messages that have not been sent, the status is
> updated to LOCAL and the Resend Message function is disabled. When all
> gaps are resolved, the status is updated to OPEN.

OPEN_SYNC
> This status occurs when SWIFT indicates that there is a gap that must be
> resolved by resending previously unacknowledged messages. Only the
> Resend Message function is allowed to operate at this time. If the client is
> trying to send the message, the message is saved locally during this time.
> After all gaps are resolved, if there are local messages, the status will be
> updated to LOCAL. If there are no local messages, the status is updated to
> OPEN.

OPEN This status occurs when the input channel is open and all gaps have been
> resolved. This status also occurs when there are no local messages that

have not been sent. Both the Send Message and Resend Message functions are allowed to send messages within the allocated window size. If the windows size has reached zero, the status is updated to SYNC.

SYNC This status occurs when the window size has reached zero and it is time for synchronization to occur. At this point, the resend handler resend any previous messages that have not been acknowledged and holds all new messages locally. Only the Resend Message function is allowed to operate at this time. If the client is trying to send a new message, the message is saved locally. After all gaps are resolved, if there are local messages, the status will be updated to LOCAL. If there are no local messages, the status is updated to OPEN.

LOCAL

This status occurs when after the gaps are all resolved (after the OPEN_SYNC or SYNC statuses have occurred), but there are new messages in local processing that have not sent. Only the Resend Message function is allowed to operate at this time. If the client is trying to send a new message, the message is saved locally. Then, if there are no other local messages, the status is changed to OPEN. However, if the window size reaches zero during this time, the status is first changed to SYNC.

CLOSE_SYNC

This status occurs when the SWIFTNet Client or SWIFTNet7 service is attempting to close the input channel (using a manual close) but there are gaps (messages that have not been acknowledged) in the Input Channel. Therefore, the status is updated to CLOSE_SYNC to resolve the gaps before closing. Only the Resend Message function is allowed to operate at this time. If the client is trying to send the message, the message is saved locally. When all gaps are resolved, the SWIFTNet Client or SWIFTNet7 service sends a Close Input Channel Request and the status is updated to CLOSING. When the CLOSING completes successfully, the status is updated to CLOSED.

CLOSE_DOWN

This status occurs when the user is attempting to disable the SWIFTNet Server Adapter or SWIFTNet7 Adapter. If there is a gap in the Input Channel, that gap is resolved before the SWIFTNet Server Adapter or SWIFTNet7 Adapter is disabled. Only the Resend Message function is allowed to operate at this time. If the client is trying to send the message, the message is saved locally. When all gaps are resolved, the SWIFTNet Client or SWIFTNet7 Client service sends a Close Input Channel Request (and the status is updated to CLOSING). When the CLOSING completes successfully, the status is updated to CLOSED and the SWIFTNet Server Adapter or SWIFTNet7 Adapter is disabled.

CLOSING

This status occurs when the SWIFTNet Client or SWIFTNet7 Client service is attempting to close the input channel. When this status appears, no other closing requests (for the same input channel name) are allowed. If the client is trying to send the message, the message is saved locally and the Resend Message function is disabled. When all gaps are resolved, the status is updated to CLOSED.

CLOSED

This status occurs when the input channel is closed. If the client is trying to send the message, the message is saved locally and the Resend Message function is disabled.

LOCKED

This status occurs when there is an error during the opening or closing of the input channel, or an error during a message send because the SWIFTNet Server or SAG is down.

Monitor the Input Channels
About this task

To monitor the status of the input channels:

Procedure
1. Select **Operations** > **SWIFTNet Monitor**.
2. To automatically refresh the display, select the **Automatically refresh every 15 seconds** option.
3. Click the **Monitor Input Channel** link to display the status of the input channels. When you open the SWIFTNet Monitor, the Monitor Queues interface is displayed.

   Note: The status is only populated after you configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter and start it.

Configuring the Sterling B2B Integrator Components to Use the Input Channel

You must configure the following three Sterling B2B Integrator components to use the input channel:
- SWIFTNet Server Adapter (or SWIFTNet7 Adapter if you are using SWIFTNet7)
- Resend Scheduler (or SWIFTNet7 Adapter Scheduler if you are using SWIFTNet 7)
- SWIFTNet Client or SWIFTNet7 Client Service (including configuring a predefined business process to create a new input channel)

Although the Resend Scheduler configuration is part of configuring the SWIFTNet Server adapter, we discuss it separately here to show its importance during the input channel operation.

Configure the SWIFTNet Server Adapter and Resend Scheduler for SWIFTNet 6.x
About this task

To configure the SWIFTNet Server adapter to use the input channel:

Procedure
1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet Server adapter or select it from the list and click **Go!**.
3. Click **Edit**.
4. Specify field settings in the Admin Console.

   Note: Specify failover processing to ensure that failover is supported if a SAG connection fails by configuring **Active-Active Configuration**. The business entities (accessible through the Business Entities wizard as part of the

SWIFTNet Client adapter configuration) are shared by both RA1 and RA2. The Business Entities wizard enables you to add multiple entities.

Field   Description

Name   Unique and meaningful name for the service configuration. Required.

Description
> Meaningful description for the service configuration, for reference purposes. Required.

Select a Group
> Select one of the options:
> - None – Do not include the configuration in a service group at this time.
> - Create New Group – Enter a unique name for a new group, which will be created with this configuration. (You can then add other services to the group as well.)
> - Select Group – If service groups already exist for this service type, they are displayed in the list. Select a group from the list.
>
> **Note:** Only select group if this adapter is clustered in a group.

SI Server IP
> The callback IP of Sterling B2B Integrator for the SWIFTNet Server. Required.
>
> **Note:** The default value is the IP address of the machine where the Sterling B2B Integrator is installed.

SI HTTP Server Adapter Port
> This is the listening port for the SWIFTNet HTTP Server Adapter. Required. The default populated value is the instance port number of the Sterling B2B Integrator instance plus 53. For example, if the Sterling B2B Integrator instance port is 34600, the listening port populated by default is 34653.
>
> **Note:** The HTTP Server adapter functions between the SWIFTNet Client adapter and the SWIFTNet Server. For an SSL connection, this value should be server name because the certificate is made with the server name.
>
> **Note:** If you are using the SWIFTNet Server adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

MEFG SWIFTNet IP
> The IP address of the SWIFTNet Server. Required.

MEFG SWIFTNet Port
> The port of the SWIFTNet Server. Required.

CLA2Client Listening Port
> The listening port used by the client command adapter (CLA2Client) running along the SWIFTNet Server. Required.

Note: This port listens for requests to start and stop the SWIFTNet Server.

MEFG SWIFTNet Home
The home directory of the SWIFTNet Server. Required.

Start MEFG SWIFTNet despite errors
Whether to start the MEFG Server if errors occur. The default is unchecked (do not start the SWIFTNet Server if errors occur). Optional.

Use SSL
Whether to enable Secure Socket Layer (SSL) over HTTP communication between the Sterling B2B Integrator and the SWIFTNet Server. Valid values are False (default) and True. Select **True** to use SSL with an Input Channel.

Note: When you select **True**, the SSL Details page is displayed. This page allows you to configure the SSL configuration for the Resend Scheduler and you must configure the parameters on this page. This is the client component and it requires client configuration, so you must configure SSL for the SWIFTNet Server adapter exactly the same as you configure for the SWIFTNet Client service. The duplicate configuration is necessary because the Resend Scheduler must be linked to a specific instance of the SWIFTNet Server adapter.

Cipher Strength
Specifies the strength of the algorithms (cipher suites) used to encrypt data. Valid values are:
- STRONG - Required if Use SSL is Must
- ALL - All cipher strengths are supported
- WEAK - Often required for international trade, because government regulations prohibit STRONG encryption from being exported

Default is ALL. Required if **Use SSL** is checked.

CA Certificate
Move one or more CA Certificates to the use column. These are the digital security certificates that the SSL server will use to authenticate the client. Required if SSL is selected.

Message Partner Client Name
The client message partner name that the SNL server application recognizes for the SWIFTNet Server client application.

Note: The Message Partner Client Name must correspond to the Application Interface Message Partner that is defined on the SAG as the client interface for the SWIFTNet Server.

Message Partner Server Name
The server message partner name that the SNL server application recognizes for the SWIFTNet Server server application.

Note: The Message Partner Server Name must correspond to the Application Interface Message Partner that is defined on the SAG as the client interface for the SWIFTNet Server.

Delivery Notification
Determines whether the server requests a delivery notification when a business partner is downloading. Possible values are True and False (default). Required.

Delivery Notification Request Type

The request type of the delivery notification is the value SI SWIFTNet Server uses in the response after getting a request from a remote client. Required.

Configure for failover SAG?

Enables you to set up active-standby configuration using two separate instances of the Remote API (RA), RA1 and RA2. Each RA should be configured to point to a different SAG to support failover processing. Possible values are True and False (default). Required.

Note: This parameter specifies whether to support failover if one SAG fails. When this parameter is set to True, you are presented with parameters for both an RA1 Profile and an RA2 Profile.
When you are operating in an environment with multiple SAGs configured in active-standby mode, setting this parameter enables you to define an alternate RA connection to a secondary SAG for failover support.

SNL Endpoint (for Store and Forward only)

The SNL endpoint used to receive data from SnF queues (for example, snl_sft). Optional—complete only if using store and forward processing.

Note: You must define endpoints on the SAG to route the InterAct messages to the correct application interface. If you are using store-and-forward, an extra endpoint is required to route messages coming from the store-and-forward queue (you can use the default endpoint for store-and-forward, **snl_sft**).

SnF Monitoring Interval (in seconds)

The store and forward monitoring interval (in seconds). Optional.

Note: This parameter enables you to indicate the interval that you want the SWIFTNet Server to check on the queue status. The SWIFTNet Server sets a timer to send the GetSnFStatusRequest message based on the value you enter.

Note: This feature is not supported in VMWare due to the manner in which VMWare delivers timer interrupts. You will need to disable this feature if you notice a slow down/delay in your system.

Return Signature List

Whether you want your own signature returned. Valid values are False (default, which indicates that normal Crypto is used) and True. Optional for T-Copy and Y-Copy implementation.

Use Input Channel (for InterAct Store and Forward only)

Whether to use the input channel with this adapter. Valid values are False (default) and True. You must select True if you are using an input channel. Required.

Note: Used for InterAct store-and-forward only. If you configure this parameter, the SWIFTNet Server opens the Input Channel automatically during the startup (when the SWIFTNet Server Adapter is enabled). This Input Channel remains open until the SWIFTNet Server is shut down (or the SWIFTNet Server Adapter is disabled). During this time, you still have an option to send message using the input channel or

without the input channel. All you need to do is to indicate this by using this parameter in SWIFTNet Client service.

SWIFTNet RA

The absolute path of the RA1 installation directory for RA1 SWIFTNet. Required. For example, **/SWIFTAlliance/RA**.

Note: This parameter specifies where to pick up the remote API and execute to SAG.

Config

The relative path of the RA1 instance configuration directory (relative to the RA installation directory). Required. For example, **RA1/cfg**.

Note: If you are using the SWIFTNet Server adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Bin   This is added to the PATH environment variable to contain the SWIFTNet Server binaries. Possible value is bin. Required.

Note: If you are using the SWIFTNet Server adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Lib   This is added to the library path environment variable. Possible value is lib. Required.

Note: If you are using the SWIFTNet Server adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Category

This is the category of RA . Possible values are:
- RA (SNL facade library to access an SAG)
- SNL (a native SNL interface)
- DEFAULT (default set for the RA1 instance)

Required.

Note: If you are using the SWIFTNet Server adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process

(replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Delivery Responder DN
The distinguished name of the responder to which delivery notifications requested by the sender are sent. Optional.

Note: If left blank, Delivery Notifications requested by the server are sent to the responder indicated in the message; otherwise, it is sent to this responder

Delivery Notification
Determines whether the RA1 server is handling a delivery notification. Possible values are True and False (default). Optional. This is used for a FileAct get.

Delivery Notification DN
Distinguished name of the responder of the delivery notification. Optional.

Request Type of Del. Notifn
Request type of the delivery notification. This is used for a FileAct Get. Optional.

Send Del. Notifn before Backend Processing
Indicates if the server will send a delivery notification before the internal process is executed. Optional.

Event Status Tracking
Indicates if the server requires all the FileAct Event statuses to be returned. Valid values are:

- Minimal (default – only Completed, Rejected, Duplicated statuses will be returned)
- Full (all statuses are returned)

Required.

SWIFTNet RA
The absolute path of the RA2 installation directory for RA2 SWIFTNet. Required (based on Active-Active configuration). For example, **/SWIFTAlliance/RA**.

Note: This parameter is only displayed if **Active-Active Configuration** is set to True.

Config
The relative path of the RA2 instance configuration directory (relative to the RA2 installation directory). Required (based on Active-Active configuration). For example, **/RA2/cfg**.

Note: This parameter is only displayed if **Active-Active Configuration** is set to True.

Note: If you are using the SWIFTNet Server adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process

(replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Bin This is added to the PATH environment variable to contain the SWIFTNet Server binaries. Required (based on Active-Active configuration).

Note: This parameter is only displayed if **Active-Active Configuration** is set to True.

Note: If you are using the SWIFTNet Server adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Lib This is added to the library path environment variable. Required (based on Active-Active configuration).

Note: This parameter is only displayed if **Active-Active Configuration** is set to True.

Note: If you are using the SWIFTNet Server adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Category
This is the category of RA2. Possible values are:
- RA (SNL facade library to access an SAG)
- SNL (a native SNL interface)
- DEFAULT (default set for the RA1 instance)

Required (based on Active-Active configuration).

Note: This parameter is only displayed if **Active-Active Configuration** is set to True.

Note: If you are using the SWIFTNet Server adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Delivery Notification
Determines whether the RA2 server is handling a delivery notification. Possible values are True and False (default). Optional. This is used for a FileAct get.

Delivery Notification DN

Distinguished name of the responder of the delivery notification. Optional.

Request Type of Del. Notifn

Request type of the delivery notification. This is used for a FileAct Get. Optional.

Send Del. Notifn before Backend Processing

Indicates if the server will send a delivery notification before the internal process is executed. Optional.

Event Status Tracking

Indicates if the server requires all the FileAct Event statuses to be returned. Valid values are:

- Minimal (default – only Completed, Rejected, Duplicated statuses will be returned)
- Full (all statuses are returned)

Required.

Input Channel Name

The name of the input channel. Required only if you specified **True** for **Use Input Channel**.

Authoriser DN

The authorized distinguished name that will be used to open the input channel. Required only if you specified **True** for **Use Input Channel**.

Force Open the Input Channel

Whether to force open the input channel or use normal mode. Valid values are False (use normal mode, which is the default) and True (force the input channel). Required only if you specified **True** for **Use Input Channel**.

Max. Resend Attempts

The maximum number of resend attempts allowed before the Sterling B2B Integrator automatically sends a Resolve Gap request to SWIFT. The default is 3. Required only if you specified **True** for **Use Input Channel**.

Run As User

Identify a user who has permission to run the scheduled activity. You can type the user ID, click the button to select the user ID from the list, and click **Save**.

Note: You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Use 24 Hour Clock Display

By default, the scheduling wizard displays times using a 12-hour clock (which designates the time in hours as a.m. or p.m.). Use this option to display times using a 24-hour clock.

Note: We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Do not use schedule

Removes all references to a schedule from the service. If you select this

option, you cannot enable the schedule in the future. You must recreate the schedule instead. Use this option only when you do not need a schedule for the service. This is the default option.

> **Note:** We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Run based on timer
Run the service at a certain time or time interval, such as every 2 hours.

> **Note:** We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Select Time
Type the time at which you want the Resend Scheduler to run.

> **Note:** We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Run daily
Run the service one or more times every day.

> **Note:** We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Run based on days of the week
Run the service on certain days of the week, such as every Monday.

> **Note:** We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Run based on days of the month
Run the service on certain days of the month, such as the 1st or 15th of every month.

> **Note:** We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Schedule Exclusions
Allows you to add any schedule anomalies (when the Resend Scheduler should not run).

> **Note:** We recommend you leave this parameter blank (that is, do not create any schedule exclusions).

Date Exclusions

Allows you to add any date anomalies (any date on which the Resend Scheduler should not run).

Note: We recommend you leave this parameter blank (that is, do not create any date exclusions).

New Business Entity

Click **add** to create a new business entity or click **edit** to modify an existing entity.

Note: You must have at least one business entity created to proceed.

Entity Identifies the security context to be used. For the client, the business entity is the requester. For the server, the business entity is the responder. Required for each configured entity to access a proprietary SWIFTNet PKI certificate to set up a valid security context.

Note: This is the distinguished name created by SWIFT. This parameter is only displayed if you edit an existing Business Entity or add a new Business Entity. The business entities are shared by both the RA1 and RA2 profiles.

UserId

The user identifier for this business entity (to log in to SWIFTNet). Required for each configured entity.

Note: The UserName is created in SAG (in the Users Module) and must also have a certificate created for it in the SAG. This parameter is only displayed if you edit an existing Business Entity or add a new Business Entity.

Password

The user password for this business entity (to log in to SWIFTNet). Required for each configured entity.

Note: This password is automatically encrypted. This parameter is only displayed if you edit an existing Business Entity or add a new Business Entity.

Message Queue

The name of the store and forward queue from which to receive messages. Required in Store and Forward mode.

Notification Queue

The Name of the store-and-forward queue to retrieve delivery notifications (optional; if empty, same as Message Queue). Required in Store and Forward mode.

Acquire queue by force

Whether to acquire the queue by force. Valid values are False (default) and True. Required.

Use Default Delivery Notification

Indicates whether to use the default delivery notification configuration on the RA1 page. Required.

Delivery Notification (Del. Notifn)

Indicates whether the sender asked the receiver to send a delivery notification. Optional. Valid values are True (default) or False.

Note: This parameter is only available when **Use Default Delivery Notification** is not selected.

Request Type of Del. Notifn

If **Delivery Notification (Del. Notifn)** is set to True, the value of this parameter is used to request a specific delivery notification message from the remote receiving server application when it returns the delivery notification. Optional.

Note: This parameter is only available when **Use Default Delivery Notification** is not selected.

Reception Directory

The full directory path where the file is received and stored during FileAct Put mode. Required for FileAct. Optional.

Download Directory

The full directory path where the file is picked up and sent to the requestor during FileAct Get mode. Required for FileAct. Optional.

Success Directory

The full directory path that must be specified when using the FileAct #OLDEST_FILE feature. Required for FileAct. Optional.

5. After configuring the SWIFTNet Server adapter in the Admin Console, click the **Enable Service for Business Process** check box on the Confirm page to enable the instance.

6. Once the SWIFTNet Server adapter is configured and saved, click the **Enabled** check box on the Services Configuration page. This starts the SWIFTNet Server. You should wait to ensure that the SWIFTNet Server starts.

7. On the Confirm page, verify that the **Enable Service for Business Processes** check box is selected to enable the adapter instance.

Configure the SWIFTNet Client Service for SWIFTNet 6.x

About this task

To configure the SWIFTNet Client service to be used in conjunction with an input channel:

Procedure

1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet Client service or select it from the list and click **Go!**.
3. Click **Edit**.
4. Specify field settings in the Admin Console.

Note: Each instance of the SWIFTNet Client service is configured for a pair of requestor/responder DNs and the SWIFTNet Client service name.

Field   Description

Name   Unique and meaningful name for the service configuration. Required.

Description

Meaningful description for the service configuration, for reference purposes. Required.

Select a Group

Select one of the options:

- None – Do not include the configuration in a service group at this time.
- Create New Group – Enter a unique name for a new group, which will be created with this configuration. (You can then add other services to the group as well.)
- Select Group – If service groups already exist for this service type, they are displayed in the list. Select a group from the list.

SWIFTNet Interface

SWIFTNet message type. Valid values are **InterAct** or **FileAct**. Required.

Store and Forward

Indicates if the file transfer is done using the store-and-forward method. Valid values are True (use Store-and-Forward) and False (default—do not use Store-and-Forward). Required. BPML element value is **SnF**.

SWIFTNet Operation

The SWIFTNet operation to send an InterAct or FileAct message. Possible values are:
- Synchronous (default)—InterAct
- Asynchronous—InterAct
- Put—FileAct (default)
- Get—FileAct

Required. BPML element value is **sync** (default) or **async for InterAct, or Put or Get for FileAct**.

Requestor DN

Distinguished name of the requestor. Required. BPML element value is **requestorDN**.

Note: This DN must be registered with the SAG instance using SWIFTNet Alliance Webstation.

Responder DN

Distinguished name of the responder. Required. BPML element value is **responderDN**.

Note: This DN must be registered with the SAG instance using SWIFTNet Alliance Webstation.

Service Name

Name of the service to which both SWIFT correspondents have subscribed. Required. BPML element value is **serviceName**.

Note: This must be a SWIFTNet service to which you are subscribed.

Authoriser DN

The distinguished name of the authorizing party. Optional.

This service allows Third Party Copy

Whether this service uses T-Copy or Y-Copy (check your service agreement with SWIFT). Only available for FileAct store-and-forward. BPML element value is **thirdPartyCopy**. Valid values are TRUE or FALSE.

This parameter is displayed only if you selected **File Act** and **True** for **Store and Forward** on SWIFTNet Client Service Interface page.

Note: If the Copy Mode is Y-Copy, the Sterling B2B Integrator sends an authorization message, which is like sending an Interact store-and-forward request. The SWIFTNet Client service is used, but you must set the **This service allows Third Party Copy** parameter to TRUE, and provide the authorization decision (either Authorised or Refused) for the AuthDecision parameter.

Request for Third Party Copy
Whether you are requesting third party copy. When the Copy feature is defined as Optional in the service agreement, you can choose whether you want the Third Party Copy to occur. BPML element value is **copyIndicator**. Valid values are TRUE or FALSE. Displayed only if you select True for **This service allows Third Party Copy**.

Note: This parameter is displayed only if you selected **True** for **This service allows Third Party Copy**.

Request for Notification from Third Party
In T-Copy mode, this setting is not applicable, the value should always be set to FALSE.

In Y-Copy mode, when the **Authorisation Notification Indicator** feature is available and defined as Optional in the service agreement, you can choose whether you want to receive the Authorisation Notification messages. BPML element value is **authNotifIndicator**. Valid values are TRUE or FALSE. Displayed only if you selected **True** for **This service allows Third Party Copy**.

Note: This parameter is displayed only if you selected **True** for **This service allows Third Party Copy**.

Request Type
Request type supported by the message exchange. Optional for InterAct and required for FileAct in SWIFTNet 6.0. BPML element value is **requestType**.

Note: In SWIFTNet 6.0 FileAct the format convention is as follows:

```
<business_area>.<type_of_syntax>.<detailed_syntax_and_format>
```

This format starts with a four-characters business area code, followed by a period (dot), followed by a three-character code that designates the type of syntax (which can be <nnn>, FIN, or xxx), followed by another period (dot), and then followed by a more detailed indication of syntax and format.

Request Reference
User reference of the request. Optional. BPML element value is **requestReference**.

Non Repudiation Required
Indicates whether non-repudiation is required. Possible values are True (when enabled this means that trading partners cannot deny that they sent a request) or False (default—when enabled this indicates that non-repudiation is not required). Optional. BPML element value is **nonRepudiation**.

End-to-End Signature Required

Whether an end-to-end signature is required. Valid values are False (default) and True. Optional.

Note: You can use an end-to-end signature regardless of whether you are using non-repudiation (for example, for SWIFT SCORE messages).

Number of Retries

Number of retries to connect to SAG. Default value is 3. Optional. BPML element value is **numOfRetries**.

Retry Delay (in seconds)

Number of delays before the next retry. Default value is 60 (seconds). Optional. BPML element value is **secInRetryDelay**.

Trace Trace for logging purposes in the SWIFTNet Server. Valid values are True and False (default). Required. BPML element value is **trace**.

Use Signature List

Whether to use a signature list. This enables you to select your own signatures. If you do not use a signature list then normal Crypto is used. Valid values are False and True. Required.

Note: This parameter is displayed only if you selected **True** for **End-to-End Signature Required**.

Return Signature List

Whether to return a signature list. Valid values are False and True. Required.

If you want a signature list returned, the SWIFTNet Server receives the requestor's own signature in the response message. This returned signature will be extracted and saved as a separate message. This message is stored in the database and is made available for Correlation search.

Note: This parameter is displayed only if you selected **True** for **End-to-End Signature Required**.

Use RND

Whether to use RND (digest reference values that terminate on "and RND"). Valid values are False (default) and True. Required.

Note: This parameter is displayed only if you selected **True** for **End-to-End Signature Required**.

Delivery Notification (Del. Notifn)

Indicates that the sender asked the receiver to send a delivery notification. Possible values are True or False (default). Optional. BPML element value is **deliveryNotification**.

Note: This parameter is only displayed when you select **True** for Store and Forward or are performing a FileAct Put. If you are performing a Put operation, you can request the responder to send you a delivery notification and specify a different Delivery Notification DN and Request Type of Delivery Notification, if desired. If you are performing a Get operation, the responder can request Delivery Notification from the requestor after receiving the file. That setting for delivery notification is configured through the SWIFTNet Server adapter.

Request Type of Delivery Notification

Used to request a specific delivery notification message from the remote receiving server application when it returns the delivery notification (when Delivery Notification is set to True). Optional. BPML element value is **requestTypeDelNotifn**.

Note: This parameter is only displayed when you select **True** for Store and Forward or a FileAct Put.

Message Priority

Indicates priority handling in the queue for store-and-forward only. Valid values are Normal (default) and Urgent. Optional. BPML element value is **messagePriority**.

Note: This value is used as a selection criterion when delivering messages from a queue, and in SWIFTNet FileAct to influence the pace of the FileAct flow.

Use Input Channel

Whether to use the input channel. Valid values are False (default) and True. Required. This parameter is displayed only if you selected **True** for **Store and Forward** and **InterAct** for **SWIFTNet interface**.

Note: Used for InterAct store-and-forward only. Select **True** if you are using an input channel. If you configure this parameter, the SWIFTNet Server opens the Input Channel automatically during the startup (when the SWIFTNet Server Adapter is enabled). This Input Channel remains open until the SWIFTNet Server is shut down (or the SWIFTNet Server Adapter is disabled). During this time, you still have an option to send message using the input channel or without the input channel. All you need to do is to indicate this by using this parameter in SWIFTNet Client service.

MEFG SWIFTNet IP

The IP address for the SWIFTNet Server. Required.

MEFG SWIFTNet Port

The port for the SWIFTNet Server. Default is 80. Optional.

Response Timeout

The timeout interval (in seconds) in which a response must be received or the message operation fails. Optional. Default is 60 seconds.

Use SSL

Whether to enable Secure Socket Layer (SSL) over HTTP communication between the Sterling B2B Integrator and the SWIFTNet Server. Valid values are None and Must.

Note: Regardless of the value you select for **Use SSL**, you must also update the business processes associated with the SWIFTNet Client service.

Cipher Strength

Indicates the strength of the cipher. Possible values are ALL (default), WEAK, and STRONG. Optional.

CA Certificate

The CA certificate of the SWIFTNet Server.

> **Note:** This is the public key certificate that must be configured to set
> up the outbound SSL channel. This page is only displayed if you set
> **Use SSL** to **Must**.

> **Note:** The SWIFTNet Client service Configuration page allows you to
> select the same CA Certificate for SSL processing a second time, and
> continues to allow additional selections of the same certificate in
> subsequent edits. If you have already selected a CA Certificate once for
> a configuration of the SWIFTNet Client service, do not select the same
> CA Certificate again, as this will result in an error when you execute
> the relevant business process.

Switch to SnF mode when real-time transmission failed
> Whether to switch to store-and-forward mode when real-time
> transmission fails. Select True if you want to switch to Store and
> Forward mode when the real-time transmission (InterAct and FileAct
> Put) has failed. Valid values are True and False.

5. On the Confirm page, verify that the **Enable Service for Business Processes**
   check box is selected.

Configure the SWIFTNet7 Adapter for SWIFTNet 7

About this task

To configure the SWIFTNet7 Adapter:

Procedure

1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet7 Adapter or select it from the list and click **Go!**.
3. Click **Edit**.
4. Specify field settings in the Admin Console.

> **Note:** Specify failover processing to ensure that failover is supported if a SAG
> connection fails by configuring **failover-SAG**.

Use the field definitions in the following table to create a new configuration of
the SWIFTNet7 adapter, or to set up the configuration provided with the
application.

> **Note:**

The business entities (accessible through the Business Entities wizard as part of
the SWIFTNet7 adapter configuration) are shared by both RA1 and RA2. The
Business Entities wizard enables you to add multiple entities.

Field   Description

Name   Unique and meaningful name for the adapter configuration. Required.

Description
> Meaningful description for the adapter configuration, for reference
> purposes. Required.

Environment
> Select the environment in which the adapter will run. Required.

> **Note:** The adapter is not eligible to run in the adapter container.

Select a Group

Select one of the options:

- None – Do not include the configuration in a service group at this time.
- Create New Group – Enter a unique name for a new group, which will be created with this configuration. (You can then add other services to the group as well.)
- Select Group – If service groups already exist for this service type, they are displayed in the list. Select a group from the list.

SI Server Address

The callback IP of Sterling B2B Integrator for the SWIFTNet Servers. Required.

Note: The default value is the IP address of the machine where the Sterling B2B Integrator is installed.

SI HTTP Server Adapter Port

This is the listening port for the SWIFTNet7 HTTP Server Adapter. Required. The default populated value is the instance port number of the Sterling B2B Integrator instance plus 53. For example, if the Sterling B2B Integrator instance port is 34600, the listening port populated by default is 34653.

Note: The HTTP Server adapter functions between the SWIFTNet7 adapter and the SWIFTNet Servers. For an SSL connection, this value should be server name because the certificate is made with the server name.

Note: If you are using the SWIFTNet7 adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

MEFG SWIFTNet Address

The IP address of the SWIFTNet Servers. Required.

MEFG SWIFTNet Port

The port of the SWIFTNet Servers. Required.

CLA2Client Listening Port

The listening port used by the client command adapter (CLA2Client) running along the SWIFTNet Servers. Required.

Note: This port listens for requests to start and stop the SWIFTNet Servers.

MEFG SWIFTNet Home

The home directory of the SWIFTNet Servers. Required.

Start MEFG SWIFTNet despite errors

Whether to start the SWIFTNet Servers if errors occur. The default is unchecked (do not start the SWIFTNet Servers if errors occur). Optional.

Use SSL

Whether to enable Secure Socket Layer (SSL) over HTTP communication between the Sterling B2B Integrator and the SWIFTNet Servers. Valid values are False (default) and True. Select **True** to use SSL with an Input Channel.

Event Status Tracking

The event status tracking for the adapter. Required. Valid selections are Minimal (Only Completed, Rejected, Duplicated status) - this is the default, and Full.

Cipher Strength

Specifies the strength of the algorithms (cipher suites) used to encrypt data. Valid values are:

- STRONG - Required if Use SSL is Must
- ALL - All cipher strengths are supported
- WEAK - Often required for international trade, because government regulations prohibit STRONG encryption from being exported

Default is ALL. Required if **Use SSL** is checked.

CA Certificate

Move one or more CA Certificates to the use column. These are the digital security certificates that the SSL server will use to authenticate the client. Required if SSL is selected.

Message Partner Client Name

The client message partner name that the SNL server application recognizes for the SWIFTNet Servers client application.

Note: The Message Partner Client Name must correspond to the Application Interface Message Partner that is defined on the SAG as the client interface for the SWIFTNet Servers.

Message Partner Server Name

The server message partner name that the SNL server application recognizes for the SWIFTNet Servers server application.

Note: The Message Partner Server Name must correspond to the Application Interface Message Partner that is defined on the SAG as the client interface for the SWIFTNet Servers.

Delivery Notification

Determines whether the server requests a delivery notification when a business partner is downloading. Possible values are True and False (default). Required.

Delivery Notification Request Type

The request type of the delivery notification is the value SI SWIFTNet Server uses in the response after getting a request from a remote client. Required.

Configure for failover SAG?

Enables you to set up Failover SAG Configuration using two separate instances of the Remote API (RA), RA1 and RA2. Each RA should be configured to point to a different SAG to support failover processing. Possible values are True and False (default). Required.

Note: This parameter specifies whether to support failover if one SAG fails. When this parameter is set to True, you are presented with

parameters for both an RA1 Profile and an RA2 Profile. When you are operating in an environment with multiple SAGs configured in failover SAG mode, setting this parameter enables you to define an alternate RA connection to a secondary SAG for failover support.

Delivery Notification
Determines whether the RA1 server is handling a delivery notification. Possible values are True and False (default). Optional. This is used for a FileAct Get.

Delivery Notification DN
Distinguished name of the responder of the delivery notification. This is used for a FileAct Get. Optional.

Request Type of Del. Notifn
Request type of the delivery notification. This is used for a FileAct Get. Optional.

Return Signature List
Whether you want to receive your own signature to safe store it in case of any dispute in the future. Valid values are False or True. Applicable only if Signature List is used. Optional for T-Copy and Y-Copy implementation.

SWIFTNet RA
The absolute path of the RA1 installation directory for RA1 SWIFTNet. Required. For example, **/SWIFTAlliance/RA**.

   Note: This parameter specifies where to pick up the remote API and execute to SAG.

Config
The relative path of the RA1 instance configuration directory (relative to the RA installation directory). Required. For example, **RA1/cfg**.

   Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Bin   This is added to the PATH environment variable to contain the SWIFTNet Server binaries. Possible value is bin. Required.

   Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Lib   This is added to the library path environment variable. Possible value is lib. Required.

   Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library,

you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Category

This is the category of RA. Possible values are:

- RA (SNL facade library to access an SAG)
- SNL (a native SNL interface)
- DEFAULT (default set for the RA1 instance)

Required.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Delivery Responder DN

The distinguished name of the responder to which delivery notifications requested by the sender are sent. Optional.

Note: If left blank, Delivery Notifications requested by the server are sent to the responder indicated in the message; otherwise, it is sent to this responder

SWIFTNet RA

The absolute path of the RA2 installation directory for RA2 SWIFTNet. Required (based on Failover SAG Configuration). For example, **/SWIFTAlliance/RA**.

Note: This parameter is only displayed if **Failover SAG Configuration** is set to True.

Config

The relative path of the RA2 instance configuration directory (relative to the RA2 installation directory). Required (based on Failover SAG Configuration). For example, **/RA2/cfg**.

Note: This parameter is only displayed if **Failover SAG Configuration** is set to True.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Bin   This is added to the PATH environment variable to contain the SWIFTNet Server binaries. Required (based on Failover SAG Configuration).

Note: This parameter is only displayed if **Failover SAG Configuration** is set to True.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Lib   This is added to the library path environment variable. Required (based on Failover SAG Configuration).

Note: This parameter is only displayed if **Failover SAG Configuration** is set to True.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Category
This is the category of RA2. Possible values are:
- RA (SNL facade library to access an SAG)
- SNL (a native SNL interface)
- DEFAULT (default set for the RA1 instance)

Required (based on Failover SAG Configuration).

Note: This parameter is only displayed if **Failover SAG Configuration** is set to True.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Delivery Responder DN
The responder to which delivery notifications requested by the sender are sent. Required (based on the failover SAG configuration).

Note: If left blank, Delivery Notifications requested by the server are sent to the responder indicated in the message; otherwise, it is sent to this responder This parameter is only displayed if failover SAG Configuration is set to True.

New Security Context
Click add to create a new security context for the Output Channel Configuration or click edit to modify an existing entry.

Note: You must have at least one security context created to proceed. This is the authorization context to open the output channel.

UserId

The user identifier for this business entity (to log in to SWIFTNet). Required for each configured entity.

Password (RA1)

The user password for this security context for RA1. Required for each configured entity.

Note: This password is automatically encrypted. This parameter is only displayed if you edit an existing Security Context or add a new Security Context.

Password (RA2)

The user password for this security context for RA2. If not specified, the RA2 password is considered to be equal to the RA1 password. Optional.

Note: This password is automatically encrypted. This parameter is only displayed if you edit an existing Security Context or add a new Security Context.

New Business Entity

Click **add** to create a new business entity or click **edit** to modify an existing entity.

Note: You must have at least one business entity created to proceed.

Entity Identifies the security context to be used. For the client, the business entity is the requester. For the server, the business entity is the responder.

The SWIFTNet7 Adapter enhances business entities. You are not required to define an exact match for every distinguished name used for requestor or responder. The SWIFTNet Client Adapter uses the best matching algorithm to find the business entity and associated context. Therefore, if all the distinguished names used belong to the same organization (for example: o=abcdefgh,o=swift), it is sufficient for you to define one and only one business entity: o=abcdefgh,o=swift. However, if necessary, (for example if you want to ensure that cn=xyz,o=abcdefgh,o=swift will use different context), you may define another business entity for this particular distinguished name.

Note: This is the distinguished name created by SWIFT. This parameter is only displayed if you edit an existing Business Entity or add a new Business Entity. The business entities are shared by both the RA1 and RA2 profiles.

UserId

The user identifier for this business entity (to log in to SWIFTNet). Required for each configured entity.

Note: The UserName is created in SAG (in the Users Module) and must also have a certificate created for it in the SAG. This parameter is only displayed if you edit an existing Business Entity or add a new Business Entity. The SWIFTNet Client Adapter allows you to use a different security context for RA1 and RA2.

UserId for RA2 (if different)
The user identifier for the RA2 business entity (to log in to SWIFTNet).
Required for each configured entity.

Note: The UserName is created in SAG (in the Users Module) and
must also have a certificate created for it in the SAG. This parameter is
only displayed if you edit an existing Business Entity or add a new
Business Entity and only displayed if Failover SAG Configuration is
enabled.

Use Default Delivery Notification
Indicates whether to use the default delivery notification configuration.
Required.

Delivery Notification (Del. Notifn)
Indicates whether the sender asked the receiver to send a delivery
notification. Optional. Valid values are True (default) or False.

Note: This parameter is only available when Use Default Delivery
Notification is not selected.

Request Type of Del. Notifn
If Delivery Notification (Del. Notifn) is set to True, the value of this
parameter is used to request a specific delivery notification message
from the remote receiving server application when it returns the
delivery notification. Optional.

Note: This parameter is only available when Use Default Delivery
Notification is not selected.

Use Input Channel (for InterAct Store and Forward only)
Whether to use the input channel with this adapter. Valid values are
False (default) and True. You must select True if you are using an input
channel. Required.

Note: Used for InterAct store-and-forward only. If you configure this
parameter, the SWIFTNet Servers opens the Input Channel
automatically during the startup (when the SWIFTNet7 Adapter is
enabled). This Input Channel remains open until the SWIFTNet Servers
is shut down (or the SWIFTNet7 Adapter is disabled). During this time,
you still have an option to send message using the input channel or
without the input channel. All you need to do is to indicate this by
using this parameter in SWIFTNet Client profile.

Use Output Channel (for Store and Forward only)
Whether to use the output channel with this adapter. Valid values are
False (default) and True. You must select True if you are using an
output channel. Required.

Note: Used for store-and-forward only.

Input Channel Name
The name of the input channel. Required only if you specified **True** for
Use Input Channel.

Authoriser DN
The authorized distinguished name that will be used to open the input
channel. Required only if you specified **True** for **Use Input Channel**.

Force Open the Input Channel

Whether to force open the input channel or use normal mode. Valid values are False (use normal mode, which is the default) and True (force the input channel). Required only if you specified True for Use Input Channel.

Max. Resend Attempts

The maximum number of retries that should be attempted before sending the Resolve Gap request. Required. The default is three.

New Queue

Click **add** to create a new queue or click **edit** to modify an existing entry.

Note: If you are using an Output Channel, you must have at least one queue created to proceed.

Queue Name

The name of the queue you are creating. Optional.

Output Channel Name

The name of the output channel. Required if you are using an Output Channel.

Note: This name must be unique and can only be associated with one SWIFTNet7 Adapter instance. By default there is an output channel with the same name as the queue name. Alternatively, you may create the output channel first using the utility provided.

Operating DN

The distinguished name of the operator. Optional.

SNL Endpoint

The endpoint for SNL. Optional.

Traffic Filter #1

Traffic filter to be applied first. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Traffic Filter #2

Traffic filter to be applied second. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Traffic Filter #3

Traffic filter to be applied third. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Traffic Filter #4

Traffic filter to be applied fourth. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Traffic Filter #5

Traffic filter to be applied fifth. Optional. Valid values are System,

InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Traffic Filter #6
Traffic filter to be applied sixth. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Reception Directory
The full directory path where the file is received and stored during FileAct Put mode. Required for FileAct. Optional.

Download Directory
The full directory path where the file is picked up and sent to the requestor during FileAct Get mode. Required for FileAct. Optional.

Enable Service for Business Process
Whether the adapter is enabled to be used with business processes. Optional.

5. After configuring the SWIFTNet7 Adapter in the Admin Console, click the **Enable Service for Business Process** check box on the Confirm page to enable the instance.

6. Once the SWIFTNet7 Adapter is configured and saved, click the **Enabled** check box on the Services Configuration page. This starts the SWIFTNet Servers. You should wait to ensure that the SWIFTNet Servers starts.

7. On the Confirm page, verify that the **Enable Service for Business Processes** check box is selected to enable the adapter instance.

Configuring the SWIFTNet7 Client Service for SWIFTNet 7
About this task

To configure the SWIFTNet7 Client Service to use the input channel, you can pass parameters to the SWIFTNet7 Client service from a business process. The following table contains the parameters passed from the business process to the SWIFTNet7 Client service:

Parameter
Description

profileName
The name of the SWIFTNet Client Profile, if specified. If you specify this parameter, the service looks for the associated SWIFTNet Client Profile and uses all the parameters defined there. However, you may override some parameters from the business process. Those parameters that can be overridden should not be related to Application Service Profile (ASP). If there is no profileName parameter is specified, all the parameters are taken only from the business process. These parameters cannot be overridden:

- "Message Type
- "Service Name
- "Request Type
- "SnF Mode
- "Operation Type
- "Non Repudiation
- "End-to-End Signing

- "Copy Indicator
- "Auth. Notification Indicator

These parameters can be overridden:

- "SWIFTNet Server Adapter - the BPML parameter is "serverAdapterName"
- "Requestor DN - the BPML parameter is "requestorDN"
- "Responder DN - the BPML parameter is "responderDN" "Authoriser DN - the BPML parameter is "authoriserDN"
- "Request Reference - the BPML parameter is "requestReference"
- "Message Priority - the BPML parameter is "messagePriority"
- "Possible Duplicate flag - the BPML parameter is "possibleDuplicate"
- "Message ID - the BPML parameter is "messageID"
- "Header Info - the BPML parameter is "HeaderInfo"
- "Business Application Header - the BPML parameter is "AppHeader"
- "Return Signature List (when SignatureList is used) - the BPML parameter is "returnSignatureList"
- "Use RND (when SignatureList is used) - the BPML parameter is "useRND"
- "Request for Del. Notification flag - the BPML parameter is "deliveryNotification"
- "Notification Queue - the BPML parameter is "notificationQueue"
- "Del. Notification as System Message - the BPML parameter is "notificationAsSystemMsg"
- "Overdue Delay - the BPML parameter is "overdueDelay"
- "Del. Notification DN - the BPML parameter is "deliveryNotificationDN"
- "Del. Notification Request Type - the BPML parameter is "deliveryNotificationRT"
- "Use Input Channel flag - the BPML parameter is "useInputChannel"
- "Recipient DNs - the BPML parameter is "recipientDN[1..N]
- "Recipient List is public - the BPML parameter is "isRecipientListPublic"
- "Third Party DNs - the BPML parameter is "thirdPartyDN[1..N]
- "Physical File Name - the BPML parameter is "physicalFilename"
- "Logical File Name - the BPML parameter is "logicalFilename"
- "Transfer Description - the BPML parameter is "transferDesc"
- "Transfer Info - the BPML parameter is "transferInfo"
- "File Description - the BPML parameter is "fileDesc"
- "File Info - the BPML parameter is "fileInfo"
- "Authorization Status - the BPML parameter is "AuthDecision"
- "Third Party to Receiver Info - the BPML parameter is "ToRcvrInfo"
- "Third Party to Sender Info - the BPML parameter is "ToSndrInfo"
- "Third Party Refusal Info - the BPML parameter is "RefuseReason"
- "Copy SnF Reference - the BPML parameter is "MessageName"
- "Queue Name - the BPML parameter is "queueName"
- "Queue Sharing Mode - the BPML parameter is "queueSharingMode"
- "Report Option/Report Time - the BPML parameter is "reportOption"
- "Requestor Pattern - the BPML parameter is "requestorPatternCriteria"

- "Input Channel Name - the BPML parameter is "inputChannelCriteria"
- "Messaging Service - the BPML parameter is "messagingServiceCriteria"
- "Country or Region Location - the BPML parameter is "locationCriteria1"
- "Report Service Name - the BPML parameter is "serviceNameCriteria"
- "Report Priority - the BPML parameter is "messagePriorityCriteria"
- "(Input/Output) Channel Pattern - the BPML parameter is "channelPatternCriteria"
- "Queue Name Pattern - the BPML parameter is "queuePatternCriteria"
- "Queue First - the BPML parameter is "queueFirstCriteria"
- "Report Version - the BPML parameter is "reportVersion"
- "Report Start Time - the BPML parameter is "reportStartTime"
- "Report End Time - the BPML parameter is "reportEndTime"

serverAdapterName
> The name of SWIFTNet7 Adapter instance or group. The service uses this instance/group name to look up its configuration for the SWIFTNet Server host and port, security context and also input channel, and so forth. If this parameter is not specified, the service will look for the default instance (SWIFTNet7Adapter).

numOfRetries
> The total number of times to resend this message in case of any transient failure. If not specified, the default is three times.

switchToSnF
> Whether to switch to store-and-forward mode when real-time transmission fails. Select True if you want to switch to Store and Forward mode when the real-time transmission (InterAct and FileAct Put) has failed. Valid values are True and False.

SnFServiceName
> The name of the store-and-forward service. Required when Switch to SnF mode when real-time transmission failed is set to True.

SnF	Indicates if the file transfer is done using the store-and-forward method. Valid values are True (use Store-and-Forward) and False (default—do not use Store-and-Forward). Required.

swiftOp
> The SWIFTNet operation to send an InterAct or FileAct message Possible values are:
> - sync (default)—InterAct
> - async—InterAct
> - put—FileAct
> - get—FileAct
>
> Required.

nonRepudiation
> Indicates whether non-repudiation is required. Possible values are TRUE (when enabled, trading partners cannot deny that they sent a request) or FALSE (default, indicating that non-repudiation is not required). Optional.

requestType
> SWIFT request type, For example, pain.001.02.01 or xsys.018.001.01.

serviceName

Name of the service to which both SWIFT correspondents have subscribed. Required.

This must be SWIFTNet service to which you are subscribed. For example: swift.devsup.fa!x1. This service name may be overwritten when sending a system message.

requestReference

User reference of the request. This parameter is alphanumeric. Optional.

requestorDN

Distinguished name of the requestor. Required.

Note: This DN must be registered with the SAG instance using SWIFTNet Alliance Webstation.

responderDN

Distinguished name of the responder. Required.

This DN must be registered with the SAG instance using SWIFTNet Alliance Webstation. This DN may be overwritten in the case of sending system message or file/message distribution.

authoriserDN

Distinguished name for the authoriser. This is used for a non-input channel to look for context.

messagePriority

Indicates priority handling in the queue for store-and-forward only. Optional.

This value is used as a selection criterion when delivering messages from a queue, and in SWIFTNet FileAct to influence the pace of the FileAct flow. Valid values are Normal or Urgent. Leave this parameter blank if you want to use the default.

messageID

Message identifier for resending a message if Possible Duplicate is set to TRUE. Optional.

Note: Normally, you only need to specify this if possibleDuplicate = TRUE.

possibleDuplicate

Indicates whether to include a trailer specifying that this message may be a duplicate.

This is an optional component of the envelope that indicates that this message may already have been sent. For example, if the system crashes during the delivery of a message, another copy of the message could be sent, with this trailer included to indicate that it may be a duplicate.

Possible values are TRUE and FALSE (default). Optional.

HeaderInfo

The Enhanced Header information. Since Enhanced Header Info is usually an XML structure, you should specify it as CDATA.

AppHeader

An XML structure that is only used in InterAct case (either real-time or SnF). If this parameter is not specified, the default AppHeader structure is used (as in version 6.x).

sign   Whether an end-to-end signature is required. Valid values are Crypto, SignatureList, or do not specify this tag if you do not want to use signing.

returnSignatureList
> Whether to return a signature list.

useRND
> Whether to use RND (digest reference values that terminate on "and RND"). Valid values are False (default) and True.

deliveryNotification
> Indicates that the sender asked the receiver to send a delivery notification. Possible values are TRUE or FALSE. Optional.
>
> **Note:** This parameter is only displayed when you select True for SnF or are performing a FileAct Put. If you are performing a Put operation, you can request the responder to send you a delivery notification and specify a different Delivery Notification DN and Request Type of Delivery Notification, if desired. If you are performing a Get operation, the responder can request Delivery Notification from the requestor after receiving the file. That setting for delivery notification is configured through the SWIFTNet Server adapter. For System Message or InterAct Real-Time, this parameter will be overwritten to FALSE. Otherwise, you can choose whether you want to receive delivery notification.

notificationQueue
> Whether a notification queue will be used. If you selected SnF = TRUE, this parameter is required.

overdueDelay
> Specify the overdue delay. For SnF = TRUE, you must specify an integer value between 5 - 20160 (minutes). If the value between 5 - 1440, the tag <Sw:OverdueWarningDelay> will be output. Otherwise the tag <Sw:OverdueTime> will be output.

notificationAsSystemMsg
> For SnF = TRUE, deliver the notification as system message TRUE/FALSE. If using overdueDelay, this will be set to TRUE.

deliveryNotificationDN
> For FileAct Real-Time, this is the delivery notification distinguished name (valid DN).

deliveryNotificationRT
> For FileAct Real-Time, this is the delivery notification Request Type.

useInputChannel
> For InterAct SnF, whether to use the input channel.

copyIndicator
> When using Copy Service this is an indicator that you want the Copy to occur. When ASP is used, this parameter is validated against ASP. Otherwise, when this parameter is specified, the system assumes you are using CopyService.

authNotifIndicator
> When using Copy Service this is an indicator that you want to receive authorisation notification from third party (for Y-Copy). If ASP is used, this parameter is validated. Otherwise, it is assumed that if this parameter is specified by the user, that this service is a Y-Copy service.

thirdPartyDN[1..N]

Copy destination(s). For T-Copy service, you can specify up to ten. For Y-Copy, you can only specify one. This is only validated if ASP is used.

recipientDN[1..N]

Allows you to specify multiple destination if it is not a copy service.

isRecipientListPublic

Whether the recipient list if public. Used only if recipient List is used.

physicalFilename

Optional. Only for FileAct. For a FileAct Put, this is the full path and the physical name of the file to send. For a FileAct Get, this is the full path and the physical name of the file to save after the Get is completed.

logicalFilename

This name is communicated to the server application. By default, this name is the physical name without the file path. Optional. Only for FileAct.

For a FileAct Put, this is the logical name of the file to be saved based on the <reception_dir>/<responder_dn>/<requestor_dn>

For a FileAct Put, this is the logical name of the file to be saved based on the <download_dir>/<responder_dn>/<requestor_dn>

transferDesc

The transfer description. Only for FileAct. Optional.

transferInfo

User information about the transfer. Only for FileAct. Optional.

Note: This must follow the BNF format (see *Appendix C* of *SNL Interface Specifications*).

fileDesc

User information about the file transfer. Only for FileAct. Optional.

fileInfo

User description about the file transfer. Only for FileAct. Optional.

Note: This must follow the BNF format (see *Appendix C* of *SNL Interface Specifications*).

AuthDecision

The third party decision. Valid values are Authorised or Refused. Use this parameter with the thirdPartyAuth parameter.

ToRcvrInfo

If you are the third party and decide to authorize a request or file notification, this parameter enables you to specify Third Party to Receiver Information. The information can be any format (plain text or XML). If you use XML format, this parameter should be CDATA.

ToSndrInfo

If you are the third party and decide to authorize a request or file notification, this parameter enables you to specify Third Party to Sender Information. The information can be in any format (plain text or XML). If you use XML format, this parameter should be CDATA.

RefuseReason

If you are the third party and you refuse a request or file notification, you

can specify a Refusal Reason in this parameter. The information can be any format (plain text or XML). If you use XML format, this parameter should be CDATA.

MessageName
> The name of the HeaderInfo message, as inserted into the mailbox. The format is ThirdParty_[CopySnFRef]. When you use Mailbox Extract service to extract the HeaderInfo message from mailbox, by default the name is available in Process Data.

queueName
> The queue name.

queueSharingMode
> The queue sharing mode. Valid values are Shared or Exclusive.

reportOption
> The report option. Depending on the system message sent, there may be different validations. For example, the xsys.004.001.01 format is RT or HHHMM. For xsys.008.001.*, the format is Full. For the xsys.018.001.01, the format is Input or Output.

requestorPatternCriteria
> This is a request for pattern criteria. You can only use a wildcard at the last level distinguished name; the organization DN cannot use a wildcard.

inputChannelCriteria
> This is the input channel criteria. Valid values are BIC_a or BIC_* (wild card behind, BIC must match requestorDN).

messagingServiceCriteria
> This is the messaging service criteria. Valid values are InterAct or FileAct.

locationCriteria
> This is the location criteria.

serviceNameCriteria
> This is the service name criteria (specify any valid service).

messagePriorityCriteria
> This is the message priority criteria. The valid values are Urgent or Normal.

channelPatternCriteria
> This is the channel pattern criteria. Valid values areBIC_a or BIC_* (wild card behind, BIC must match requestorDN).

queuePatternCriteria
> This is the queue pattern criteria. Valid values are BIC_a or BIC_* (wild card behind, BIC must match requestorDN).

queueFirstCriteria
> This is the queue first criteria. Valid values are TRUE or FALSE.

reportVersion
> This is the report version. Valid values are 1 or 2.

reportStartTime
> This is the start time of the report in Zulu time format (for example, 2010-05-30T15:34:21Z).

reportEndTime
> This is the end time of the report in Zulu time format (for example, 2010-05-30T15:34:21Z).

Configuring the SWIFTNet7 Adapter Scheduler for SWIFTNet 7

About this task

To configure the SWIFTNet7 Adapter Scheduler to use the input channel:

Procedure

1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet7 Adapter Scheduler or select it from the list and click **Go!**.
3. Click **Edit**.
4. Specify field settings in the Admin Console.

   Note: The business entities (accessible through the Business Entities wizard as part of the SWIFTNet Client adapter configuration) are shared by both RA1 and RA2. The Business Entities wizard enables you to add multiple entities.

   Field   Description

   Name   Unique and meaningful name for the service configuration. Required.

   Description
   Meaningful description for the service configuration, for reference purposes. Required.

   Select a Group
   Select one of the options:
   - None – Do not include the configuration in a service group at this time.
   - Create New Group – Enter a unique name for a new group, which will be created with this configuration. (You can then add other services to the group as well.)
   - Select Group – If service groups already exist for this service type, they are displayed in the list. Select a group from the list.

   Run As User
   Identify a user who has permission to run the scheduled activity. You can type the user ID, click the button to select the user ID from the list, and click **Save**.

   Note: You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

   Use 24 Hour Clock Display
   By default, the scheduling wizard displays times using a 12-hour clock (which designates the time in hours as a.m. or p.m.). Use this option to display times using a 24-hour clock.

   Note: We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

   Do not use schedule
   Removes all references to a schedule from the service. If you select this option, you cannot enable the schedule in the future. You must recreate the schedule instead. Use this option only when you do not need a schedule for the service. This is the default option.

Note: We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Run based on timer
Run the service at a certain time or time interval, such as every 2 hours.

Note: We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Select Time
Type the time at which you want the Resend Scheduler to run.

Note: We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

At startup
Select this checkbox if you want the SWIFTNet7 Adapter to run at system startup.

Run daily
Run the service one or more times every day.

Note: We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Run based on days of the week
Run the service on certain days of the week, such as every Monday.

Note: We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Run based on days of the month
Run the service on certain days of the month, such as the 1st or 15th of every month.

Note: We recommend that you set the Resend Scheduler to **Run based on timer** and set it for one minute under normal usage (that is, every one minute). You must configure the parameters on the Schedule Type page for the Resend Scheduler to work correctly.

Schedule Exclusions
Allows you to add any schedule anomalies (when the Resend Scheduler should not run).

Note: We recommend you leave this parameter blank (that is, do not create any schedule exclusions).

Date Exclusions

>Allows you to add any date anomalies (any date on which the Resend Scheduler should not run).

>**Note:** We recommend you leave this parameter blank (that is, do not create any date exclusions).

Schedule Settings

>Allows you to select a time interval in which the scheduled activity is run every day.

5. After configuring the SWIFTNet7 Adapter Scheduler in the Admin Console, click the **Enable Service for Business Process** check box on the Confirm page to enable the instance.

6. On the Confirm page, verify that the **Enable Service for Business Processes** check box is selected to enable the SWIFTNet7 Adapter Scheduler instance.

Create a New Input Channel

About this task

Each BIC8 that is subscribed to at least one service working in store-and-forward mode has a default/generic input channel that is automatically created by SWIFT. The number of input channels should be limited and most users only use the generic input channel because, for most users, the messaging interface uses the same input channel for all traffic for the different services and applications.

However, if you want to create an additional input channel, a sample business process is provided with the Sterling B2B Integrator.

Note: Any additional input channel may be subject to SWIFT charges, so please contact SWIFT to avoid any unexpected charges.

To create a new input channel, only need to enable the SWIFTNet Server Adapter or SWIFTNet7 Adapter to start the SWIFTNet Server so it can forward the create request to SWIFT. You do not need to configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to use the input channel.

This is the sample business process you can use to create a new input channel for SWIFTNet 6.x:

```
<process name="SWIFTNetCreateInputChannel">
   <sequence name="SWIFTNetCreateInputChannel">
      <operation name="set user token">
         <participant name="SetUserToken"/>
         <output message="SetUserTokenMessage">
            <assign to="USER_TOKEN">admin</assign>
            <assign to="." from="*"></assign>
         </output>
         <input message="inmsg">
            <assign to="." from="*"></assign>
         </input>
      </operation>
      <!-- build Create request -->
      <operation>
         <participant name="SWIFTNetClientService"/>
         <output message="createInputChannelRequest">
            <assign to="." from="*"></assign>
            <assign to="authoriserDN">Put a value here</assign>
            <assign to="inputChannelName">Put a value here</assign>
         </output>
```

```
            <input message="inmsg">
                <assign to="." from="*"></assign>
            </input>
        </operation>
    </sequence>
</process>
```

This is the sample business process you can use to create a new input channel for SWIFTNet 7:

```
<process name="SWIFTNet7CreateInputChannel">
    <sequence name="SWIFTNetCreateInputChannel">
        <operation name="set user token">
            <participant name="SetUserToken"/>
            <output message="SetUserTokenMessage">
                <assign to="USER_TOKEN">admin</assign>
                <assign to="." from="*"></assign>
            </output>
            <input message="inmsg">
                <assign to="." from="*"></assign>
            </input>
        </operation>
        <operation name="Send CreateInputChannelSnFRequest">
        <participant name="SWIFTNet7ClientService"/>
        <output message="createInputChannelRequest">
                <assign to="." from="*"></assign>
                <assign to="authoriserDN">Put a value here</assign>
                <assign to="inputChannelName">Put a value here</assign>
            </output>
            <input message="createInputChannelResponse">
                <assign to="." from="*"></assign>
            </input>
        </operation>
        <operation name="Receive CreateInputChannelSnFResponse">">
        <participant name="SWIFTNet7ClientService"/>
        <output message="handleInputChannelResponse">
                <assign to="." from="*"></assign>
            </output>
            <input message="inputChannelOut">
                <assign to="." from="*"></assign>
            </input>
        </operation>

    </sequence>
</process>
```

To create an input channel:

Procedure

1. Edit the sample business process, SWIFTNetCreateInputChannel (for SWIFTNet 6.x) or SWIFTNet7CreateInputChannel (for SWIFTNet7), and supply the following parameters:

   - **authoriserDN**, which must have at least one RBAC role for store-and-forward.
   - **inputChannelName**, which must be composed of the following:

         inputChannelName = domain "_" component ["!" environment]

   The domain identifies the institution, (that is, the BIC-8 in lowercase text). The component allows identification of different input channels for a specified BIC-8, and you can choose this part of the name. The environment identifies whether the queue is used on ITB, Pilot, or Live, and uses the same naming conventions as the service name.

Input channel names are always in lowercase text (for example, bankus33_system2!x).

2. After you have finished editing, save the business process.

3. Configure the SWIFTNet Client service and the SWIFTNet Server Adapter or SWIFTNet7 Adapter. You do not have to configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to use the input channel; instead, you can select **False** for **Use Input Channel**.

4. Enable the SWIFTNet Server Adapter or SWIFTNet7 Adapter, which starts the SWIFTNet Server. Wait to ensure that the SWIFTNet Server is started.

5. Execute the SWIFTNetCreateInputChannel (for SWIFTNet 6.x) or SWIFTNet7CreateInputChannel (for SWIFTNet7) business process to send the Create Input Channel request.

6. Monitor the execution of the business process. If it is successful, you have successfully created your new input channel. To start using your new input channel, you must configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to use the new input channel name you just created and you must restart the SWIFTNet Server Adapter or SWIFTNet7 Adapter for the change to occur.

Delete an Existing Input Channel

About this task

Once you delete an input channel, you cannot use it. Therefore, you must be careful when you want to delete an input channel because all the message history for the input channel must have already expired. We recommend that you carefully assess whether an input channels might still be in use before deleting it.

Note: The default/generic input channel cannot be deleted.

To delete a new input channel, you do not need to configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to use the input channel. You just need to enable the SWIFTNet Server Adapter or SWIFTNet7 Adapter to start the SWIFTNet Server so it can forward the delete request to SWIFT.

This is the sample business process to delete an existing input channel in SWIFTNet 6.x:

```
<process name="SWIFTNetDeleteInputChannel">
    <sequence name="SWIFTNetDeleteInputChannel">
        <operation name="set user token">
            <participant name="SetUserToken"/>
            <output message="SetUserTokenMessage">
                <assign to="USER_TOKEN">admin</assign>
                <assign to="." from="*"></assign>
            </output>
            <input message="inmsg">
                <assign to="." from="*"></assign>
            </input>
        </operation>
        <!-- build Delete request -->
        <!--  W A R N I N G   N O T E  -->
        <!-- Once deleted, the input channel cannot be re-created or used anymore -->
        <operation>
            <participant name="SWIFTNetClientService"/>
            <output message="deleteInputChannelRequest">
                <assign to="." from="*"></assign>
                <assign to="authoriserDN">Put a value here</assign>
                <assign to="inputChannelName">Put a value here</assign>
```

```
      </output>
      <input message="inmsg">
         <assign to="." from="*"></assign>
      </input>
   </operation>
   </sequence>
</process>
```

This is the sample business process to delete an existing input channel in SWIFTNet 7:

```
<process name="SWIFTNet7DeleteInputChannel">
   <sequence name="SWIFTNetDeleteInputChannel">
      <operation name="set user token">
         <participant name="SetUserToken"/>
         <output message="SetUserTokenMessage">
            <assign to="USER_TOKEN">admin</assign>
            <assign to="." from="*"></assign>
         </output>
         <input message="inmsg">
            <assign to="." from="*"></assign>
         </input>
      </operation>
      <!-- build Delete request -->
      <!--  W A R N I N G   N O T E  -->
      <!-- Once deleted, the input channel cannot be re-created or used anymore -->
      <operation>
         <participant name="SWIFTNet7ClientService"/>
         <output message="deleteInputChannelRequest">
            <assign to="." from="*"></assign>
            <assign to="serverAdapterName">Put a value here</assign>
            <assign to="authoriserDN">Put a value here</assign>
            <assign to="inputChannelName">Put a value here</assign>
         </output>
         <input message="deleteInputChannelResponse">
            <assign to="." from="*"/>
         </input>
      </operation>
      <operation name="Receive DeleteInputChannelSnFResponse">
         <participant name="SWIFTNet7ClientService"/>
         <output message="handleInputChannelResponse">
            <assign to=""." from="*"
         </output>
         <input message="inputChannelOut">
            <assign to="." from="*">
         </input>
      </operation>
   </sequence>
</process>
```

To delete an input channel:

Procedure

1. Edit the sample business process, SWIFTNetDeleteInputChannel or SWIFTNet7DeleteInputChannel, and supply the following parameters:
   - **authoriserDN**, which must have at least one RBAC role for store-and-forward.
   - **inputChannelName**, which must be the existing input channel name.
2. After you have finished editing, save the business process.
3. Configure the SWIFTNet Client or SWIFTNet7 Client service and the SWIFTNet Server Adapter or SWIFTNet7 Adapter. You do not have to configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to use the input channel; instead, you can select **False** for **Use Input Channel**.

4. Enable the SWIFTNet Server Adapter or SWIFTNet7 Adapter, which starts the SWIFTNet Server. Wait to ensure that the SWIFTNet Server is started.

5. Execute the SWIFTNetDeleteInputChannel or SWIFTNet7DeleteInputChannel business process to send the Delete Input Channel request.

6. Monitor the execution of the business process. If it is successful, you have successfully deleted the new input channel.

Opening an Input Channel

You can either open an input channel automatically or manually.

Open an Input Channel Automatically

About this task

The SWIFTNet Server Adapter or SWIFTNet7 Adapter allows you to configure an input channel to be opened when the adapter is enabled.

To open an input channel automatically:

Procedure

1. Configure the SWIFTNet Server Adapter or SWIFTNet7 Adapter to use the input channel. See "Configure the SWIFTNet Server Adapter and Resend Scheduler for SWIFTNet 6.x" on page 137 or"Configure the SWIFTNet7 Adapter for SWIFTNet 7" on page 152 for more information.

2. Enable the SWIFTNet Server Adapter or SWIFTNet7 Adapter.

3. Access the Current Process page from the **Administration** menu by selecting **Business Processes** > **Current Processes** so you can monitor the business process execution. The SWIFTNet Server triggers a callback to the Sterling B2B Integration, which eventually bootstraps the predefined SWIFTNetOpenInputChannel or SWIFTNet7OpenInputChannel business process to open the input channel. When both the SWIFTNetOpenInputChannel or SWIFTNet7OpenInputChannel business process and the handleSWIFTNetOpenInputChannel or SWIFTNet7OpenInputChannel business process have completed successfully, the input channel has been successfully opened. If an error occurs, you should check the input channel configuration in the SWIFTNet Server Adapter or SWIFTNet7 Adapter.

Results

Once the input channel is open, you can send message using this input channel. The resend scheduler works automatically in the background until the SWIFTNet Server Adapter or SWIFTNet7 Adapter is disabled.

Open an Input Channel Manually

About this task

You can open the input channel manually if you create and invoke a business process similar to the SWIFTNetOpenInputChannel or SWIFTNet7OpenInputChannel business process.

Note: Do not modify the SWIFTNetOpenInputChannel or SWIFTNet7OpenInputChannel business process because that may affect your ability to automatically open an input channel.

The business process you will need to create must rely on the input channel configuration in the SWIFTNet Server Adapter or SWIFTNet7 Adapter and you must have already configured this adapter with the appropriate input channel settings.

Note: This manual operation will not function if the SWIFTNet Server Adapter or SWIFTNet7 Adapter is not configured with the correct input channel settings or the adapter is not enabled.

You may want to manually open an input channel in these circumstances:
- When automatically opening the input channel fails because there is a gap in the channel (that is, the channel was not closed properly) and the SWIFTNet Server Adapter or SWIFTNet7 Adapter was configured to open in Normal mode, which will cause the open request to fail because it will need to be opened in Force mode. To resolve this issue, you can manually invoke the SWIFTNetManualOpenInputChannel business process (see example below) with the Force mode equal to TRUE.
- When the SWIFTNet Client or SWIFTNet7 Client service encounters an error when sending a message using the input channel, and this sets the status to LOCKED. To resolve this issue, you can manually invoke the SWIFTNetManualOpenInputChannel business process (see example below) with the Force mode equal to TRUE.
- When you close the input channel manually and need to open the input channel again. In this situation you can manually invoke the SWIFTNetManualOpenInputChannel business process (see example below) with the Force mode equal to FALSE.

You can supply values for the forceOpen parameter (this is an optional parameter and the default value is FALSE) and the serverAdapterName parameter (this is an optional parameter and the default value is SWIFTNetServerAdapter) in the business process above. The forceOpen parameter indicates whether to open in Force mode (if set to TRUE) and the serverAdapterName parameter indicates where to get the input channel configuration from (from which instance of the SWIFTNet Server Adapter or SWIFTNet7 Adapter). If you are using the default SWIFTNet Server Adapter or SWIFTNet7 Adapter, you do not need to specify this parameter.

Note: The adapter must be enabled when you execute this business process.

To open an input channel manually:

Procedure
1. Configure and enable the SWIFTNet Server Adapter or SWIFTNet7 Adapter.
2. Create a new business process based on the example above and name it SWIFTNetManualOpenInputChannel.
3. Supply the following parameters in the business process:
   - **forceOpen**, which indicates whether to open in Force mode (if set to TRUE). This is an optional parameter and the default value is FALSE.
   - **serverAdapterName**, which indicates where to get the input channel configuration from (from which instance of the SWIFTNet Server Adapter or SWIFTNet7 Adapter). If you are using the default SWIFTNet Server Adapter or SWIFTNet7 Adapter, you do not need to specify this parameter. This is an optional parameter and the default value is SWIFTNetServerAdapter.
4. After you have finished editing, save the business process.

5. Execute the business process you created, SWIFTNetManualOpenInputChannel, to send the Open Input Channel request.
6. If the business process successfully completes, the input channel has been opened successfully. If an error occurs, you should check the input channel configuration in the SWIFTNet Server Adapter or SWIFTNet7 Adapter.

Results

Once the input channel is open, you can send messages using this input channel. The resend scheduler should work automatically in the background until the SWIFTNet Server Adapter or SWIFTNet7 Adapter is disabled.

Closing the Input Channel

You can either close the input channel automatically or manually.

Close the Input Channel Automatically
About this task

The SWIFTNet Server Adapter or SWIFTNet7 Adapter allows the input channel to be closed when the adapter is disabled.

To close an input channel automatically:

Procedure
1. Disable the SWIFTNet Server Adapter or SWIFTNet7 Adapter and check the message shown on the Service Configuration page. The message will depend on the current input channel status. For example, if the message indicates that the status is OPEN and will be disabled automatically after any gaps are resolved, this means the adapter delayed the shutdown process to ensure that any gaps in the input channel are resolved and all current messages receive their acknowledgements. The resend scheduler automatically handles the synchronization during this closing process. Once all gaps are resolved, the resend handler invokes the predefined SWIFTNetCloseInputChannel or SWIFTNet7CloseInputChannel business process to close the channel.
2. You can monitor the Current Process page if you wish. Access the Current Process page from the **Administration** menu by selecting **Business Processes** > **Current Processes** so you can monitor the business process execution.

Results

When the business process completes successfully, the input channel is closed successfully, as well. If you see an error, you may need to check the error details and try to close the input channel manually.

Close the Input Channel Manually
About this task

You can also close the input channel manually (that is, without shutting down the SWIFTNet Server Adapter or SWIFTNet7 Adapter) by creating and invoking a business process similar to the SWIFTNetCloseInputChannel or SWIFTNet7CloseInputChannel business process.

Note: Do not modify the SWIFTNetCloseInputChannel or SWIFTNet7CloseInputChannel business process because this may affect the automatic opening of the input channel.

The business process you create must rely on the input channel configuration in the SWIFTNet Server Adapter or SWIFTNet7 Adapter and the adapter must be enabled with the appropriate input channel settings.

To close the input channel manually:

Procedure

1. Configure (with the appropriate input channel settings) and enable the SWIFTNet Server Adapter or SWIFTNet7 Adapter.
2. Create a new business process based on the example above and name it SWIFTNetManualCloseInputChannel.
3. Supply the following parameter in the business process: **serverAdapterName**, which indicates where to get the input channel configuration from (from which instance of the SWIFTNet Server Adapter or SWIFTNet7 Adapter). If you are using the default SWIFTNet Server Adapter or SWIFTNet7 Adapter, you do not need to specify this parameter. This is an optional parameter and the default value is SWIFTNetServerAdapter. This instance of the adapter must be enabled when you execute the business process.
4. After you have finished editing, save the business process.
5. Execute the business process you created, SWIFTNetManualCloseInputChannel, to send the Close Input Channel request.
6. If the business process successfully completes, the input channel has been closed successfully. If an error occurs, it may indicate that your request to close the input channel is pending until any gaps are resolved. If this occurs, the resend scheduler attempts to resolve any gaps and the input channel will be closed automatically once all gaps resolved. If you try to send message(s) while the resend scheduler is processing, the messages are stored on local processing. Once the channel is closed, if you need to open it again, see "Open an Input Channel Manually" on page 174 for more information.

Send an InterAct Message Using the Input Channel

About this task

Once the input channel is open, you can use the SWIFTNet Client service (for SWIFTNet 6.x) or the SWIFTNet Client Profile (for SWIFTNet7) to send messages using the input channel.

Note: You can only use the input channel to send InterAct messages in store-and-forward mode.

To send an InterAct message using the input channel:

Procedure

1. Configure the SWIFTNet Client service (for SWIFTNet 6.x) or the SWIFTNet Client Profile (for SWIFTNet7) (please refer to "Configuring the Sterling B2B Integrator Components to Use the Input Channel" on page 137) to use the input channel. You must also be sure that the interface used is InterAct and store-and-forward mode.

Alternatively, you can pass the necessary parameter through the BPML. The following is a sample business process for SWIFTNet 6.x that you can use to send messages using the input channel:

```
<process name="SWIFTNet-IA-Sync-SnF-InputChannel">
  <sequence name="SWIFTNetClientService">
    <operation name="set user token">
      <participant name="SetUserToken"/>
      <output message="SetUserTokenMessage">
        <assign to="USER_TOKEN">admin</assign>
        <assign to="." from="*"></assign>
      </output>
      <input message="inmsg">
        <assign to="." from="*"></assign>
      </input>
    </operation>

    <operation>
      <participant name="SWIFTNetClientService"/>
      <output message="handleClientRequest">
        <assign to="." from="*"></assign>
        <assign to="interfaceMode">interact</assign>
        <assign to="swiftOp">sync</assign>
        <assign to="requestorDN">o=test,o=swift</assign>
        <assign to="responderDN">o=test,o=swift</assign>
        <assign to="serviceName">swift.generic.iast!x</assign>
        <assign to="SnF">TRUE</assign>
        <assign to="nonRepudiation">FALSE</assign>
        <assign to="deliveryNotification">TRUE</assign>
        <assign to="useInputChannel">TRUE</assign>
        <assign to="serverAdapterName">SWIFTNetServerAdapter</assign>
      </output>
      <input message="testing">
        <assign to="." from="*"></assign>
      </input>
    </operation>

  </sequence>
</process>
```

The parameters used in the BPML are very similar to the parameters used to send messages in InterAct store-and-forward mode without using the input channel.

However, you need to assign a new parameter, **useInputChannel**, to TRUE if you want to use the input channel. You also can supply the serverAdapterName parameter (the parameter is optional and the default value is SWIFTNetServerAdapter). The serverAdapterName indicates from which instance of the SWIFTNet Server Adapter or SWIFTNet7 Adapter the input channel configuration should be taken. If you are using the default SWIFTNet Server Adapter or SWIFTNet7 Adapter, you do not need to specify this parameter.

2. Configure the SWIFTNet Server Adapter (please refer to "Configuring the Sterling B2B Integrator Components to Use the Input Channel" on page 137) to use the input channel.
3. Enable the SWIFTNet Server Adapter and monitor the Current Process page to verify that the input channel is successfully opened.
4. Execute the sample business process above (with the appropriate payload) to send the message using the input channel.

Results

If an error occurs, the resend scheduler automatically resends the message. If the retry count has exceeded the maximum retry attempts setting in the SWIFTNet Server adapter, the resend scheduler automatically sends a Resolve Gap request to notify SWIFT to skip this sequence number and proceed with the subsequent number.

Chapter 9. SWIFTNet Output Channel

The SWIFTNet 7.0 release introduces output channels and queue sharing. In previous versions, a queue could not be shared by different messaging interfaces at the same time because SWIFT only allowed one queue session to be opened at one time.

By introducing the output channel in version 7.0, SWIFT allows queue sharing by enabling a queue to use different output channels and the interface to open several output channels on the same queue at the same time.

To support the use of output channel, the application includes the following functionality:

- Ability to create a new output channel
- Ability to delete an existing output channel
- Ability to open an output channel
- Ability to close an output channel

Configuring the Sterling B2B Integrator Components to Use the Output Channel

You must configure the following three Sterling B2B Integrator components to use the output channel:

- SWIFTNet7 Adapter
- SWIFTNet7 Adapter Scheduler
- SWIFTNet7 Client Service (including configuring a predefined business process to create a new output channel)

Configure the SWIFTNet7 Adapter for SWIFTNet 7
About this task

To configure the SWIFTNet7 Adapter:

Procedure

1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet7 Adapter or select it from the list and click **Go!**.
3. Click **Edit**.
4. Specify field settings in the Admin Console.

   Note: Specify failover processing to ensure that failover is supported if a SAG connection fails by configuring **failover-SAG**.

   Use the field definitions in the following table to create a new configuration of the SWIFTNet7 adapter, or to set up the configuration provided with the application.

   Note:

The business entities (accessible through the Business Entities wizard as part of the SWIFTNet7 adapter configuration) are shared by both RA1 and RA2. The Business Entities wizard enables you to add multiple entities.

Field   Description

Name   Unique and meaningful name for the adapter configuration. Required.

Description
Meaningful description for the adapter configuration, for reference purposes. Required.

Environment
Select the environment in which the adapter will run. Required.

Note: The adapter is not eligible to run in the adapter container.

Select a Group
Select one of the options:
- None – Do not include the configuration in a service group at this time.
- Create New Group – Enter a unique name for a new group, which will be created with this configuration. (You can then add other services to the group as well.)
- Select Group – If service groups already exist for this service type, they are displayed in the list. Select a group from the list.

SI Server Address
The callback IP of Sterling B2B Integrator for the SWIFTNet Servers. Required.

Note: The default value is the IP address of the machine where the Sterling B2B Integrator is installed.

SI HTTP Server Adapter Port
This is the listening port for the SWIFTNet7 HTTP Server Adapter. Required. The default populated value is the instance port number of the Sterling B2B Integrator instance plus 53. For example, if the Sterling B2B Integrator instance port is 34600, the listening port populated by default is 34653.

Note: The HTTP Server adapter functions between the SWIFTNet7 adapter and the SWIFTNet Servers. For an SSL connection, this value should be server name because the certificate is made with the server name.

Note: If you are using the SWIFTNet7 adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

MEFG SWIFTNet Address
The IP address of the SWIFTNet Servers. Required.

MEFG SWIFTNet Port
The port of the SWIFTNet Servers. Required.

CLA2Client Listening Port

The listening port used by the client command adapter (CLA2Client) running along the SWIFTNet Servers. Required.

Note: This port listens for requests to start and stop the SWIFTNet Servers.

MEFG SWIFTNet Home

The home directory of the SWIFTNet Servers. Required.

Start MEFG SWIFTNet despite errors

Whether to start the SWIFTNet Servers if errors occur. The default is unchecked (do not start the SWIFTNet Servers if errors occur). Optional.

Use SSL

Whether to enable Secure Socket Layer (SSL) over HTTP communication between the Sterling B2B Integrator and the SWIFTNet Servers. Valid values are False (default) and True. Select **True** to use SSL with an Input Channel.

Event Status Tracking

The event status tracking for the adapter. Required. Valid selections are Minimal (Only Completed, Rejected, Duplicated status) - this is the default, and Full.

Cipher Strength

Specifies the strength of the algorithms (cipher suites) used to encrypt data. Valid values are:

- STRONG - Required if Use SSL is Must
- ALL - All cipher strengths are supported
- WEAK - Often required for international trade, because government regulations prohibit STRONG encryption from being exported

Default is ALL. Required if **Use SSL** is checked.

CA Certificate

Move one or more CA Certificates to the use column. These are the digital security certificates that the SSL server will use to authenticate the client. Required if SSL is selected.

Message Partner Client Name

The client message partner name that the SNL server application recognizes for the SWIFTNet Servers client application.

Note: The Message Partner Client Name must correspond to the Application Interface Message Partner that is defined on the SAG as the client interface for the SWIFTNet Servers.

Message Partner Server Name

The server message partner name that the SNL server application recognizes for the SWIFTNet Servers server application.

Note: The Message Partner Server Name must correspond to the Application Interface Message Partner that is defined on the SAG as the client interface for the SWIFTNet Servers.

Delivery Notification

Determines whether the server requests a delivery notification when a business partner is downloading. Possible values are True and False (default). Required.

Delivery Notification Request Type

The request type of the delivery notification is the value SI SWIFTNet Server uses in the response after getting a request from a remote client. Required.

Configure for failover SAG?

Enables you to set up Failover SAG Configuration using two separate instances of the Remote API (RA), RA1 and RA2. Each RA should be configured to point to a different SAG to support failover processing. Possible values are True and False (default). Required.

Note: This parameter specifies whether to support failover if one SAG fails. When this parameter is set to True, you are presented with parameters for both an RA1 Profile and an RA2 Profile.
When you are operating in an environment with multiple SAGs configured in failover SAG mode, setting this parameter enables you to define an alternate RA connection to a secondary SAG for failover support.

Delivery Notification

Determines whether the RA1 server is handling a delivery notification. Possible values are True and False (default). Optional. This is used for a FileAct Get.

Delivery Notification DN

Distinguished name of the responder of the delivery notification. This is used for a FileAct Get. Optional.

Request Type of Del. Notifn

Request type of the delivery notification. This is used for a FileAct Get. Optional.

Return Signature List

Whether you want to receive your own signature to safe store it in case of any dispute in the future. Valid values are False or True. Applicable only if Signature List is used. Optional for T-Copy and Y-Copy implementation.

SWIFTNet RA

The absolute path of the RA1 installation directory for RA1 SWIFTNet. Required. For example, **/SWIFTAlliance/RA**.

Note: This parameter specifies where to pick up the remote API and execute to SAG.

Config

The relative path of the RA1 instance configuration directory (relative to the RA installation directory). Required. For example, **RA1/cfg**.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process

(replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Bin This is added to the PATH environment variable to contain the SWIFTNet Server binaries. Possible value is bin. Required.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Lib This is added to the library path environment variable. Possible value is lib. Required.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Category
This is the category of RA. Possible values are:
- RA (SNL facade library to access an SAG)
- SNL (a native SNL interface)
- DEFAULT (default set for the RA1 instance)

Required.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Delivery Responder DN
The distinguished name of the responder to which delivery notifications requested by the sender are sent. Optional.

Note: If left blank, Delivery Notifications requested by the server are sent to the responder indicated in the message; otherwise, it is sent to this responder

SWIFTNet RA
The absolute path of the RA2 installation directory for RA2 SWIFTNet. Required (based on Failover SAG Configuration). For example, **/SWIFTAlliance/RA**.

Note: This parameter is only displayed if **Failover SAG Configuration** is set to True.

Config

The relative path of the RA2 instance configuration directory (relative to the RA2 installation directory). Required (based on Failover SAG Configuration). For example, **/RA2/cfg**.

Note: This parameter is only displayed if **Failover SAG Configuration** is set to True.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Bin   This is added to the PATH environment variable to contain the SWIFTNet Server binaries. Required (based on Failover SAG Configuration).

Note: This parameter is only displayed if **Failover SAG Configuration** is set to True.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Lib   This is added to the library path environment variable. Required (based on Failover SAG Configuration).

Note: This parameter is only displayed if **Failover SAG Configuration** is set to True.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Category

This is the category of RA2. Possible values are:
* RA (SNL facade library to access an SAG)
* SNL (a native SNL interface)
* DEFAULT (default set for the RA1 instance)

Required (based on Failover SAG Configuration).

Note: This parameter is only displayed if **Failover SAG Configuration** is set to True.

Note: If you are using the SWIFTNet7 Adapter in your current installation, prior to installing a new version of the Standards Library, you need to note the value you have configured for this parameter. This parameter may be overwritten during the upgrade process (replaced with the default value). If this parameter is overwritten, you need to restore it to the original value after the upgrade process is complete.

Delivery Responder DN

The responder to which delivery notifications requested by the sender are sent. Required (based on the failover SAG configuration).

Note: If left blank, Delivery Notifications requested by the server are sent to the responder indicated in the message; otherwise, it is sent to this responder This parameter is only displayed if failover SAG Configuration is set to True.

New Security Context

Click add to create a new security context for the Output Channel Configuration or click edit to modify an existing entry.

Note: You must have at least one security context created to proceed. This is the authorization context to open the output channel.

UserId

The user identifier for this business entity (to log in to SWIFTNet). Required for each configured entity.

Password (RA1)

The user password for this security context for RA1. Required for each configured entity.

Note: This password is automatically encrypted. This parameter is only displayed if you edit an existing Security Context or add a new Security Context.

Password (RA2)

The user password for this security context for RA2. If not specified, the RA2 password is considered to be equal to the RA1 password. Optional.

Note: This password is automatically encrypted. This parameter is only displayed if you edit an existing Security Context or add a new Security Context.

New Business Entity

Click **add** to create a new business entity or click **edit** to modify an existing entity.

Note: You must have at least one business entity created to proceed.

Entity Identifies the security context to be used. For the client, the business entity is the requester. For the server, the business entity is the responder.

The SWIFTNet7 Adapter enhances business entities. You are not required to define an exact match for every distinguished name used for requestor or responder. The SWIFTNet Client Adapter uses the best matching algorithm to find the business entity and associated context. Therefore, if all the distinguished names used belong to the same

organization (for example: o=abcdefgh,o=swift), it is sufficient for you to define one and only one business entity: o=abcdefgh,o=swift. However, if necessary, (for example if you want to ensure that cn=xyz,o=abcdefgh,o=swift will use different context), you may define another business entity for this particular distinguished name.

Note: This is the distinguished name created by SWIFT. This parameter is only displayed if you edit an existing Business Entity or add a new Business Entity. The business entities are shared by both the RA1 and RA2 profiles.

UserId

The user identifier for this business entity (to log in to SWIFTNet). Required for each configured entity.

Note: The UserName is created in SAG (in the Users Module) and must also have a certificate created for it in the SAG. This parameter is only displayed if you edit an existing Business Entity or add a new Business Entity. The SWIFTNet Client Adapter allows you to use a different security context for RA1 and RA2.

UserId for RA2 (if different)

The user identifier for the RA2 business entity (to log in to SWIFTNet). Required for each configured entity.

Note: The UserName is created in SAG (in the Users Module) and must also have a certificate created for it in the SAG. This parameter is only displayed if you edit an existing Business Entity or add a new Business Entity and only displayed if Failover SAG Configuration is enabled.

Use Default Delivery Notification

Indicates whether to use the default delivery notification configuration. Required.

Delivery Notification (Del. Notifn)

Indicates whether the sender asked the receiver to send a delivery notification. Optional. Valid values are True (default) or False.

Note: This parameter is only available when Use Default Delivery Notification is not selected.

Request Type of Del. Notifn

If Delivery Notification (Del. Notifn) is set to True, the value of this parameter is used to request a specific delivery notification message from the remote receiving server application when it returns the delivery notification. Optional.

Note: This parameter is only available when Use Default Delivery Notification is not selected.

Use Input Channel (for InterAct Store and Forward only)

Whether to use the input channel with this adapter. Valid values are False (default) and True. You must select True if you are using an input channel. Required.

Note: Used for InterAct store-and-forward only. If you configure this parameter, the SWIFTNet Servers opens the Input Channel automatically during the startup (when the SWIFTNet7 Adapter is

enabled). This Input Channel remains open until the SWIFTNet Servers is shut down (or the SWIFTNet7 Adapter is disabled). During this time, you still have an option to send message using the input channel or without the input channel. All you need to do is to indicate this by using this parameter in SWIFTNet Client profile.

Use Output Channel (for Store and Forward only)

Whether to use the output channel with this adapter. Valid values are False (default) and True. You must select True if you are using an output channel. Required.

Note: Used for store-and-forward only.

Input Channel Name

The name of the input channel. Required only if you specified **True** for **Use Input Channel**.

Authoriser DN

The authorized distinguished name that will be used to open the input channel. Required only if you specified **True** for **Use Input Channel**.

Force Open the Input Channel

Whether to force open the input channel or use normal mode. Valid values are False (use normal mode, which is the default) and True (force the input channel). Required only if you specified True for Use Input Channel.

Max. Resend Attempts

The maximum number of retries that should be attempted before sending the Resolve Gap request. Required. The default is three.

New Queue

Click **add** to create a new queue or click **edit** to modify an existing entry.

Note: If you are using an Output Channel, you must have at least one queue created to proceed.

Queue Name

The name of the queue you are creating. Optional.

Output Channel Name

The name of the output channel. Required if you are using an Output Channel.

Note: This name must be unique and can only be associated with one SWIFTNet7 Adapter instance. By default there is an output channel with the same name as the queue name. Alternatively, you may create the output channel first using the utility provided.

Operating DN

The distinguished name of the operator. Optional.

SNL Endpoint

The endpoint for SNL. Optional.

Traffic Filter #1

Traffic filter to be applied first. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Traffic Filter #2
> Traffic filter to be applied second. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Traffic Filter #3
> Traffic filter to be applied third. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Traffic Filter #4
> Traffic filter to be applied fourth. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Traffic Filter #5
> Traffic filter to be applied fifth. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Traffic Filter #6
> Traffic filter to be applied sixth. Optional. Valid values are System, InterAct, FileAct, Urgent, Normal, System_Urgent, System_Normal, InterAct_Urgent, InterAct_Normal, FileAct_Normal, FileAct_Urgent, Not Applicable.

Reception Directory
> The full directory path where the file is received and stored during FileAct Put mode. Required for FileAct. Optional.

Download Directory
> The full directory path where the file is picked up and sent to the requestor during FileAct Get mode. Required for FileAct. Optional.

Enable Service for Business Process
> Whether the adapter is enabled to be used with business processes. Optional.

5. After configuring the SWIFTNet7 Adapter in the Admin Console, click the **Enable Service for Business Process** check box on the Confirm page to enable the instance.

6. Once the SWIFTNet7 Adapter is configured and saved, click the **Enabled** check box on the Services Configuration page. This starts the SWIFTNet Servers. You should wait to ensure that the SWIFTNet Servers starts.

7. On the Confirm page, verify that the **Enable Service for Business Processes** check box is selected to enable the adapter instance.

Configure the SWIFTNet7 Client Service
About this task

To configure the SWIFTNet7 Adapter:

Procedure

1. Select **Deployment** > **Services** > **Configuration**.
2. Search for SWIFTNet7 Client service or select it from the list and click **Go!**.

3. Click **Edit**.
4. Specify field settings in the Admin Console.

   Note: Each instance of the SWIFTNet7 Client service is configured for a pair of requestor/responder DNs and the SWIFTNet7 Client service name.

   Use the field definitions in the following table to create a new configuration of the SWIFTNet7 Client service.

   Name   Unique and meaningful name for the service configuration. Required.

   Description
   > Meaningful description for the service configuration, for reference purposes. Required.

   Select a Group
   > Select one of the options:
   > - None – Do not include the configuration in a service group at this time.
   > - Create New Group – Enter a unique name for a new group, which will be created with this configuration. (You can then add other services to the group as well.)
   > - Select Group – If service groups already exist for this service type, they are displayed in the list. Select a group from the list.

   Store and Forward
   > Indicates if the file transfer is done using the store-and-forward method. Valid values are True (use Store-and-Forward) and False (default—do not use Store-and-Forward). Required. BPML element value is SnF.

   SWIFTNet Operation
   > The SWIFTNet operation to send an InterAct or FileAct message. Possible values are:
   > - Synchronous (default)—InterAct
   > - Asynchronous—InterAct
   > - Put—FileAct
   > - Get—FileAct
   >
   > Required. When SWIFTNet Operation is FileAct, you must select either Put or Get. If you do not select an operation, the service uses Synchronous as the default value. BPML element value is sync (default) or async for InterAct, or Put or Get for FileAct.

   Requestor DN
   > Distinguished name of the requestor. Required. BPML element value is requestorDN. This DN must be registered with the SAG instance using SWIFTNet Alliance Webstation.

   Responder DN
   > Distinguished name of the responder. Required. BPML element value is responderDN. Note: This DN must be registered with the SAG instance using SWIFTNet Alliance Webstation.

   Service Name
   > Name of the service to which both SWIFT correspondents have subscribed. Required. BPML element value is serviceName. This must be a SWIFTNet service to which you are subscribed.

Authoriser DN

The distinguished name of the authorizing party. Optional.

This service allows Third Party Copy

Whether this service uses T-Copy or Y-Copy (check your service agreement with SWIFT). BPML element value is thirdPartyCopy. Valid values are TRUE or FALSE. This parameter is displayed only if you selected File Act and True for Store and Forward on SWIFTNet7 Client service Interface page.

Note: If the Copy Mode is Y-Copy, Sterling B2B Integration sends an authorization message, which is like sending an Interact store-and-forward request. The SWIFTNet7 Client service is used, but you must set the This service allows Third Party Copy parameter to TRUE, and provide the authorization decision (either Authorised or Refused) for the AuthDecision parameter.

Request for Third Party Copy

Whether you are requesting third party copy. When the Copy feature is defined as Optional in the service agreement, you can choose whether you want the Third Party Copy to occur. BPML element value is copyIndicator. Valid values are TRUE or FALSE. Displayed only if you select True for This service allows Third Party Copy. This parameter is displayed only if you selected True for **This service allows Third Party Copy**.

Request for Notification from Third Party

In T-Copy mode, this setting is not applicable, the value should always be set to FALSE.

In Y-Copy mode, when the Authorisation Notification Indicator feature is available and defined as Optional in the service agreement, you can choose whether you want to receive the Authorisation Notification messages. BPML element value is authNotifIndicator. Valid values are TRUE or FALSE. Displayed only if you selected True for This service allows Third Party Copy. This parameter is displayed only if you selected True for This service allows Third Party Copy.

Request Type

Request type supported by the message exchange. Optional for InterAct and required for FileAct in SWIFTNet 6.0. BPML element value is requestType. Note: In SWIFTNet 6.0 FileAct the format convention is as follows:

```
<business_area>.<type_of_syntax>.<detailed_syntax_and_format>
```

This format starts with a four-characters business area code, followed by a period (dot), followed by a three-character code that designates the type of syntax (which can be <nnn>, FIN, or xxx), followed by another period (dot), and then followed by a more detailed indication of syntax and format.

Request Reference

User reference of the request. Optional. BPML element value is requestReference.

Non Repudiation Required

Indicates whether non-repudiation is required. Possible values are True (when enabled this means that trading partners cannot deny that they

sent a request) or False (default—when enabled this indicates that non-repudiation is not required). Optional. BPML element value is nonRepudiation.

Switch to SnF mode when real-time transmission failed
Indicates whether you want to switch to store-and-forward mode if a real-time transmission (InterAct or a FileAct Put) has failed. Possible values are True or False (default). Required. BPML element value is switchToSnF.

Store and Forward Service Name
The name of the store-and-forward service. Required when Switch to SnF mode when real-time transmission failed is set to True. BPML element value is SnFServiceName.

End-to-End Signature Required
Whether an end-to-end signature is required. Valid values are False (default) and True. Optional. You can use an end-to-end signature regardless of whether you are using non-repudiation (for example, for SWIFT SCORE messages).

Number of Retries
Number of retries to connect to SAG. Default value is 3. Optional. BPML element value is numOfRetries.

Retry Delay (in seconds)
Time that will elapse before the next retry. Default value is 60 (seconds). Optional. BPML element value is secInRetryDelay.

Trace Trace for logging purposes in the SWIFTNet Server. Valid values are True and False (default). Required. BPML element value is trace.

Use Signature List
Whether to use a signature list. This enables you to select your own signatures. If you do not use a signature list then normal Crypto is used. Valid values are False and True. Required. This parameter is displayed only if you selected True for End-to-End Signature Required.

Return Signature List
Whether to return a signature list. Valid values are False and True. Required. If you want a signature list returned, the SWIFTNet Server receives the requestor's own signature in the response message. This returned signature will be extracted and saved as a separate message. This message is stored in the database and is made available for Correlation search. This parameter is displayed only if you selected True for End-to-End Signature Required.

Use RND
Whether to use RND (digest reference values that terminate on "and RND"). Valid values are False (default) and True. Required. This parameter is displayed only if you selected True for End-to-End Signature Required.

Delivery Notification (Del. Notifn)
Indicates that the sender asked the receiver to send a delivery notification. Possible values are True or False (default). Optional. BPML element value is deliveryNotification.

Note: This parameter is only displayed when you select True for Store and Forward or are performing a FileAct Put. If you are performing a Put operation, you can request the responder to send you a delivery

notification and specify a different Delivery Notification DN and Request Type of Delivery Notification, if desired. If you are performing a Get operation, the responder can request Delivery Notification from the requestor after receiving the file. That setting for delivery notification is configured through the SWIFTNet7 Adapter.

Request Type of Delivery Notification
Used to request a specific delivery notification message from the remote receiving server application when it returns the delivery notification (when Delivery Notification is set to True). Optional. BPML element value is requestTypeDelNotifn. This parameter is only displayed when you select True for Store and Forward or a FileAct Put.

Message Priority
Indicates priority handling in the queue for store-and-forward only. Valid values are Normal (default) and Urgent. Optional. BPML element value is messagePriority. This value is used as a selection criterion when delivering messages from a queue, and in SWIFTNet FileAct to influence the pace of the FileAct flow.

Use Input Channel
Whether to use the input channel. Valid values are False (default) and True. Required. This parameter is displayed only if you selected True for Store and Forward and InterAct for SWIFTNet interface.

Note: Used for InterAct store-and-forward only. Select True if you are using an input channel. If you configure this parameter, the SWIFTNet Server opens the Input Channel automatically during the startup (when the SWIFTNet7 Adapter is enabled). This Input Channel remains open until the SWIFTNet Server is shut down (or the SWIFTNet7 Adapter is disabled). During this time, you still have an option to send message using the input channel or without the input channel. All you need to do is to indicate this by using this parameter in SWIFTNet7 Client service.

MEFG SWIFTNet IP
The IP address for the SWIFTNet Server. Required.

MEFG SWIFTNet Port
The port for the SWIFTNet Server. Default is NULL. Optional.

Response Timeout
The timeout interval (in seconds) in which a response must be received or the message operation fails. Optional. Default is 60 seconds.

Use SSL
Whether to enable Secure Socket Layer (SSL) over HTTP communication between Sterling B2B Integration and the SWIFTNet Servers. Valid values are None and Must. Note: Regardless of the value you select for Use SSL, you must also update the business processes associated with the SWIFTNet Client service.

Cipher Strength
Indicates the strength of the cipher. Possible values are ALL (default), WEAK, and STRONG. Optional.

CA Certificate
The CA certificate of the SWIFTNet Server.

Note: This is the public key certificate that must be configured to set up the outbound SSL channel. This page is only displayed if you set Use SSL to Must.

Note: The SWIFTNet7 Client service Configuration page allows you to select the same CA Certificate for SSL processing a second time, and continues to allow additional selections of the same certificate in subsequent edits. If you have already selected a CA Certificate once for a configuration of the SWIFTNet7 Client service, do not select the same CA Certificate again, as this will result in an error when you execute the relevant business process.

Switch to SnF mode when real-time transmission failed
Whether to switch to store-and-forward mode when real-time transmission fails. Select True if you want to switch to Store and Forward mode when the real-time transmission (InterAct and FileAct Put) has failed. Valid values are True and False.

Physical Filename
Physical name of the file to send. Required if put or get is selected for the SWIFTNet Operation. BPML element value is physicalFilename.

Logical Filename
Logical name of the file to send. This name is communicated to the SWIFTNet Server. By default, this name is the Physical Filename without the path. Optional. BPML element value is logicalFilename.

File Information
User information about the file transfer. Optional. BPML element value is fileInfo.

File Description
User description about the file transfer. Optional. BPML element value is fileDesc.

SWIFTNet Interface
SWIFTNet message type. Valid values are InterAct or FileAct. Required.

5. On the Confirm page, verify that the **Enable Service for Business Processes** check box is selected to enable the service instance.

Create a New Output Channel

About this task

Each BIC8 that is subscribed to at least one service working in store-and-forward mode has a default/generic output channel that is automatically created by SWIFT. The number of output channels should be limited and most users only use the generic input channel because, for most users, the messaging interface uses the same input channel for all traffic for the different services and applications.

However, if you want to create an additional output channel, a sample business process is provided with the Sterling B2B Integrator.

Note: Any additional output channel may be subject to SWIFT charges, so please contact SWIFT to avoid any unexpected charges.

To create a new output channel, you only need to enable the SWIFTNet7 Adapter to start the SWIFTNet Server so it can forward the create request to SWIFT. You do not need to configure the SWIFTNet7 Adapter to use the output channel.

This is the sample business process you can use to create a new input channel for SWIFTNet 7:

```
<process name="SWIFTNet7CreateOutputChannel">
    <sequence name="SWIFTNetCreateOutputChannel">
        <operation name="set user token">
            <participant name="SetUserToken"/>
            <output message="SetUserTokenMessage">
                <assign to="USER_TOKEN">admin</assign>
                <assign to="." from="*"></assign>
            </output>
            <input message="inmsg">
                <assign to="." from="*"></assign>
            </input>
        </operation>
        <operation name="Send CreateOutputChannelSnFRequest">
        <participant name="SWIFTNet7ClientService"/>
        <output message="createOutputChannelRequest">
                <assign to="." from="*"></assign>
                <assign to="authoriserDN">Put a value here</assign>
                <assign to="OutputChannelName">Put a value here</assign>
            </output>
            <input message="createOutputChannelResponse">
                <assign to="." from="*"></assign>
            </input>
        </operation>
        <operation name="Receive CreateOutputChannelSnFResponse">">
        <participant name="SWIFTNet7ClientService"/>
        <output message="handleOutputChannelResponse">
                <assign to="." from="*"></assign>
        </output>
        <input message="outputChannelOut">
            <assign to="." from="*"></assign>
        </input>
        </operation>

    </sequence>
</process>
```

To create an output channel:

Procedure

1. Edit the sample business process, SWIFTNet7CreateOutputChannel or and supply the following parameters:

   - **authoriserDN**, which must have at least one RBAC role for store-and-forward.

   - **outputChannelName**, which must be composed of the following:

         outputChannelName = domain "_" component ["!" environment]

   The domain identifies the institution, (that is, the BIC-8 in lowercase text). The component allows identification of different input channels for a specified BIC-8, and you can choose this part of the name. The environment identifies whether the queue is used on ITB, Pilot, or Live, and uses the same naming conventions as the service name.

   Output channel names are always in lowercase text (for example, bankus33_system2!x).

   - **desc**, which is the name description.

2. After you have finished editing, save the business process.

3. Configure the SWIFTNet7 Client service. You do not have to configure the SWIFTNet7 Adapter to use the output channel; instead, you can select **False** for **Use Output Channel**.

4. Enable the SWIFTNet7 Adapter, which starts the SWIFTNet Servers. Wait to ensure that the SWIFTNet Servers are started.

5. Execute the SWIFTNet7CreateOutputChannel business process to send the Create Output Channel request.

6. Monitor the execution of the business process. If it is successful, you have successfully created your new output channel. To start using your new input channel, you must configure the SWIFTNet7 Adapter to use the new output channel name you just created and you must restart the SWIFTNet7 Adapter for the change to occur.

Delete an Existing Output Channel

About this task

Once you delete an output channel, you cannot use it. Therefore, you must be careful when you want to delete an output channel because all the message history for the output channel must have already expired. We recommend that you carefully assess whether an output channels might still be in use before deleting it.

Note: The default/generic output channel cannot be deleted.

To delete a new output channel, you do not need to configure the SWIFTNet7 Adapter to use the output channel. You just need to enable the SWIFTNet7 Adapter to start the SWIFTNet Server so it can forward the delete request to SWIFT.

This is the sample business process to delete an existing output channel :

```
<process name="SWIFTNet7DeleteoutputChannel">
   <sequence name="SWIFTNetDeleteoutputChannel">
      <operation name="set user token">
         <participant name="SetUserToken"/>
         <output message="SetUserTokenMessage">
            <assign to="USER_TOKEN">admin</assign>
            <assign to="." from="*"></assign>
         </output>
         <output message="inmsg">
            <assign to="." from="*"></assign>
         </output>
      </operation>
      <!-- build Delete request -->
      <!--  W A R N I N G   N O T E  -->
      <!-- Once deleted, the output channel cannot be re-created or used anymore -->
      <operation>
         <participant name="SWIFTNet7ClientService"/>
         <output message="deleteoutputChannelRequest">
            <assign to="." from="*"></assign>
            <assign to="serverAdapterName">Put a value here</assign>
            <assign to="authoriserDN">Put a value here</assign>
            <assign to="outputChannelName">Put a value here</assign>
         </output>
         <output message="deleteoutputChannelResponse">
            <assign to="." from="*"/>
         </output>
      </operation>
      <operation name="Receive DeleteoutputChannelSnFResponse">
         <participant name="SWIFTNet7ClientService"/>
         <output message="handleoutputChannelResponse">
            <assign to=""." from="*"
         </output>
         <output message="outputChannelOut">
            <assign to="." from="*">
```

```
            </output>
        </operation>
    </sequence>
</process>
```

To delete an output channel:

Procedure
1. Edit the sample business process, SWIFTNet7DeleteoutputChannel, and supply the following parameters:
   - **authoriserDN**, which must have at least one RBAC role for store-and-forward.
   - **outputChannelName**, which must be the existing output channel name.
2. After you have finished editing, save the business process.
3. Configure the SWIFTNet7 Client service and the SWIFTNet7 Adapter. You do not have to configure the SWIFTNet7 Adapter to use the output channel; instead, you can select **False** for **Use output Channel**.
4. Enable the SWIFTNet7 Adapter, which starts the SWIFTNet Servers.
5. Execute the SWIFTNet7DeleteoutputChannel business process to send the Delete output Channel request.
6. Monitor the execution of the business process. If it is successful, you have successfully deleted the new output channel.

Opening an Output Channel

You can either open an output channel automatically or manually.

Open an Output Channel Automatically
About this task

the SWIFTNet7 Adapter allows you to configure an output channel to be opened when the adapter is enabled.

To open an output channel automatically:

Procedure
1. Configure the SWIFTNet7 Adapter to use the output channel. See "Configure the SWIFTNet7 Adapter for SWIFTNet 7" on page 152 for more information.
2. Enable the SWIFTNet7 Adapter.
3. Access the Current Process page from the **Administration** menu by selecting **Business Processes** > **Current Processes** so you can monitor the business process execution. The SWIFTNet Server triggers a callback to the Sterling B2B Integration, which eventually bootstraps the predefined SWIFTNet7OpenOutputChannel business process to open the output channel. When both the SWIFTNet7OpenOutputChannel business process and the SWIFTNet7OpenOutputChannel business process have completed successfully, the output channel has been successfully opened. If an error occurs, you should check the output channel configuration in the SWIFTNet7 Adapter.

Results

Once the output channel is open, you can send message using this output channel. The resend scheduler works automatically in the background until the SWIFTNet7 Adapter is disabled.

Open an Output Channel Manually
About this task

You can open the output channel manually if you create and invoke a business process similar to the SWIFTNetOpenOutputChannel or SWIFTNet7OpenOutputChannel business process.

Note: Do not modify the SWIFTNetOpenOutputChannel business process because that may affect your ability to automatically open an output channel.

The business process you will need to create must rely on the output channel configuration in the SWIFTNet7 Adapter and you must have already configured this adapter with the appropriate output channel settings.

Note: This manual operation will not function if the SWIFTNet7 Adapter is not configured with the correct output channel settings or the adapter is not enabled.

You may want to manually open an output channel in these circumstances:
* When automatically opening the output channel fails because there is a gap in the channel (that is, the channel was not closed properly) and the SWIFTNet7 Adapter was configured to open in Normal mode, which will cause the open request to fail because it will need to be opened in Force mode. To resolve this issue, you can manually invoke the SWIFTNetManualOpenOutputChannel business process (see example below) with the Force mode equal to TRUE.
* When the SWIFTNet7 Client service encounters an error when sending a message using the output channel, and this sets the status to LOCKED. To resolve this issue, you can manually invoke the SWIFTNetManualOpenOutputChannel business process (see example below) with the Force mode equal to TRUE.
* When you close the output channel manually and need to open the output channel again. In this situation you can manually invoke the SWIFTNetManualOpenOutputChannel business process (see example below) with the Force mode equal to FALSE.

This is the sample SWIFTNetManualOpenOutputChannel business process to open an output channel manually:

```
<process name="SWIFTNetManualOpenOutputChannel">
  <sequence name="SWIFTNetManualOpenoutputChannel">
   <operation name="set user token">
    <participant name="SetUserToken"/>
    <input message="SetUserTokenMessage">
      <assign to="USER_TOKEN">admin</assign>
      <assign to="." from="*"/>
    </input>
   </operation>
   <!-- build Open request -->
   <operation name="Service">
     <participant name="SWIFTNet7ClientService"/>
     <output message="openOutputChannelRequest">
       <assign to="." from="*"/>
       <assign to="forceOpen">TRUE</assign>
```

```
      <assign to="serverAdapterName">SWIFTNetServer7Adapter</assign>
    </output>
    <input message="inmsg">
      <assign to="." from="*"/>
    </input>
   </operation>
  </sequence>
</process>
```

You can supply values for the forceOpen parameter (this is an optional parameter and the default value is FALSE) and the serverAdapterName parameter (this is an optional parameter and the default value is SWIFTNetServerAdapter) in the business process above. The forceOpen parameter indicates whether to open in Force mode (if set to TRUE) and the serverAdapterName parameter indicates where to get the output channel configuration from (from which instance of the SWIFTNet7 Adapter). If you are using the default SWIFTNet7 Adapter, you do not need to specify this parameter.

Note: The adapter must be enabled when you execute this business process.

To open an output channel manually:

Procedure
1. Configure and enable the SWIFTNet7 Adapter.
2. Create a new business process based on the example above and name it SWIFTNetManualOpenOutputChannel.
3. Supply the following parameters in the business process:
   - **forceOpen**, which indicates whether to open in Force mode (if set to TRUE). This is an optional parameter and the default value is FALSE.
   - **serverAdapterName**, which indicates where to get the output channel configuration from (from which instance of the SWIFTNet7 Adapter). Three parameters are required: outputChannelName (the output channel name), serverAdapterName (the SWIFTNet7Adapter name), and queueName (the queue name).
4. After you have finished editing, save the business process.
5. Execute the business process you created, SWIFTNetManualOpenOutputChannel, to send the Open Output Channel request.
6. If the business process successfully completes, the output channel has been opened successfully. If an error occurs, you should check the output channel configuration in the SWIFTNet7 Adapter.

Results

Once the output channel is open, you can send messages using this output channel. The resend scheduler should work automatically in the background until the SWIFTNet7 Adapter is disabled.

Closing the Output Channel

You can either close the output channel automatically or manually.

Close the Output Channel Automatically

About this task

The SWIFTNet7 Adapter allows the output channel to be closed when the adapter is disabled.

To close an output channel automatically:

Procedure

1. Disable the SWIFTNet7 Adapter and check the message shown on the Service Configuration page. The message will depend on the current output channel status. For example, if the message indicates that the status is OPEN and will be disabled automatically after any gaps are resolved, this means the adapter delayed the shutdown process to ensure that any gaps in the output channel are resolved and all current messages receive their acknowledgements. The resend scheduler automatically handles the synchronization during this closing process. Once all gaps are resolved, the resend handler invokes the predefined SWIFTNet7CloseOutputChannel business process to close the channel.

2. You can monitor the Current Process page if you wish. Access the Current Process page from the **Administration** menu by selecting **Business Processes** > **Monitor > Current Processes** so you can monitor the business process execution.

Results

When the business process completes successfully, the output channel is closed successfully, as well. If you see an error, you may need to check the error details and try to close the output channel manually.

Close the Output Channel Manually

About this task

You can also close the output channel manually (that is, without shutting down the SWIFTNet7 Adapter) by creating and invoking a business process similar to the SWIFTNet7CloseOutputChannel business process.

Note: Do not modify the SWIFTNet7CloseOutputChannel business process because this may affect the automatic opening of the output channel.

The business process you create must rely on the output channel configuration in the SWIFTNet7 Adapter and the adapter must be enabled with the appropriate output channel settings.

This is a sample business process to close an output channel manually:

```
<process name="SWIFTNet7CloseOutputChannel">
  <sequence name="SWIFTNetCloseoutputChannel">
   <operation name="set user token">
    <participant name="SetUserToken"/>
    <input message="SetUserTokenMessage">
      <assign to="USER_TOKEN">admin</assign>
      <assign to="." from="*"/>
    </input>
   </operation>
   <!-- build Close request -->
   <operation name="Send CloseOutputChannelSnFRequest">
     <participant name="SWIFTNet7ClientService"/>
```

```
    <output message="closeOutputChannelRequest">
      <assign to="outputChannelName">swiftbic_oc1!x</assign>
      <assign to="serverAdapterName">SWIFTNet7Adapter</assign>
      <assign to="queueName">swiftbic_generic!x</assign>
      <assign to="." from="*"/>
    </output>
    <input message="closeOutputChannelResponse">
      <assign to="." from="*"/>
    </input>
  </operation>
  <operation name="Receive CloseOutputChannelSnFResponse">
    <participant name="SWIFTNet7ClientService"/>
    <output message="handleOutputChannelResponse">
     <assign to="." from="*"/>
    </output>
    <input message="outputChannelOut">
     <assign to="." from="*"/>
    </input>
  </operation>
  <onFault>
   <operation name="Receive CloseOutputChannelSnFResponse">
     <participant name="SWIFTNet7ClientService"/>
     <output message="handleOutputChannelResponse">
       <assign to="." from="*"/>
     </output>
     <input message="outputChannelOut">
       <assign to="." from="*"/>
     </input>
   </operation>
  <onFault>
 </sequence>
</process>
```

To close the output channel manually:

Procedure

1. Configure (with the appropriate output channel settings) and enable the SWIFTNet7 Adapter.

2. Create a new business process based on the example above and name it SWIFTNetManualCloseOutputChannel.

3. Supply the following parameter in the business process: **serverAdapterName**, which indicates where to get the output channel configuration from (from which instance of the SWIFTNet7 Adapter). If you are using the default SWIFTNet7 Adapter, you do not need to specify this parameter. This is an optional parameter and the default value is SWIFTNetServerAdapter. This instance of the adapter must be enabled when you execute the business process.

4. After you have finished editing, save the business process.

5. Execute the business process you created, SWIFTNetManualCloseOutputChannel, to send the Close Output Channel request.

6. If the business process successfully completes, the output channel has been closed successfully. If an error occurs, it may indicate that your request to close the output channel is pending until any gaps are resolved. If this occurs, the resend scheduler attempts to resolve any gaps and the output channel will be closed automatically once all gaps resolved. If you try to send message(s) while the resend scheduler is processing, the messages are stored on local processing. .

Chapter 10. Document Tracking for SWIFTNet Transport

The document tracking support within the SWIFTNet Client service and SWIFTNet Server Adapter or SWIFTNet7 Adapter provides you with a document-centric view of the whole process of SWIFTNet messaging. This gives you the ability to monitor the workflow not only from business process point of view, but also from the actual document point of view. In a single view, you can see how the document is transformed/translated from one form to another within Sterling B2B Integrator, and how the request and response document correlate with each other.

To enable this feature, all the business processes that are related to the SWIFTNet workflow must have the Document Tracking option enabled when you check in or edit the business processes. To do so and enable the framework to track, select the **Document Tracking** check box on the Process Levels page. Set the following options as needed and leave the rest of the business process parameters as the defaults:

- On the Deadline Settings page, set the deadline and notification options, if necessary.
- On the Life Span page, set the life span, if necessary.

Monitor the SWIFTNet Data Flow

About this task

The Data Flow Monitoring feature now enables you to view inbound and outbound SWIFTNet data flows. For both inbound and outbound, you can correlate and view the transformation of the selected document, and you can also see the SWIFTNet Message ID that is related to the document.

Note: Message ID is a unique identifier in SWIFTNet that is required to enable the user to discern duplicate documents—you can tell if a document is a duplicate of another document if the Message IDs of both documents are the same. The Message ID is extremely useful in enabling you to reconcile documents.

To perform an advanced search for SWIFTNet business process data flows:

Procedure

1. From the **Business Process** menu, select **Monitor** > **Advanced Search** > **Data Flows**.
2. In the Business Process Monitor Data Flows page, specify any combination of the following search criteria, as appropriate:

Field	Description
Search	
Endpoint	The remote endpoint of the data flows to search for. Host name or IP address. Optional.
Direction	Direction of the data flows to search for. Optional. Valid values are: • Inbound • Outbound

Field	Description
Protocol	Protocol for the data flows to search for. Optional. Valid values are: • AS2 • AS3 • HTTP • FTP • SFTP • Mailbox Service • MBI • OFTP • IBM Sterling Connect:Direct • WebDAV • SAP • EBICS • SWIFTNet
Status	Current or final status of a data flow. Optional. Select one of the following options: • Normal • Error
Document Name	For data flows associated with a specific document, enter the document name. Optional.
Data Size	Range of size of the data transferred to search for. From/To in bytes, KB, MB, or GB. Optional.
DateRange	From – The beginning date and time for data flows to search for To – The end date and time for data flows to search for **Note:** Select the calendar icon to the right of the date to access calendar information. Optional.
Save search results values by using tag	Enter a string for use in repeating the search in another session. Required.
Results per page	Select how many results to display per page. Required. Valid values are: • 10 • 25 • 50 • 100 • 200 • 250 • 400 • 500 Default is 10.
List Directly	
By Data Flow ID	Type the data flow ID for which you want to search.

3. Click **Go!** The Monitor page opens, listing the business process data flows that match your search criteria.

4. Click the Root Document Name corresponding to the data flow you want to view. There are two different types of data flows:
   - If the Root Document Name is **SWIFTNetRequest**, it corresponds to an outbound data flow in which the Sterling B2B Integrator is acting as the client making a SWIFTNet request to the SWIFTNet Alliance Gateway (SAG).
   - If the Root Document Name is **PsHttpDocument_node_***, it corresponds to an inbound data flow in which the Sterling B2B Integrator is acting as the server processing the request from the client.

Monitor the SWIFTNet Communication Session Records

About this task

Sterling B2B Integrator creates communication session records for any associated authentication, authorization, file transfer, or non-file transfer records, even if a document is not transferred and no data flow record is created. For example, session data can include a user connecting to a mailbox using FTP, receiving messages, and then quitting the FTP session.

To view SWIFTNet communications sessions records:

Procedure

From the **Administration** menu, select **Business Processes** > **Monitor** > **Advanced Search** > **Communication Sessions**.
Complete the fields using the following descriptions:

Field   Description

Endpoint
> The remote endpoint of the communication sessions to search for. Host name or IP address. Optional.

Protocol
> Protocol for the communication sessions to search for. Optional. Valid values are:
> - AS2
> - AS3
> - HTTP
> - FTP
> - SFTP
> - Mailbox Service
> - MBI
> - OFTP
> - IBM Sterling Connect:Direct
> - WebDAV
> - SAP
> - EBICS
> - SWIFTNet

Date Range
> From - The beginning date and time to search for communication sessions
>
> To - The end date and time to search for communication sessions

Note: Select the calendar icon to the right of the date to access calendar information.

Optional.

Principal

Search for communication sessions associated with a Principal participant. Optional.

Secure Mode

Search for communication sessions in a secure mode. Optional. Valid values are:

- SSL
- CCC

Save search results values by using tag

Enter a string for use in repeating the search in another session. Required.

Results per page

Select how many results to display per page. Required. Valid values are:

- 10
- 25
- 50
- 100
- 200
- 250
- 400
- 500

Default is 10.

List Directly

By Communication Session ID

Chapter 11. SWIFT Editor

Sterling B2B Integrator provides you with an online editing interface, the SWIFT Editor, which enables you to correct a SWIFT message (MT and MX) that was returned due to an error in translation (a validation check failed) or transmission (including a process failure such as a negative acknowledgement (NAK)).

Important: Document repair is not supported when using WebSphere Transformation Extender for compliance checking for SWIFT (MT and MX).

Sterling B2B Integrator maintains a link between every SWIFT message that is returned for reprocessing and its historical predecessor, so there is a record of every rejected message all the way back to the initial submission, including references to each person that modified the message and the return code (error status and reason).

The SWIFT Editor contains a sidebar on the left side that provides useful information about the document history and error reports that includes the reasons for an error that occurred.

The SWIFT Editor also enables you to search through the applicable code lists quickly and easily through the Tools section in the SWIFT Editor, including the following:
* SWIFT_Addresses List
* SWIFT_BaseAddresses List
* SWIFT_IBANFormats Code List
* SWIFT_Currencies Code List
* SWIFT_Countries Code List (this SWIFT code list contains countries and regions)
* SWIFT_BICPlusIBAN and BICPlusIBAN Code Lists
* SWIFT_SEPARouting Code List
* NISOLanguage Code List

The editing process requires two different roles (performed by two separate people), per the SWIFT guidelines. Your system administrator will add the appropriate permission (listed in the table below) to the user accounts for both roles to enable each role to perform properly.

Role	Description	Permission Assigned
Editor	This is the person who edits a SWIFT message.	SWIFT Message Edit
Reviewer	This is the person who submits modified messages to be resent.	SWIFT Message Resend

The combination of both roles provides the "four eyes" validation required by SWIFT.

The editing process is as follows:

1. An e-mail alert is sent to the configured address when an outbound translation has errors during FIN enveloping or when a NAK is received.
2. The Editor searches for the returned message through the EDI Correlations search (searching for messages with ReadyForEdit status), and displays the message in the SWIFT Editor.
3. After the Editor repairs the message (or verifies that it does not need repair) and saves it, a separate e-mail alert is sent to the address configured for the Reviewer (the person responsible for auditing the repair and resending the message).
4. The Reviewer receives the e-mail alert and audits the message (looking for messages with ReadyForResend status). The Reviewer accesses the SWIFT Editor in read-only mode. The error report specifies the original errors and the modified and/or added fields as links to their position in the document.
5. After reviewing a repair message, the Reviewer determines how to handle it. The Reviewer can:
   - Reject changes and mark the message for further edit.
   - Abort the repair process entirely.
   - Resend the message using the specified business process.

Editor Tasks

The Editor repairs and saves SWIFT messages that are returned or rejected because of translation or transmission errors.

The following caveats apply:
- The error report displayed in the SWIFT Editor when a Reviewer accesses a repaired document shows all errors and links them to their position in the SWIFT Editor.
- All groups and records can be collapsed and expanded as needed to focus viewing.
- Highlighting an error icon in the SWIFT Editor displays the error message for reference.
- The SWIFT Editor allows users to modify field values and add or delete instances of repeating groups and records.

Edit SWIFT Messages

About this task

To find messages in Ready to Edit status (Editor role task) so you can edit them:

Note: If you click **Cancel** in the SWIFT Document Editor, you will be asked for a confirmation. Click **Yes** on the Confirmation dialog box and a Cancel screen with a status bar is displayed. Once the status of the cancellation is complete, click **Return** to return to the main page

Procedure
1. From the **Administration** menu, select **Business Process** > **Monitor** > **Advanced Search** > **EDI Correlation**.
2. In the Search Options area, specify the following search criteria, as appropriate:

Field	Description	Action
All Level Options		
Location	EDI correlations maintained in a specific location.	Select **Live Tables** – Display live (active) EDI correlations.
Search Level Type	EDI processing level.	Select **Transaction** – For the search query, display results from the transaction level.
Test Mode	Mode of the Sterling B2B Integrator system where documents that contain the EDI correlations were created.	If desired, select one of the following options: • Any (default) • Test • Production • Information • Interchange is a test • Syntax only test • Echo request • Echo response Optional.
Direction	Flow of the documents that contain the EDI correlations.	If desired, select one of the following options: • Any (default) • Inbound • Outbound Optional.
Sender ID	ID for the organization that is sending documents.	Type the identifier of the sender. Optional.
Receiver ID	ID for the receiving organization.	Type the identifier of the receiver. Optional.
Sender ID Qualifier	Qualifier used with the Sender ID to define the organization that is sending documents.	Type the qualifier of the sender. Optional.
Receiver ID Qualifier	Qualifier used with the Receiver ID for the receiving organization.	Type the qualifier of the receiver. Optional.
Start Date	Documents in progress or completed after the specified start date and time.	Using the following formats, type a starting date and time range and select **A.M.** or **P.M.**: • Date – MM/DD/YYYY • Time – HR:MN:SC **Note:** Defaults to a range of the last 24 hours. Optional.

Field	Description	Action
End Date	Documents in progress or completed before the specified end date and time.	Using the following formats, type an end date and time range and select **A.M.** or **P.M.**: • Date – MM/DD/YYYY • Time – HR:MN:SC **Note:** Defaults to a range of the last 24 hours. Optional.
Transaction Level Options		
Transaction Set ID	ID of the transaction set indicated in the document.	If desired, type the ID of the transaction set. Optional.
Compliance Status	Status of compliance checking at the transaction set level.	If desired, select one of the following options: • Any (default) • OK • NOT OK Optional.
Message Repair Status	Status of message repair (for SWIFT documents only).	Select **ReadyForEdit** status.

3. Click **Go!** to display the EDI correlation records that match your search criteria.

4. For SWIFT documents, on the EDI Correlation Transaction Results page, click **info** in the Detail column for the document you want to edit or click the **View** icon to display the message in read-only mode. The Read Only dialog displays the business names and their associated components of data, as opposed to viewing the document to display the raw data of the SWIFT message. Click **Close** to exit the Read Only dialog.

5. Next to Document Repair Status, select the **ReadyForEdit** link to access that message in the SWIFT Editor.

   Note: The Document Repair Status is ReadyForEdit and is a link only if you have the necessary permission to access and edit the document in the SWIFT Editor. If you correct a failed document and save it, the status is changed to ReadyforResend and an e-mail is sent to the address specified in the enveloping.properties property file.

6. Review the errors in the Error Report (left side of the window), select each error link, and repair the error as necessary. This may include changing the content of the field, or adding or deleting fields.

   Note: To add an occurrence of a repeating field or group, select **Add** at the appropriate point in the message structure. To delete an occurrence of a field or group, select **Delete** where appropriate.

7. Click **View Text** if you want to access a plain text display of the message.

8. When you are finished editing the message, click **Validate** to validate the message for SWIFT compliance. If the message contains validation errors, the SWIFT editor retains the message so you can correct the errors and perform further edits.

9. Once validation is successful, click **Save**.

10. Review the changes on the Confirm page, and click **Finish**. This sends an e-mail to the Reviewer, notifying that the message has been repaired.

11. In the EDI Correlation Transaction Detail Results page, you can view data details such as message format (MX or MT). If desired, in the EDI Correlation Transaction Detail Results page, click **info** to the right of Document Correlations to get more details about the message.

12. In the Document Correlation Details page, view details about the message you selected, and to see the correlation between the message and corresponding EDI document or data. The details available include:
    - time stamp
    - scope
    - process ID
    - document name
    - data value

    Note: When you access the returned SWIFT message through the Document Correlation Details page, a tree-view is displayed on the left to allow you to link directly to the previous version(s) of the message. The right pane of the tree view displays the correlation details of the SWIFT message.

13. If you want to view the SWIFT message as text, select the document link at the top right of the page.

Reviewer Tasks

The Reviewer repairs and saves SWIFT messages that are returned or rejected because of translation or transmission errors.

The following caveats apply:
- The error report displayed in the SWIFT Editor when a Reviewer accesses a repaired document shows all errors and links to the position of the error in the SWIFT Editor.
- All groups and records in the message can be collapsed and expanded (as needed) to focus your viewing.
- Highlighting an error icon in the SWIFT Editor Change Report (displayed on the left side of the window) displays the error message for reference.

Search for SWIFT Messages

About this task

To search for messages in Ready to Resend status (Reviewer role task):

Note: If you click **Cancel** in the SWIFT Document Editor, you will be asked for a confirmation. Click **Yes** on the Confirmation dialog box and a Cancel screen with a status bar is displayed. Once the status of the cancellation is complete, click **Return** to return to the main page

Procedure

1. From the **Administration** menu, select **Business Process** > **Monitor** > **Advanced Search** > **EDI Correlation**.

2. In the Search Options area, specify the following search criteria, as appropriate:

Field	Description	Action
All Level Options		
Location	EDI correlations maintained in a specific location.	Select **Live Tables** – Display live (active) EDI correlations.
Search Level Type	EDI processing level.	Select **Transaction** – For the search query, display results from the transaction level.
Test Mode	Mode of the application system where documents that contain the EDI correlations were created.	If desired, select one of the following options: • Any (default) • Test • Production • Information • Interchange is a test • Syntax only test • Echo request • Echo response Optional.
Direction	Flow of the documents that contain the EDI correlations.	If desired, select one of the following options: • Any (default) • Inbound • Outbound Optional.
Sender ID	ID for the organization that is sending documents.	Type the identifier of the sender. Optional.
Receiver ID	ID for the receiving organization.	Type the identifier of the receiver. Optional.
Sender ID Qualifier	Qualifier used with the Sender ID to define the organization that is sending documents.	Type the qualifier of the sender. Optional.
Receiver ID Qualifier	Qualifier used with the Receiver ID for the receiving organization.	Type the qualifier of the receiver. Optional.
Start Date	Documents in progress or completed after the specified start date and time.	Using the following formats, type a starting date and time range and select **A.M.** or **P.M.**: • Date – MM/DD/YYYY • Time – HR:MN:SC **Note:** Defaults to a range of the last 24 hours. Optional.

Field	Description	Action
End Date	Documents in progress or completed before the specified end date and time.	Using the following formats, type an end date and time range and select **A.M.** or **P.M.**: • Date – MM/DD/YYYY • Time – HR:MN:SC **Note:** Defaults to a range of the last 24 hours. Optional.
Transaction Level Options		
Transaction Set ID	ID of the transaction set indicated in the document.	If desired, type the ID of the transaction set. Optional.
Compliance Status	Status of compliance checking at the transaction set level.	If desired, select one of the following options: • Any (default) • OK • NOT OK Optional.
Message Repair Status	Status of message repair (for SWIFT documents only).	Select **ReadyForResend** status.

3. Click **Go!** to display the EDI correlation records that match your search criteria.

4. For SWIFT documents, on the EDI Correlation Transaction Results page, click **info** in the Detail column for the document you want to audit or click the **View** icon to display the message in read-only mode. The Read Only dialog displays the business names and associated data components. Click **Close** to exit the Read Only dialog.

5. Next to Document Repair Status, select the **ReadyForResend** link to display the SWIFT Editor.

   Note: The Document Repair Status is ReadyForResend and is a link only if you have the necessary permission to audit the document in the SWIFT Message Editor.

6. Review the changes made in the Change Report (left side of the window), select each link, and review the change as necessary.

7. When you are finished auditing the entire message, click **Next**.

8. Specify how you want to handle this message by performing one of the following:
   • Mark this document for further edit (sends the message back to the Editor for further editing).
   • Resend this document (sends the document using the business process you select).
   • Abort (terminates the audit process).

9. If you are resending the message, select the appropriate business process from the **Execute Business Process** list. By default it is the SWIFTEnvelope business process.

10. If you want to disable validation prior to resending the message, select **Turn Off Validation**.

> **Note:** Select this option if you want to send a message that does not conform to SWIFT validation rules.

11. Review the Confirm page, and click **Finish**.

Set Default Configuration Options

About this task

To set the default configuration options for the SWIFT Message Entry Workstation and SWIFT Editor (Creator Role Task), perform the following:

Procedure

1. From the **Administration** menu, select **Business Process** > **Message Entry Workstation**.

2. Under **Configure**, next to **Edit Configuration**, click **Go!**.

   Note: The Edit Configuration option is displayed only if you have the Message Entry Configure permission added to your user account.

3. In the View Configuration page, review the configuration settings. If you need to edit the settings, click Go!.

4. In the **Edit Configuration** page, specify the following details, as required, and click **Next**:

   Field   **Description**

   Show - Default send BP
   Enables you to display a default business process in the user interface for sending messages. Selecting the check box enables this feature. Optional.

   Show - Turn off validation
   Enables you to allow the user to have the option to disable validation when sending or resending messages. Selecting the check box enables this feature; clearing the check box indicates that users cannot turn validation off. Optional.

   Default Sender ID
   Type the default Sender ID BIC so the Sender ID field is automatically populated with this value when you create a message. The Sender ID BIC is usually the same for every message. Optional.

   Accepter Lookup Alias
   Enables you to specify that the **Accepter Lookup Alias** option is not displayed in the user interface but is instead defaulted to the message format/message type or just the message type. Select one of the following options:

   - Always provide value (this is the default and indicates that the user will be prompted to provide the Accepter Lookup Alias value)
   - Use format: MessageFormat_MessageType
   - Use format: MessageType

   Note: If you select either of the "Use format" options, the Accepter Lookup Alias defaults to the format you specify and the user does not have the ability to override that format.

   Optional.

5. Click **Next**.

6. On the Confirm Configuration page, click **Finish**.

Chapter 12. SWIFT Message Entry Workstation

Sterling B2B Integrator provides you with the SWIFT Message Entry Workstation functionality that enables you to create, edit, review, copy, delete, and track a SWIFT message.

The SWIFT Message Entry Workstation enables you to create a new SWIFT message (MT or MX), and to enter the message data based on the message structure. Additionally, the SWIFT Message Entry Workstation provides validation functionality that flags syntactical errors that may need to be fixed before sending the message. The SWIFT Message Entry Workstation also allows you to repair errors in the message.

The extent of functionality that is available to each user of the SWIFT Message Entry Workstation is determined by the security permissions that your system administrator assigns to your user account, based on your role. In the Sterling B2B Integrator, each user account has permissions assigned to it, so that the user can perform necessary functions (for SWIFT Message Entry Workstation, this means that you have the required permissions to perform necessary actions for SWIFT messages, depending on your role). The message creation process requires three different roles (performed by three separate people as described in the table below):

Role	Description	Permissions that Must be Assigned to the User Account
Creator	This is the person who creates a message. The Creator can also validate a message or edit default configuration specifications.	• Message Entry Create • Message Entry Edit • Message Entry Delete • Message Entry Copy • Message Entry Configuration
Verifier	This is the person who can submit the created or edited messages to be sent. The Verifier can also validate the messages.	Message Entry Review
Tracker	This is the person who tracks the history of the messages.	Message Entry Track

Note: Each time a user adds, modifies, or deletes a message through the SWIFT Message Entry Workstation, the action is logged to create an audit trail based on the user who performed the event.

Each SWIFT message created through the SWIFT Message Entry Workstation evolves in stages. The possible statuses of a SWIFT message during this evolution are as follows:
- Draft (this is the initial stage and subsequent edits)
- Ready to send
- Sent

• Rejected

SWIFT Message Entry Workstation Display and Usability Features

The SWIFT Message Entry Workstation uses visual cues to indicate whether a component (group, field, and so forth) is mandatory or optional, to clearly identify each:

Component Description
     Displayed With

Mandatory Group
     Dark Blue Background

Mandatory Choice Blocks
     Highlighted in Dark Blue

Mandatory Subfield Labels
     Dark Blue Text

Mandatory Fields and subfield text boxes
     Dark Blue Label with Yellow Background

Optional Fields and subfield text boxes
     Grey Label with White Background

Mandatory Subfield of an Optional Parent Block
     Blue Label with White Background

Also, you can easily access the SWIFT code lists provided with the Sterling B2B Integrator, by using the following:

Note: The country or region codes, currency codes, and code words are available as lists from which you can choose values. These lists are always available in the Tools section of the SWIFT Editor.

• List of the Currencies codes (with description), including Active and Historic currency support for MX messages.
• List of the Countries or Regions codes (with description).
• Lists for Code Words

     Note: Qualifier fields that have a data source name present are treated differently; they have a lookup associated with code word subfields.

The Sterling B2B Integrator provides default values for qualifiers subfields and a default "Start of block" and "End of block" for fields. Additionally, for date fields you are prompted with a **date choice icon** (various date formats defined by MT and MX messages are supported, as well as ISO-8601 Date Time format, which is used in MX messages). Date validations are also supported, and for Time-only formats the Time format is displayed next to the text box for the field.

Installing Available Message Types

The Sterling B2B Integrator automatically includes the newest versions of SWIFT Message Types, but older versions of the MT files are available in jar files that are located in the **<install_dir>/container/Applications** directory. If you want to create a document using an older version of an MT, you need to first install the jar file that contains the appropriate MT.

Note: The 2002 version MT messages are available by installing the SWIFT_2005_MEW.jar file.

Install Older MT Versions

About this task

Complete the following steps to install a jar file containing an older version of an MT:

Note: The jar files are located in the **<install_dir>/container/Applications** directory.

Procedure

1. Locate the jar file for the version of MT you want to create.
2. Navigate to the **<install_dir>/bin** directory and type the following command to invoke installation script:

   ```
   ./InstallService.sh [version].jar
   ```

   Note: Replace [version] in the command above with the name of the jar file containing the appropriate version of the MT. The jar file are named in the format **SWIFT_<year>_MEW.jar**.
3. Press **Enter**.

   You are notified that the installation is complete.

SWIFT Message Creation Process

The message creation process is as follows:

Task	Process
1	The Creator logs in to the Sterling B2B Integrator and the user permissions assigned to the Creator are loaded.
2	Using the SWIFT Message Entry Workstation, the Creator creates a new SWIFT message or searches for a SWIFT message which is in Draft state for further editing. **Note:** When you create a new message, your Sender ID value is populated by default with the value specified on the Configuration page. **Note:** The Sterling B2B Integrator automatically includes the newest versions of SWIFT Message Types, but older versions of the MT files are available in jar files. See "Installing Available Message Types" on page 218 for information on how to install older MT versions if you need to create messages using those versions.
3	After the Creator creates the message, he or she can validate the message to check for SWIFT compliance and mark the message as **MARK AS READY TO SEND**.
4	The Verifier reviews the message and can also perform the following tasks: • Reject the message and mark the message for further edit, including the reasons for rejection (if desired). • Send the message, which invokes the business process (SWIFTMessageEntryOutbound is the default). Once the business process is invoked, the scope of Message Entry Workstation ends.

Note: If you want to disable validation prior to sending the message, select **Turn Off Validation**.

5   If the message is rejected by the Verifier, the Creator can edit the changes using the SWIFT Editor and then mark the message as **Ready for Resend**, and the Verifier can again mark it as **Reject** or **Send**.

6   The Tracker can track the history of the message and review the modifications that were made to the message, including the user who made each change (along with any remarks noted about why a change was made). Also the tracker can view the raw data for the message, regardless of the message status.

Create a SWIFT Message

About this task

To create new messages in the SWIFT Message Entry Workstation (Creator Role Task):

Note: If you click **Cancel** in the SWIFT Document Editor, you will be asked for a confirmation. Click **Yes** on the Confirmation dialog box and a Cancel screen with a status bar is displayed. Once the status of the cancellation is complete, click **Return** to return to the main page

Procedure

1. From the **Administration** menu, select **Business Process** > **Message Entry Workstation**.
2. Under Create, next to New Message, click **Go!**.

   Note: The New Message option is displayed only if you have the Message Entry Create permission added to your user account.
3. In the **Select Message Name** page, enter the following details, as required, and click **Next**:

   Field   Description

   Message name
   The name of the SWIFT message. Required.

   Sender ID
   Identifier for the organization that is sending this SWIFT message. Optional.

   Note: The Sender ID value is required for some SWIFT messages since this value is used to perform validation against your data.

   Note: When you create a new message, your Sender ID value is populated by default with the value specified on the Configuration page.

   Receiver ID
   Identifier for the organization receiving this SWIFT message. Optional.

   Note: The Receiver ID value is required for some SWIFT messages since this value is used to perform validation against your data.
4. In the Select Message Information page, enter the following details (as required), and click **Next**.

Field   Description

Standard
> The standard required to create the SWIFT message.

Standard Version
> The version of the standard required to create the message.

Message Type
> The type of message being created.

5. In the Confirm page, click **Finish**. This launches the **SWIFT Editor** to enable you to type the values in the required parameters to create the new SWIFT message.

    Note: Any data typed for the **Optional Repetitive Group** is displayed under the **Added Groups** section in the Confirmation page. To add an occurrence of a repeating field or group, click **Add** at the appropriate point in the message structure. To delete an occurrence of a field or group, click **Delete** where appropriate.

6. In the SWIFT Editor page, click **View Text** if you want to access a plain text display of the message.

7. In the SWIFT Editor page, the **Validate** function enables you to validate the message for SWIFT compliance so that you can avoid translation errors during enveloping.

8. If any errors occur during validation, review the errors in the Validation Report (left side of the window), select each error link, and repair the error as necessary (repairing may include changing the content of the field, or adding or deleting fields).

    Note: If a SWIFT pass-through map generates the maximum number of translation errors on the input side of the map, it is possible that the output side of the map contains translation errors that are not reported since error processing stops once the maximum number of errors is reached. In this scenario, an error report is not displayed when the output message data is edited through the Message Entry Workstation and EDI Correlation Search, but the message has not been sent due to translation errors and no errors are displayed in the report for the Message Entry Workstation. So, to see validation errors for the output message data, click **Validate** once the message is loaded in the Message Entry Workstation. Then, you can make any necessary fixes and resend the message.

9. You can either save the message for further edits by selecting the SAVE option or submit the message for review by selecting the MARK AS READY TO SEND option. When the action is completed successfully, you are prompted with the following message: "The system update completed successfully." Once a message status is **Ready to Send**, the message can be reviewed by the Verifier.

10. An e-mail alert is sent to the Verifier when the status of the message is changed from DRAFT to MARK AS READY TO SEND. The e-mail address of the Verifier is specified in the **messageentryworkstation.properties.in** property file. Any changes to this e-mail address can be done by editing the customer_overrides.properties file in the following format: **messageentry.SWIFT.READY_TO_SEND.<username> = e-mail** address of the verifier, where **username** is the user who marks the messages as MARK AS READY TO SEND.

Note: If there is no e-mail address specified in the entry **messageentry.SWIFT.READY_TO_SEND.<username>** the system searches for the e-mail address specified in the entry **messageentry.SWIFT.READY_TO_SEND.default**. All ReadyToSend messages will be directed to this e-mail address. By default this is the e-mail address of the system administrator.

11. After you click **Finish** in the Confirm page, you can click **View Text** to access an XML display of the message.

Search for a SWIFT Message

About this task

SWIFT Message Entry Workstation enables you to easily search for messages as follow:

Procedure

1. From the **Administration** menu, select **Business Processes** > **Message Entry Workstation**.

   The parameters described in the Search Parameters table are displayed. The following table describes the search parameters displayed on the Message Entry Workstation page.

   Note: If you click **Cancel** in the SWIFT Document Editor, you will be asked for a confirmation. Click **Yes** on the Confirmation dialog box and a Cancel screen with a status bar is displayed. Once the status of the cancellation is complete, click **Return** to return to the main page
   T

Table 2. Search Parameters

Field	Description	Action
Message Name	The name used for the SWIFT message.	Type the name of the SWIFT message for which you are searching.
Standard	The standard used for the SWIFT message.	Select the SWIFT standard used for creating the message.
Standard Version	The version used for creating the SWIFT message.	Select the standard version used for the SWIFT message.
Message Type	The type of message being created.	Select the type of message that was used to create the message.
Sender ID	Identifier for the organization that is sending the SWIFT message.	Type the identifier of the sender.
Receiver ID	Identifier for organization receiving the SWIFT message.	Type the identifier of the receiver.
Author	The user who created the message.	Type the name of the person who created the message.

Table 2. Search Parameters (continued)

Field	Description	Action
Message Status	Status of the message.	Select the status of the message from the following options: • DRAFT • READY TO SEND • SENT • REJECTED
Start Date	Documents in progress or completed after the specified start date and time.	Using the following formats, type a starting date and time range and select **A.M.** or **P.M.**: • Date – MM/DD/YYYY • Time – HR:MN:SC **Note:** Defaults to a range of the last 24 hours.
End Date	Documents in progress or completed before the specified end date and time.	Using the following formats, type an end date and time range and select **A.M.** or **P.M.**: • Date – MM/DD/YYYY • Time – HR:MN:SC **Note:** Defaults to a range of the last 24 hours.

2. Depending on the search criteria, the Search Result page displays various parameters as follows:

Field   Description

Select Enables the Creator to copy, edit, delete, or get information of a message.

Name Displays the name of the message.

Standard
Displays the SWIFT message standard.

Version
Displays the version of the SWIFT standard.

Type Displays the type of SWIFT message.

Sender ID
Identifier for the organization that is sending the SWIFT message.

Receiver ID
Identifier for the organization receiving the SWIFT message.

Created On
Displays the date and time when the message was created.

Status Displays the status of the message.

Note: If there are no messages found using the search criteria you specify, the message "No Messages match your search criteria" is displayed.

3. Depending on the results displayed for the message for which you searched, you can click **copy**, **edit**, **delete**, or **info** from the Select Field for that SWIFT message.

4. If you click **info**, you can select **View** to display the message, including the business names and associated data components. Click **Close** to exit the Read Only dialog.

5. Click **Return** to return to the main page of the SWIFT Message Entry Workstation.

Verify SWIFT Messages

About this task

The Verifier may not create messages using the SWIFT Message Entry Workstation, but may only verify and validate existing messages.

The following caveats apply to the Verifier role:
- The Verifier uses the SWIFT Editor in read-only mode.
- All groups and records in the message can be collapsed and expanded (as needed) so you can focus your viewing of specific parts of the message.
- Clicking **info** in the Change Report (displayed on the left side of the SWIFT Editor page) displays the details of the changes made for reference.

   Note: When you view a change report in the Message Entry Workstation, if you select the error code link it may not direct you to the field with the error. If this occurs, select the field link instead of the error code link.

To search a message, the verifier performs the following actions:

Note: If you click **Cancel** in the SWIFT Document Editor, you will be asked for a confirmation. Click **Yes** on the Confirmation dialog box and a Cancel screen with a status bar is displayed. Once the status of the cancellation is complete, click **Return** to return to the main page

Procedure

1. From the **Administration** menu, select **Business Process** > **Message Entry Workstation**.

2. The Verifier searches for a SWIFT message based on the search criteria described in Table 2 on page 222.

3. After searching for a SWIFT message (using the procedure for searching a message as described in the section "Search for a SWIFT Message" on page 222), select it and click **Send**. The SWIFT Editor is displayed in read-only mode.

4. When you are finished auditing the entire message, click **Next**.

5. In the Confirm page, specify how you want to handle this message by performing one of the following options:
   - Select the action **Send** to invoke a business process. The default business process is SWIFTMessageEntryOutbound). This business process looks up the envelope that is used to send the message.

Note: If during the message creation process, the Sender ID and Receiver ID were specified for the message, the values for those parameters are automatically populated to these fields and you are unable to modify the values.

- Select the action **Reject** (optionally, you can provide reason for rejection). The **Reject** action updates the message to Rejected status. For each Rejected message, the Creator must make the necessary edits prior to marking the document **Ready for Resend**. (Please note that this document must be updated if the feature of a mail being sent to the Creators' mailbox when a message is rejected is introduced.)

  Note: By default, the business process used for Execute Business Process is **SWIFTMessageEntryOutbound,** which is specified in the **messageentryworkstation.properties.in** property file. If you want to specify a different business process, you must add the name of that business process to the customer_overrides.properties file in the following format: envelope.messageentry=NameOfBusinessProcess (where NameOfBusinessProcess is the name of the business process).

6. Review the Confirm page and click **Finish**.

   Note: The Verifier can only access documents with Ready to Send status. The Verifier is not able to take any action on a message with any other status.

7. An e-mail alert is sent when the status of the message is changed from READY TO SEND to REJECTED. The e-mail alert is sent to the user who last modified the message (not necessarily the user who created it). An e-mail address is specified for each user when the system administrator creates the user accounts.

   Note: If no e-mail address is specified for the user who last modified the message, the system checks for the e-mail address specified in the **messageentryworkstation.properties.in** property file. If neither e-mail address is specified, the system searches for the e-mail address specified in the entry **messageentry.SWIFT.REJECT.default** which is available in the **messageentryworkstation.properties.in** property file. By default, this is the e-mail address of the system administrator. You can change this e-mail address by editing the customer_overrides.properties file in the following format: **messageentry.SWIFT.<username>.mailid** = **e-mail address** of the Creator or the person who last edited the message, where **username** is the user who marks the message as REJECTED.

Track SWIFT Messages

About this task

A Tracker can track the history of a message and view the changes made by each user. The Tracker can also view the raw data for the different message statuses.

In order to track a message, the Tracker must perform the following actions:

Note: If you click **Cancel** in the SWIFT Document Editor, you will be asked for a confirmation. Click **Yes** on the Confirmation dialog box and a Cancel screen with a status bar is displayed. Once the status of the cancellation is complete, click **Return** to return to the main page

Procedure

1. From the **Administration** menu, select **Business Process** > **Message Entry Workstation**.
2. The Tracker searches for a message based on the search criteria as described in the Table 2 on page 222.

   Depending on the search criteria you specify, the Search Result page displays various parameters as described in Table 2 on page 222.
3. In the Message page, click **info** next to the appropriate message to display the following details:

Field   Description

Document
>   Click **info** to view the raw data for the message.

Modified on
>   The date the message was modified.

Modified by
>   The user who modified the message.

Status   The status of the message after modification.

Remarks
>   Remarks made by a user or by the system, if any.

View   Click the **View** icon to display the message in read-only mode. The Read Only dialog displays the business names and associated data components. Click **Close** to exit the Read Only dialog.

Delete a Message

About this task

The SWIFT Message Entry Workstation allows you to delete a message if you have the appropriate permission assigned to your user account. Both the Creator and the Verifier are able to delete messages.

To delete a message, you must perform the following actions:

Note: If you click **Cancel** in the SWIFT Document Editor, you will be asked for a confirmation. Click **Yes** on the Confirmation dialog box and a Cancel screen with a status bar is displayed. Once the status of the cancellation is complete, click **Return** to return to the main page

Procedure

1. From the **Administration** menu, select **Business Process** > **Message Entry Workstation**.
2. Search for a message based on the search criteria described in Table 2 on page 222.

   Depending on the search criteria you specify, the Search Result displays various parameters as described in Table 2 on page 222.

   Note: You must select the **Delete** function from the Select menu in the same row as the message that you want to delete. You are prompted with a delete confirmation window.
3. Click **OK** to delete the message.

4. In the Delete Resources page, review the following message details to ensure you are deleting the right message and click **Next**:

Field   Description

Message Name
　　Displays the name of the SWIFT message that you want to delete.

Standard
　　Displays the SWIFT message standard.

Standard Version
　　Displays the version of the SWIFT standard.

Message Type
　　Displays the type of SWIFT message.

Created On
　　Displays the date and time when the message was created.

Created By
　　Displays the name of the user who created the message.

Current Status
　　Displays the current status of the SWIFT message.

5. In the Confirm page, click **Delete**. When the deletion is complete, the message "the system update completed successfully" is displayed.
6. Click **Return** to return to the main page of the SWIFT Message Entry Workstation.

Copy a Message

About this task

The SWIFT Message Entry Workstation allows you to copy a message if you have the Message Entry Copy permission assigned to your user account.

To copy a message, you must perform the following actions:

Note: If you click **Cancel** in the SWIFT Document Editor, you will be asked for a confirmation. Click **Yes** on the Confirmation dialog box and a Cancel screen with a status bar is displayed. Once the status of the cancellation is complete, click **Return** to return to the main page

Procedure

1. From the **Administration** menu, select **Business Process** > **Message Entry Workstation**.
2. Search for a message based on the search criteria described in Table 2 on page 222.

   Depending on the search criteria you specify, the Search Result displays various parameters as described in Table 2 on page 222.

   Note: You must select the **Copy** function from the Select menu in the same row as the message that you want to copy.

3. Click **Copy** to display the Copy Message page. Enter the following details, as required, and click **Next**:

Field   Description

Message Name

The name of the copy of the SWIFT message. Required.

Sender ID

Identifier for the organization that is sending the SWIFT message. Optional.

Note: The Sender ID value is required for some SWIFT messages since this value is used to perform validation against your data.

Receiver ID

Identifier for the organization receiving the SWIFT message. Optional.

Note: The Receiver ID value is required for some SWIFT messages since this value is used to perform validation against your data.

Note: The status of a copied message is independent from the status of the source message. For example, if the source message was in Draft status when a copy of the message was made, the copied message continues to show the status as Draft even if the status of the source message has since changed to Sent or Rejected.

You are prompted with a Copy Message confirmation window with the following details:

Field   Description

Message Name

Displays the name of the copied SWIFT message.

Copy of Message

Displays the original SWIFT message name from which the copy was created.

Standard

Displays the SWIFT message standard.

Standard Version

Displays the version of the SWIFT standard.

Message Type

Displays the type of SWIFT message.

Sender ID

Identifier for the organization that is sending the SWIFT message.

Note: If a Sender ID was not specified when you created the copy, this parameter contains the indicator Not Provided.

Receiver ID

Identifier for the organization receiving the SWIFT message.

Note: If a Receiver ID was not specified when you created the copy, this parameter contains the indicator Not Provided.

4. Click **Finish** to create a copy of the SWIFT message. Once the copy is successfully created, the message "The system update completed successfully" is displayed.

5. Click **Return** to return to the main Message Entry Workstation page.

Chapter 13. SWIFTNet Routing Rule

The SWIFTNet Routing Rule is used by the SWIFTNet Server adapter or SWIFTNet7 Adapter to manage interactive messages from the SWIFTNet Server, and enables you to configure how you want to process when you receive either an InterAct request payload or FileAct file that was transferred. The rule routes an incoming request message to a user-defined business process based on the following parameters:

- requestorDN
- responderDN
- requestType
- serviceName

You configure these four parameters and create the SWIFTNet routing rule through the SWIFTNet Routing Rule interface. The SWIFTNet Routing Rule page enables you to assign a routing method (any business process, mailbox, or file system directory) to a set of Requestor, Responder, Service, and Request Type. After a message is parsed, the routing rule or rules you have created are applied to the routing of SWIFTNet messages (Funds, MX, or generic XML format), and the server response messages are constructed and sent back to SWIFTNet through the SWIFTNet Server. The routing rule is evaluated after:

- In case of InterAct, it is evaluated when a request is received.
- In case of InterAct, it is evaluated when a request is received.
- In case of FileAct Get, it is evaluated when the transfer is done on requestor side.

You can also use a wildcard (*) for these parameters. Using a wildcard, you can configure a generic routing rule that handles multiple messages from different requestors. Depending on which parameter you are configuring, the wildcard can be used at the beginning, middle, or end of the pattern. For the requestor DN or responder DN, it can be used at the beginning. For the request type it can be used at the end. Additionally you can set the priority of the routing rule to determine which rule supersedes another if there are two or more routing rules with different business processes that match an incoming request or file because you are using the wildcard feature.

Also, the priority function for the SWIFTNet routing rules necessitates that the Sterling B2B Integrator only allow one user to perform update on the priority at any particular time. So, when a user is creating, editing, deleting or importing a SWIFTNet routing rule, a lock will be created so that no other user is allowed to perform those operations.

Note: The wildcard must appear only once in routing criteria. For example: ***,o=swift** is allowed, but ***,o=swift,*** is not allowed because in the latter example the wildcard appears twice in the string.

You would want to use the wildcard function if, for example, the responder has 1,000 requestors that he or she might receive InterAct messages from, and would therefore need to configure 1,000 routing rules even though he or she intends to

invoke the same business process for all the requestors. In this scenario, the user can create one SWIFTNet routing rule and use the wildcard feature to solve his or her business problem.

Note: If you are using CHIPS adapter with SWIFTNet as transport interface, the SWIFTNet Routing Rule will be automatically created for you.

You can also export and import SWIFTNet routing rules.

Note: Each time a user adds, modifies, or deletes a SWIFTNet routing rule, the action is logged to create an audit trail based on the user who performed the event.

Create a SWIFTNet Routing Rule

About this task

To create a SWIFTNet routing rule and select the routing method to process incoming SWIFTNet requests:

Procedure

1. Create a business process to which the inbound messages will be routed by the SWIFTNet Server Adapter or SWIFTNet7 Adapter.
2. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Routing Rule**.
3. To the right of **Create new SWIFTNet routing rule,** click **Go!**.
4. Complete the following parameters and click **Next**:

Parameter
> **Description**

SWIFTNet Routing Rule Name
> Type the name of the SWIFTNet routing rule. Required.
>
> **Note:** Special characters such as ! @ # % ^ ( ) + ? , < > { } [ ] | ; " ' / are not permitted.

Requestor DN
> Type the distinguished name of the requestor or use a wildcard (*) in the string. Required.
>
> **Note:** If you use a wildcard in the string for this value, SWIFTNet specifies that a valid DN always ends with **o=swift**, so the wildcard character is only allowed at the beginning of the string. For example, ***,o=abcd,o=swift** is allowed, but **o=abcd,*,o=swift** is not allowed.
>
> **Note:** This value should be taken from the Requestor's SWIFTNet Client service configuration (or from the BPML parameters passed to the service), in accordance with the arrangement made between requestor and responder.

Responder DN
> Type the distinguished name of the responder or use a wildcard (*) in the string. Required.
>
> **Note:** If you use a wildcard in the string for this value, SWIFTNet specifies that a valid DN always ends with **o=swift**, so the wildcard

character is only allowed at the beginning of the string. For example, ***,o=abcd,o=swift** is allowed, but **o=abcd,*,o=swift** is not allowed.

> **Note:** This value should be taken from the Requestor's SWIFTNet Client service configuration (or from the BPML parameters passed to the service), in accordance with the arrangement made between requestor and responder.

Service Name
> Type the name of the SWIFTNet Client service instance configuration that you created in step 1 or use a wildcard (*) in the string. Required.

> **Note:** This value should be taken from the Requestor's SWIFTNet Client service configuration (or from the BPML parameters passed to the service), in accordance with the arrangement made between requestor and responder.

Request Type
> Type the request type (message type and name) supported by the file transfer responder or use a wildcard (*) in the string. Optional.

> **Note:** If you use a wildcard, it must appear at the end of the string. The SWIFT Request Type generally specifies the category first (for example, pain.001.01.01). Therefore, **pain.*** is allowed but ***.001.01** is not allowed.

> **Note:** This value should be taken from the Requestor's SWIFTNet Client service configuration (or from the BPML parameters passed to the service), in accordance with the arrangement made between requestor and responder.

5. Complete the following parameter and click **Next**:

Assign Priority for this Routing Rule (relative to other match rules)
> Enables you to assign priority to the routing rule, if the routing parameters using the wildcard are matched to an existing routing rule. This resolves the issue that occurs when you are using wildcards and an incoming request corresponds to two existing routing rules. In this case, the rule with higher priority will be invoked. Optional.

> **Note:** If no other rules match the routing parameters used for this rule, the priority will be assigned automatically.

6. Complete the following parameters and click **Next**:

Invoke Business Process
> This is the default selection, indicating that messages will be routed based on a selected business process.

Business Process
> Select the business process that will be invoked to manage the request and handle the response to a SWIFTNet message (this is the business process that you created in step 2 that will be called when the requestor DN, responder DN, service name, and requestor type match). Required if you select **Invoke Business Process**.

Invoke Mode
> Whether the mode of invocation is synchronous (default) or asynchronous. Required if you select **Invoke Business Process**.

Store into Mailbox

Indicates that messages will be routed to a selected mailbox.

Note:

If you utilize this option, the routing rule calls the Add Mailbox service and if the mailbox does not exist, it will be automatically created.

Filter by Name

Use to filter mailboxes by name to narrow your selection criteria. Only accessible if you select **Store into Mailbox**.

Parent Mailbox

Select the Parent Mailbox into which messages will be routed. Required if you select Store into Mailbox.

Note: In the Parent Mailbox list, you are not able to pick any mailbox that contains a SWIFT DN (for example, /o=ptscfrnn,o=swift, etc.). Therefore, those DNs are not displayed in the list.

Store into File System

Indicates that messages will be routed to the specified directory in your file system

Note: If you utilize this option, the routing rule calls the File System Adapter and if the directory you specify does not exist, it will be automatically created.

Root Directory

The directory into which messages will be routed. Required if you select **Store into File System**.

Please specify an XML file that serve as a default response payload whenever appropriate.

Type the response template that serves as the default response payload.

Note: This response template is only applicable for Interact Real-time (which requires that a response be sent back immediately) when the action you selected is asynchronous (one-way). The response template is also required, if Action is set as mailbox or file system (in which case the response payload will be taken from response template file).

Note: The Sterling B2B Integrator attempts to locate any other routing rule defined in the system that matches with your routing rule. You can specify the priority of your rule in relation to those matches rules.

7. Click **Finish** to save the routing rule. The rule is now in effect for all incoming SWIFTNet messages (Funds, MX, or generic XML format).

Searching for a SWIFTNet Routing Rule

To edit or delete a SWIFTNet routing rule you must first specify the appropriate rule. You can locate a specific routing rule in two ways:

- Search for the routing rule by name.
- Select the routing rule from an alphabetical list.

Searching for the routing rule by name is more precise and provides fewer results. Searching from an alphabetical list will result in a list of all routing rules or all rules beginning with a specified letter or digit.

Once you search for the routing rule, you can easily edit or delete it from the SWIFTNet Routing Rule interface. You can also update an existing rule by changing its priority.

Note: While creating, editing, or deleting a routing rule, a lock is held on the routing rule. Therefore, when you are creating, editing, or deleting a routing rule, no other routing rule can be created, modified, or deleted.

Search for a Routing Rule by Name
About this task

To search for a routing rule by name:

Procedure
1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Routing Rule**.
2. In the Search section, type the name of the routing rule. Case does not matter and you can type part of a name.

   The Sterling B2B Integrator returns a list of matches unless no routing rules meet the criteria you specified.

Search for a Routing Rule from a List
About this task

To select a routing rule from a list:

Procedure
1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Routing Rule**.
2. In the List section, select one of the following:
   - Alphabetically – Select **All** and click **Go!**
   - Alphabetically – Select a specific letter or digit (0 - 9) and click **Go!**

     The Sterling B2B Integrator returns a list of matches unless no routing rules meet your criteria.

Deleting a Routing Rule
About this task

If you delete a routing rule that has matching rules applied to it, the priority of other lower-priority routing rules is changed.

Exporting and Importing a SWIFTNet Routing Rule

The Sterling B2B Integrator Import/Export feature enables you to save time and increase the accuracy of duplicating supported resources on different environments that are set up for unique purposes. To import and export resources from one application environment to another application environment, both environments must be the same version.

When you import a matching routing rule that uses wildcards and choose **Update = Yes** during the import, the Import Routing Rule function overrides routing rules whose name is an exact match with the imported one.

The system does not allow two routing rules (two rules with different names) to have the same combination keys (Requestor DN, Responder DN, Service Name, and Request Type). Therefore, during the import process, if the imported rule has the same combination keys as another routing rule in the system, the import (update) is rejected.

When you import a rule that does not use combination keys that duplicate a combination that already exists in the system (insert cases), the rule is assigned to the lowest sequential priority.

Chapter 14. SWIFTNet Service Profile

The HeaderInfo block is optional, except for those services that mandate it. If the HeaderInfo block is not used, it must not be present, and if it is used, it must be validated by the schema.

The SWIFTNet Service Profile enables you to easily port Service Profiles from one application instance to another. This function allows you to associate SWIFTNet Request Type with a Schema for Header Validation. You need to create the SWIFTNet Service Profile and associate the request type with the selected schema. This allows the Sterling B2B Integrator to validate the HeaderInfo when it is present in the request.

Note: The schema must be saved in Sterling B2B Integrator.

The Request Type parameter can accept a wildcard (*) to be used only at the end of the string. To determine which Service Profile to be used for a particular Request Type, the Sterling B2B Integrator uses a best-match policy. For example, if there are two Service Profile defined, for pain.* and pain.001.*, and the actual request type is pain.002.001, then the first one will be selected.

Two SWIFTNet Service Profiles are preloaded into Sterling B2B Integrator. The **pacs.*** and **pains.*** service profiles are associated with the Transaction Count schema and set to **Required for validation**. The Transaction Count and Payment Summary schemas are also preloaded into the Sterling B2B Integrator.

You can also import and export SWIFTNet Service Profiles from one application instance to another.

Create a SWIFTNet Service Profile

About this task

To create a SWIFTNet service profile:

Procedure

1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Service Profile**.
2. To the right of **Create new SWIFTNet service profile,** click **Go!**.
3. Complete the following parameters and click **Next**:

   Parameter
   > **Description**

   Request Type
   > Type the request type. A wildcard (*) is only allowed at the end of the string, for example for example, **pacs.*** or **pacs.001.***. Required.

   Schema Name
   > Select the schema used to validate the header information for this request type. Required.

Validation Type

Select whether validation is mandatory or should only be used if header information is specified. Optional. Valid values are:

- Validates only if Header Information is specified (default)
- Validation of Header Information is required

4. Click **Finish** to save the service profile.

Searching for a SWIFTNet Request Type

To edit or delete a SWIFTNet request type, you must first locate the appropriate request type. You can locate a specific request type in two ways:

- Search for the request type by name.
- Select the request type from an alphabetical list.

Searching for the request type by name is more precise and provides fewer results. Searching from an alphabetical list will result in a list of all request type or all types beginning with a specified letter or digit.

Once you search for the request type , you can easily edit or delete it from the SWIFTNet Service Profile interface.

Search for a Request Type by Name
About this task

To search for a request type by name:

Procedure

1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Service Profile**.
2. In the Search section, type the name of the request type. Case does not matter and you can type part of a name.

   The Sterling B2B Integrator returns a list of matches unless no request type meet the criteria you specified.
3. When the list of matches is returned, click **edit** next to the request type you want to modify, or click **delete** next to the request type you want to remove.

Search for a Request Type from a List
About this task

To select a request type from a list:

Procedure

1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Service Profile**.
2. In the List section, select one of the following:
   - Alphabetically – Select **All** and click **Go!**
   - Alphabetically – Select a specific letter or digit (0 - 9) and click **Go!**

     The Sterling B2B Integrator returns a list of matches unless no request type meet your criteria.
3. When the list of matches is returned, click **edit** next to the request type you want to modify, or click **delete** next to the request type you want to remove.

Exporting and Importing a SWIFTNet Service Profile

The Sterling B2B Integrator Import/Export feature enables you to save time and increase the accuracy of duplicating supported resources on different environments that are set up for unique purposes. To import and export resources from one application environment to another application environment, both environments must be the same version.

Chapter 15. SWIFTNet Copy Service Profile

The SWIFTNet Copy Service Profile page enables you, as a third party, to specify the relationship between a Service Name and Copy Mode (for T- or Y-Copy). With T-Copy, the third party copy is for information only. For Y-Copy, third party authorization is needed.

With the release of SWIFTNet 6.1, the Copy Service was available for FileAct. For SWIFTNet 7.0, the Copy Service is also available for InterAct SnF -- both T-Copy and Y-Copy model are available. Also, in SWIFTNet 7.0, both Copy Types Header and Full are supported by SWIFT for FileAct and InterAct (Full only). Additionally, in SWIFTNet 7.0 for T-Copy, the request can be automatically copied to up to ten Third Party recipients for information only. In Y-Copy, the request is copied to a Third Party, and requires the Third Party to approve it before it can be delivered to the intended recipient (responder DN).

Note: You cannot define both T-Copy and Y-Copy for the same service. Additionally, if the UseASP flag is set to true and ASPs are imported into your instance, the SWIFTNet Copy Service Profiles are not created for the services. However, if the useASP flag is disabled, SWIFTNet Copy Service Profiles must be created for the services irrespective of whether ASPs for the services are imported into your instance or not.

Create a SWIFTNet Copy Service Profile

About this task

To create a SWIFTNet Copy Service Profile:

Procedure

1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Copy Service Profile**.
2. To the right of **Create new SWIFTNet copy service profile,** click **Go!**.
3. Complete the following parameters and click **Next**:

   Parameter
   > **Description**

   Service Name
   > Type the copy service name. Required.

   Copy Mode
   > Select whether the copy mode is T-Copy (third party copy is for information only - this is the default) or Y-Copy (third party authorization is needed). Optional.

4. Click **Finish** to save the SWIFTNet Copy Service Profile.

Searching for a SWIFTNet Copy Service Profile

To edit or delete a SWIFTNet Copy Service Profile, you must first locate the appropriate SWIFTNet Copy Service Profile. You can locate a specific SWIFTNet Copy Service Profile in two ways:

- Search for the SWIFTNet Copy Service Profile by service name.

• Select the SWIFTNet Copy Service Profile from an alphabetical list.

Searching for the SWIFTNet Copy Service Profile by service name is more precise and provides fewer results. Searching from an alphabetical list will result in a list of all SWIFTNet Copy Service Profiles or all types beginning with a specified letter or digit.

Once you search for the SWIFTNet Copy Service Profile, you can easily edit or delete it from the SWIFTNet Copy Service Profile interface.

Search for a SWIFTNet Copy Service Profile by Name
About this task

To search for a SWIFTNet Copy Service Profile by name:

Procedure
1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Copy Service Profile**.
2. In the Search section, type the service name of the SWIFTNet Copy Service Profile. Case does not matter and you can type part of a name.

   The Sterling B2B Integrator returns a list of matches unless no SWIFTNet Copy Service Profile meet the criteria you specified.
3. When the list of matches is returned, click **edit** next to the SWIFTNet Copy Service Profile you want to modify, or click **delete** next to the SWIFTNet Copy Service Profile you want to remove.

Search for a SWIFTNet Copy Service Profile from a List
About this task

To select a SWIFTNet Copy Service Profile from a list:

Procedure
1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Copy Service Profile**.
2. In the List section, select one of the following:
   • Alphabetically – Select **All** and click **Go!**
   • Alphabetically – Select a specific letter or digit (0 - 9) and click **Go!**

     The Sterling B2B Integrator returns a list of matches unless no SWIFTNet Copy Service Profiles meet your criteria.
3. When the list of matches is returned, click **edit** next to the SWIFTNet Copy Service Profile you want to modify, or click **delete** next to the SWIFTNet Copy Service Profile you want to remove.

Chapter 16. SWIFTNet Client Profile

The SWIFTNet Client Profile enables you to create a profile and store it in the database for repeated usage. This is especially useful for ad-hoc testing with a partner. This feature is available to be used with the SWIFTNet7 Client service only. If the SWIFTNet7ClientService is called from a business process and profileName is specified, the service retrieves the client profile and uses the information stored there to send a message to the SWIFTNet Server.

This wizard uses the Application Service Profile (ASP) feature introduced in SWIFTNet 7. SWIFT provides the ASP that is imported into the system when you create a SWIFTNet Client Profile, the wizard looks up in the ASP (which was already imported into the system) to determine the possible values for each of the parameter in the user interface (only if the property **useASP** from <Install_DIR>/properties/swiftnet7.properties is set to true). However, if you do not want to make use of the ASP feature, you can still create a SWIFTNet Client Profile. In this case the wizard still notifies you which parameters are required/allowed, but only to a certain extent because the unavailability of the ASP itself.

If you are not using ASP, you should set the useASP parameter in the swiftnet7.properties file to FALSE so that you can use the Client Profile.

When ASP is used, the wizard is able to populate the fields with the correct options. For example, if Non-Repudiation is optional, you are able to select TRUE/FALSE. If the service is a Store-and-Forward Service, the wizard only shows the fields related to Store-and-Forward. Another example is if the service is a Copy Service, and ThirdPartySelectable is TRUE, the wizard populates a list of third party DNs that the user can select. If the Copy Mode is Y-Copy, the wizard can guide you to only select one third party DN, but if the Copy Mode is T-Copy, the wizard allows you to select up to ten third party DNs.

All these options are not available if either ASP is not used or useASP=false. Using the example above, you would still able to specify the list of third party DNs, but the DN value might not be valid.

If a service has thirty third party DNs defined and if ThirdPartySelectable flag is set to TRUE, the Requestor can select third parties from the list to which the request must be copied. For T-Copy Service, the requestor is allowed to choose up to ten third party DNs. For Y-Copy Service, the requestor is only allowed to choose one. However, if the ThirdPartySelectable flag is set to FALSE, the requestor is not allowed to choose, so the request is copied to all the third party DN(s) defined in the list.

When sending an InterAct message with the Possible Duplicate flag is set to True, you can specify the Message ID (if it is known) or leave it blank. When sending a FileAct message, you may specify Physical Filename, Logical Filename, File Description, File Information, Transfer Description and Transfer Information. A System message is an InterAct SnF message with a special payload that communicates with SWIFT, not with the final responder. When sending a System message, you may specify system-message related parameters to automatically create the system-message payload.

For SWIFTNet 7.0, the SWIFTNet Client Profile also allows you to take advantage of the SWIFTNet 7.0 features related to Message or File Distribution.

You can also import and export SWIFTNet Client Profiles from one application instance to another.

Create a SWIFTNet Client Profile

About this task

To create a SWIFTNet Client Profile:

Procedure

1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Client Profile**.
2. To the right of **Create new SWIFTNet Client Profile,** click **Go!**.
3. Complete the following parameters and click **Next**.

   Note: If you select System for Message Type, additional parameter(s) may be displayed based on which system message you choose.

   Profile Name
   > Type the name of the client profile. Required.

   Message Type
   > Select the type of message. Valid values are InterAct (default), FileAct, and System. Required.

   Requestor DN
   > This is the distinguished name of the requestor. Required.
   >
   > **Note:** This DN must be registered with the SAG instance using SWIFTNet Alliance Webstation. This value should be taken from the Requestor's SWIFTNet7 Client service configuration (or from the BPML parameters passed to the service), in accordance with the arrangement made between requestor and responder.

   Authoriser DN
   > This is the distinguished name of the authorizer, specified in the context of the input channel. Optional.
   >
   > **Note:** This DN must be registered with the SAG instance using SWIFTNet Alliance Webstation. This value should be taken from the Requestor's SWIFTNet7 Client service configuration (or from the BPML parameters passed to the service), in accordance with the arrangement made between requestor and responder.

   Service Name
   > Name of the service to which both SWIFT correspondents have subscribed. Required.
   >
   > **Note:** This must be a SWIFTNet service to which you are subscribed. This value should be taken from the Requestor's SWIFTNet7 Client service configuration (or from the BPML parameters passed to the service), in accordance with the arrangement made between requestor and responder.

Request Type

Type the request type. A wildcard (*) is only allowed at the end of the string, for example for example, pacs.* or pacs.001.*. Optional.

Note: This value should be taken from the Requestor's SWIFTNet7 Client service configuration (or from the BPML parameters passed to the service), in accordance with the arrangement made between requestor and responder.

Request Reference

User reference of the request. Optional.

Message Priority

The priority of the message. Valid values are Normal Priority (default) and Urgent Priority. Optional.

Number of Retries

The number of retries to connect to SAG in the case of non-fatal errors. Optional. Default is 3.

Possible Duplicate

Whether the trailer is a possible duplicate. Valid values are False (default) and True. Optional.

Message ID

A unique identifier in SWIFTNet that is required to enable you to discern duplicate documents. Optional.

Physical Filename

Physical name of the file to send. Required if put or get is selected for the SWIFTNet Operation. Required. Only displayed if you select FileAct as the **Message Type**.

Logical Filename

Logical name of the file to send. This name is communicated to the application SWIFTNet Server. By default, this name is the Physical Filename without the path. Required. Only displayed if you select FileAct as the **Message Type**.

File Description

User description about the file transfer. Optional. Only displayed if you select FileAct as the **Message Type**.

File Information

User information about the file transfer. Optional. Only displayed if you select FileAct as the **Message Type**.

Transfer Description

User description of the transfer. Optional. Only displayed if you select FileAct as the **Message Type**.

Transfer Information

User information about the transfer. Optional. Only displayed if you select FileAct as the **Message Type**.

Authorization Status

The status of the transfer. Valid values are Authorized (default) and Refused. Required. Only displayed if you select System as the **Message Type**.

Third Party To Receiver Info

When the **Authorization Status** is Authorized, this parameter contains

the information that the third party wants to convey to the original receiver of the request. Optional. Only displayed if you select System as the **Message Type**.

Third Party To Sender Info

When the **Authorization Status** is Authorized, this parameter contains the information that the third party wants to convey to the original sender of the request. Optional. Only displayed if you select System as the **Message Type**.

Third Party Refusal Info

When the **Authorization Status** is Refused, this parameter contains the refusal reason that the third party wants to convey to the original sender of the request. Optional. Only displayed if you select System as the **Message Type**.

Copy SnF Reference

The copy reference value, which allows you to specifically choose which message the third party should authorize or refuse. You (as the third party) can find this in the incoming request (CopySnFRef tag) or in the third party mailbox (because for Y-Copy that requires third party approval, it is inserted into the mailbox under message name ThirdParty_[CopySnFRef]). Required. Only displayed if you select System as the **Message Type**.

4. Click **Next** and complete the following parameters:

SWIFTNet7 Adapter

Name of the adapter instance or adapter group to be used when sending this request. Optional.

SnF Mode

The store-and-forward mode. Specify True for Store-and-Forward or False for Real-Time.

Operation Type

The operation type. For InterAct it is either Asynchronous (only for Real-Time) or Synchronous (for Store-and-Forward). For FileAct it is either Get (only for Real-Time) or Put (for Store-and-Forward).

RMA Filter

The RMA filter to be used.

Non Repudiation

Indicates whether non-repudiation is required. Possible values are True (when enabled this means that trading partners cannot deny that they sent a request) or False (default—when enabled this indicates that non-repudiation is not required). Optional.

End to End Signing

Specification for end-to-end signing. Valid values are No Signing, Crypto (encrypted). or Signature List. Optional.

Return Signature List

Whether to return a signature list. Valid values are False (default) and True. Optional. If you want a signature list returned, the SWIFTNet Server receives the requestor's own signature in the response message. This returned signature will be extracted and saved as a separate message. This message is stored in the database and is made available for Correlation search. This parameter is only populated if End to End Signing is set to Signature List.

Use RND

Whether to use RND (digest reference values that terminate on "and RND"). Valid values are False (default) and True. Optional. This parameter is only populated if End to End Signing is set to Signature List.

Request For Del. Notification

Whether delivery notification is requested after getting a request from a remote client. Valid values are False (default) and True. Optional.

Note: Only is displayed if you selected InterAct(SnF) or FileAct(SnF) as the Message Type. It is not displayed if you selected FileAct (Real Time) or InterAct (Real Time).

Del. Notification Queue

The queue used for delivery notification. Required. Only is displayed if you selected InterAct(SnF) or FileAct (Real Time) or FileAct(SnF) as the Message Type. It is not displayed if you selected InterAct (Real Time).

Del. Notification As System Message

Whether requested delivery notification is delivered as a system message. Valid values are False and True (default). Optional. Only is displayed if you selected InterAct(SnF) or FileAct (Real Time) or FileAct(SnF) as the Message Type. It is not displayed if you selected InterAct (Real Time).

Overdue Delay

A time interval (of up to 14 days) after which you want to be warned that a delivery notification is overdue. Optional. Only is displayed if you selected InterAct(SnF) or FileAct (Real Time) or FileAct(SnF) as the Message Type. It is not displayed if you selected InterAct (Real Time).

Header Info

Allows you to specify up to 1,000 characters of header information. Optional. Only is displayed if you selected InterAct(SnF) or FileAct (Real Time) or FileAct(SnF) as the Message Type. It is not displayed if you selected InterAct (Real Time).

Copy Mode

Allows you to indicate the copy mode.

Copy Indicator

Whether a copy service is being used. Valid values are Not in Use (service is a non-copy service), True (T-Copy), and False (Y-Copy). Optional. Only displayed if you select InterAct or FileAct as the Message Type and SnF Mode is set to true.

Auth. Notification Indicator

Whether you want to be notified when an authorization is sent by the third party. Valid values are False (default) and True. Optional. Only displayed if you selected InterAct or FileAct as the Message Type. This is for Y-Copy context, to request notification as soon as the Third Party authorizes the message.

If a request is rejected by third party, notification is sent to the requestor regardless of how this parameter is set.

Third Party Destination(s)

In SWIFTNet 7.0, T-Copy allows you to copy the Request to maximum of ten third party destinations. Before sending the request, the

SWIFTNet7 Client service validates the list of destinations. It is not possible to use the Message/File Distribution feature together with Copy Destination list.

Responder DN
The distinguished name of the responder. Required.

Responder List
Allows you to specify a list of up to 1,000 recipients that will receive the request. The application allows you to incorporate the signature list when sending requests to the distribution list.

Responder List is Public
Whether the list is private or public (whether or not all recipients will know who the other recipients are, such as a CC or BCC list used in electronic mail.) Valid values are TRUE (list is public - this is the default) and FALSE (list is private). Only displayed if you selected InterAct(SnF) or FileAct(SnF) as the Message Type. It is not displayed if you selected FileAct (Real Time) or InterAct (Real Time).

Use Input Channel
When sending an InterAct SnF message, you may request to use Input Channel. Valid values are False (do not use an input channel - this is the default) and True. Optional. Only displayed if you selected InterAct or System as the Message Type. Only displayed if you selected InterAct(SnF) or FileAct(SnF) as the Message Type. It is not displayed if you selected FileAct (Real Time) or InterAct (Real Time).

5. Click **Finish** to save the SWIFTNet Client Profile.
6. If you want to test your SWIFTNet Client Profile, search for the profile and click the execute icon next to the profile name in the list.

Searching for a SWIFTNet Client Profile

To edit or delete a SWIFTNet Client Profile, you must first locate the appropriate SWIFTNet Client Profile. You can locate a specific SWIFTNet Client Profile in two ways:

- Search for the SWIFTNet Client Profile by service name.
- Select the SWIFTNet Client Profile from an alphabetical list.

Searching for the SWIFTNet Client Profile by service name is more precise and provides fewer results. Searching from an alphabetical list will result in a list of all SWIFTNet Client Profiles or all types beginning with a specified letter or digit.

Once you search for the SWIFTNet Client Profile, you can easily edit or delete it from the SWIFTNet Client Profile interface.

Search for a SWIFTNet Client Profile by Name
About this task

To search for a SWIFTNet Client Profile by name:

Procedure

1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Client Profile**.
2. In the Search section, type the service name of the SWIFTNet Client Profile. Case does not matter and you can type part of a name.

The Sterling B2B Integrator returns a list of matches unless no SWIFTNet Client Profile meet the criteria you specified.

3. When the list of matches is returned, click **edit** next to the SWIFTNet Client Profile you want to modify, or click **delete** next to the SWIFTNet Client Profile you want to remove. The SWIFT correlation contains a separate Delivery Notification column with an info link to show a more comprehensive status of the message delivery to each Recipient (Responder DN, Status, and Workflow ID).

Search for a SWIFTNet Client Profile from a List

About this task

To select a SWIFTNet Client Profile from a list:

Procedure

1. From the Sterling B2B Integrator **Deployment** menu, select **Adapter Utilities** > **SWIFTNet Client Profile**.
2. In the List section, select one of the following:
   - Alphabetically – Select **All** and click **Go!**
   - Alphabetically – Select a specific letter or digit (0 - 9) and click **Go!**

     The Sterling B2B Integrator returns a list of matches unless no SWIFTNet Client Profiles meet your criteria.
3. When the list of matches is returned, click **edit** next to the SWIFTNet Client Profile you want to modify, or click **delete** next to the SWIFTNet Client Profile you want to remove. The SWIFT correlation contains a separate Delivery Notification column with an info link to show a more comprehensive status of the message delivery to each Recipient (Responder DN, Status, and Workflow ID).

Export and Import a SWIFTNet Client Profile

About this task

The Import/Export feature enables you to save time and increase the accuracy of duplicating supported resources on different environments that are set up for unique purposes. To import and export resources from one Sterling B2B Integration environment to another Sterling B2B Integration environment, both environments must be the same version.

Chapter 17. Configuring the WebSphere MQ Adapter/Suite to Communicate with SWIFT

The WebSphere MQ adapter and the WebSphere MQ Suite enable you to configure the Sterling B2B Integrator to send and receive SWIFT messages to/from SWIFTNet through the WebSphere MQ Interface for SWIFTAlliance Access (MQSA). Also enables you to send and receive messages to/from SWIFTNet though SWIFTNet Alliance Access (SAA) and the SWIFTNet Remote API Host Adapter (RAHA).

This table describes the tasks necessary to configure the Sterling B2B Integrator to communicate with SWIFTNet through either the WebSphere MQ adapter or the WebSphere MQ Suite:

Number	Task	For More Information
1	Configure the Sterling B2B Integrator to retrieve messages through SAA and MQSA.	"Configure the Sterling B2B Integrator to Retrieve Messages"
2	Configure the Sterling B2B Integrator to send messages through SAA and MQSA.	"Configure the Sterling B2B Integrator to Send Messages" on page 250
3	Disable the UMID and Block S options for all queues within the MQ interface.	"Configure the UMID and Block S Options" on page 251

Configure the Sterling B2B Integrator to Retrieve Messages

About this task

Complete these steps to configure the Sterling B2B Integrator to retrieve messages from SWIFTNet through SAA and MQSA:

Note: See *WebSphere MQ Adapter* and *WebSphere MQ Suite* documentation for more details on the general MQ options.

To Configure an MQ Adapter service instance:

Procedure

1. Go to **Deployment** > **Services** > **Configuration**.
2. Next to New Service, click **Go!**.
3. Select **WebSphere MQ Adapter** and click **Next**.
4. Type a name for the service and a description, and click **Next**.
5. On the "WebSphere MQ Parameters page, type the following information and click **Next**:
   - Set **Host Name** to the name or IP address of the machine hosting the WebSphere MQ that receives messages from SWIFT.
   - Set **Listening Port** to the port number for the MQ installation (if it is something other than the default).
   - Set **Queue Manager** to the name of the queue manager that contains the SWIFTNet message queues.

- Set **Queue Name** to the name of the queue set up to receive messages from SWIFT.
- Set **Server Connection Channel** to the name of the connection channel associated with the queue manager.
- Set **User ID** and **Password** to the login information for the queue that is configured to receive messages, if required.
- Select the **Receiving messages from WebSphere MQ (Sync)** or **Receiving messages from WebSphere MQ (Async)** option, depending which mode you wish to use.
- Set the other parameters based on the specific MQ configuration desired.

6. Click **Finish** to save the configuration.
7. Create a business process to retrieve the messages. The following example business process retrieves a message from MQ in synchronous mode, assuming the service configuration **FromSAA** was created in step 1. Optional parameters have been set to make the business process wait for up to 10 seconds for a message to appear before timing out:

```
<process name="FromSAABP">
   <operation name="WebSphere MQ Adapter">
      <participant name="FromSAA"/>
      <output message="WebsphereMQInputMessage">
         <assign to="." from="*"></assign>
         <assign to="rcv_MQGMO_wait">Yes</assign>
         <assign to="rcv_MQGMO_waitInterval">10000</assign>
      </output>
      <input message="inmsg">
         <assign to="." from="*"></assign>
      </input>
   </operation>
</process>
```

Configure the Sterling B2B Integrator to Send Messages

About this task

Complete the following steps to configure the Sterling B2B Integrator to send messages to SWIFTNet through SAA and MQSA:

To Configure an MQ Adapter service instance:

Procedure

1. Go to **Deployment** > **Services** > **Configuration**.
2. Next to New Service, click **Go!**.
3. Select **WebSphere MQ Adapter** and click **Next**.
4. Type a name for the service and a description, and click **Next**.
5. On the "WebSphere MQ Parameters page, type the following information and click **Next**:
   - Set **Host Name** to the name or IP address of the machine hosting the WebSphere MQ that receives messages from SWIFT.
   - Set **Listening Port** to the port number for the MQ installation (if it is something other than the default).
   - Set **Queue Manager** to the name of the queue manager that contains the SWIFTNet message queues.
   - Set **Queue Name** to the name of the queue set up to receive messages from SWIFT.

- Set **Server Connection Channel** to the name of the connection channel associated with the queue manager.
- Set **User ID** and **Password** to the login information for the queue that is configured to receive messages, if required.
- Select the **Sending messages to WebSphere MQ** option.
- Set the other parameters based on the specific MQ configuration desired.

6. Click **Finish** to save the configuration.
7. Create a business process to send the message. The following example business process sends a message from MQ in synchronous mode, assuming the service configuration **ToSAA** was created in step 1. .

```
<process name="ToSAABP">
    <operation name="WebSphere MQ Adapter">
      <participant name="ToSAA"/>
      <output message="WebsphereMQInputMessage">
        <assign to="." from="*"></assign>
        <assign to="snd_MQMD_msgType">DATAGRAM</assign>
      </output>
      <input message="inmsg">
        <assign to="." from="*"></assign>
      </input>
    </operation>
</process>
```

Configure the UMID and Block S Options

About this task

The Sterling B2B Integrator does not support the UMID and block S options, so those must be disabled for all Queues within the MQSeries® Interface.

Complete these steps to disable the **UMID** and **Block S** options:

Procedure

1. Open the SWIFTAlliance Workstation application.
2. Double-click the **MQSeries Interface**.
3. Open each queue and verify the following options:
   - For both From MQ and To MQ queues, **Include UMID** must be set to No.
   - For the To MQ queues, **Include Block S** must be set to Without Block S.

Chapter 18. SWIFT Error Codes

The Translation service produces a Translation Status report. The report contains information about the translation of the document and any compliance errors. Errors from the input side of the map are listed under the Input tab. Errors from the output side of the map are listed under the Output tab.

Compliance errors can also be specified with the extended rule *cerror*.

This section describes translation status errors and error codes that may be generated when using SWIFT.

Note: Please refer to the specific sections of the SWIFT documentation (parts II and III) to view the details of how field tag and code word validation is performed for all field tags.

Text Validation Error Codes

The following table contains the Translation Status report error numbers and their corresponding SWIFT error codes for text validation. For more information on a specific SWIFT error code, see the SWIFT User Handbook.

Table 3. SWIFT error codes for text validation

Translation Status Report Error Number	SWIFT Error Code
901	M50
902	M60
903	T00
904	T01
905	T02
906	T03
907	T04
908	T05
909	T06
910	T07
911	T08
912	T09
913	T10
914	T11
915	T12
916	T13
917	T14
918	T15
919	T16
920	T17

Table 3. SWIFT error codes for text validation (continued)

Translation Status Report Error Number	SWIFT Error Code
921	T18
922	T19
923	T20
924	T21
925	T22
926	T23
927	T24
928	T25
929	T26
930	T27
931	T28
932	T29
933	T30
934	T31
935	T32
936	T33
937	T34
938	T35
939	T36
940	T37
941	T38
942	T39
943	T40
944	T41
945	T42
946	T43
947	T44
948	T45
949	T46
950	T47
951	T48
952	T49
953	T50
954	T51
955	T52
956	T53
957	T54
958	T55

Table 3. SWIFT error codes for text validation (continued)

Translation Status Report Error Number	SWIFT Error Code
959	T56
960	T57
961	T58
962	T59
963	T60
964	T61
965	T62
966	T63
967	T64
968	T65
969	T66
970	T67
971	T68
972	T69
973	T70
974	T71
975	T72
976	T73
977	T74
978	T75
979	T76
980	T77
981	T78
982	T79
983	T80
984	T81
985	T82
986	T83
987	T84
988	T85
989	T86
990	T87
991	T88
992	T89
993	T90
994	T91
995	T92
996	T93

Translation Status Report Error Number	SWIFT Error Code
997	T94
998	T95
999	T96
1000	T97
1001	T98
1002	T99

MUG-textval Rules Specific Error Codes

The following table contains the Translation Status report error numbers and their corresponding SWIFT error codes for SWIFT MUG-textval rules. For more information on a specific SWIFT error code, see the SWIFT User Handbook.

Table 4. SWIFT error codes for MUG-textval rules

Translation Status Report Error Number	SWIFT Error Code
1003	G01
1004	G02
1005	G03
1006	G04
1007	G05
1008	G06
1009	G07
1010	G08
1011	G09
1012	G10
1013	G11
1014	G12
1015	G13
1016	G14
1017	G15
1018	G16
1019	G17
1902	G18
1903	G19
1904	G20
1905	G21
1906	G22
1907	G23
1908	G24

Value-Added Service Messages Special Error Codes

The following table contains the Translation Status report error numbers and the corresponding SWIFT error codes for value-added service messages. For more information on a specific SWIFT error code, see the SWIFT User Handbook.

Table 5. SWIFT error codes for value-added service messages

Translation Status Report Error Number	SWIFT Error Code
1020	B01
1021	B02
1022	B03
1023	B04
1024	B05

Message Syntax and Semantic Rule Codes

The following section contains the error codes for SWIFT message syntax and semantic rules.

The C, D, and E error code ranges refer to rule numbers (these represent the message validation rules), are documented as follows:

- Rules 000-099 as C00-C99
- Rules 100-199 as D00-D99
- Rules 200-299 as E00-E99

Rules 000-099

The following table contains the Translation Status report error numbers and the corresponding SWIFT error codes for message syntax and semantic rules. For more information on a specific SWIFT error code, see the SWIFT User Handbook.

Table 6. SWIFT error codes for message syntax and semantic rules (C00 - C99)

Translation Status Report Error Number	SWIFT Error Code
1025	C00
1026	C01
1027	C02
1028	C03
1029	C04
1030	C05
1031	C06
1032	C07
1033	C08
1034	C09
1035	C10
1036	C11

Table 6. SWIFT error codes for message syntax
and semantic rules (C00 - C99) (continued)

Translation Status Report Error Number	SWIFT Error Code
1037	C12
1038	C13
1039	C14
1040	C15
1041	C16
1042	C17
1043	C18
1044	C19
1045	C20
1046	C21
1047	C22
1048	C23
1049	C24
1050	C25
1051	C26
1052	C27
1053	C28
1054	C29
1055	C30
1056	C31
1057	C32
1058	C33
1059	C34
1060	C35
1061	C36
1062	C37
1063	C38
1064	C39
1065	C40
1066	C41
1067	C42
1068	C43
1069	C44
1070	C45
1071	C46
1072	C47
1073	C48
1074	C49

Table 6. SWIFT error codes for message syntax and semantic rules (C00 - C99) (continued)

Translation Status Report Error Number	SWIFT Error Code
1075	C50
1076	C51
1077	C52
1078	C53
1079	C54
1080	C55
1081	C56
1082	C57
1083	C58
1084	C59
1085	C60
1086	C61
1087	C62
1088	C63
1089	C64
1090	C65
1091	C66
1092	C67
1093	C68
1094	C69
1095	C70
1096	C71
1097	C72
1098	C73
1099	C74
1100	C75
1101	C76
1102	C77
1103	C78
1104	C79
1105	C80
1106	C81
1107	C82
1108	C83
1109	C84
1110	C85
1111	C86
1112	C87

Table 6. SWIFT error codes for message syntax
and semantic rules (C00 - C99) (continued)

Translation Status Report Error Number	SWIFT Error Code
1113	C88
1114	C89
1115	C90
1116	C91
1117	C92
1118	C93
1119	C94
1120	C95
1121	C96
1122	C97
1123	C98
1124	C99

Rules 100-199

The following table contains the Translation Status report error numbers and the corresponding SWIFT error codes for message syntax and semantic rules. For more information on a specific SWIFT error code, see the SWIFT User Handbook.

Table 7. SWIFT error codes for message syntax
and semantic rules (D00 - D99)

Translation Status Report Error Number	SWIFT Error Code
1125	D00
1126	D01
1127	D02
1128	D03
1129	D04
1130	D05
1131	D06
1132	D07
1133	D08
1134	D09
1135	D10
1136	D11
1137	D12
1138	D13
1139	D14
1140	D15
1141	D16
1142	D17

Table 7. SWIFT error codes for message syntax and semantic rules (D00 - D99) (continued)

Translation Status Report Error Number	SWIFT Error Code
1143	D18
1144	D19
1145	D20
1146	D21
1147	D22
1148	D23
1149	D24
1150	D25
1151	D26
1152	D27
1153	D28
1154	D29
1155	D30
1156	D31
1157	D32
1158	D33
1159	D34
1160	D35
1161	D36
1162	D37
1163	D38
1164	D39
1165	D40
1166	D41
1167	D42
1168	D43
1169	D44
1170	D45
1171	D46
1172	D47
1173	D48
1174	D49
1175	D50
1176	D51
1177	D52
1178	D53
1179	D54
1180	D55

Table 7. SWIFT error codes for message syntax and semantic rules (D00 - D99) (continued)

Translation Status Report Error Number	SWIFT Error Code
1181	D56
1182	D57
1183	D58
1184	D59
1185	D60
1186	D61
1187	D62
1188	D63
1189	D64
1190	D65
1191	D66
1192	D67
1193	D68
1194	D69
1195	D70
1196	D71
1197	D72
1198	D73
1199	D74
1200	D75
1201	D76
1202	D77
1203	D78
1204	D79
1205	D80
1206	D81
1207	D82
1208	D83
1209	D84
1210	D85
1211	D86
1212	D87
1213	D88
1214	D89
1215	D90
1216	D91
1217	D92
1218	D93

Translation Status Report Error Number	SWIFT Error Code
1219	D94
1220	D95
1221	D96
1222	D97
1223	D98
1224	D99

Rules 200-299

The following table contains the Translation Status report error numbers and the corresponding SWIFT error codes for message syntax and semantic rules. For more information on a specific SWIFT error code, see the SWIFT User Handbook.

Table 8. SWIFT error codes for message syntax and semantic rules (E00 - E99)

Translation Status Report Error Number	SWIFT Error Code
1225	E00
1226	E01
1227	E02
1228	E03
1229	E04
1230	E05
1231	E06
1232	E07
1233	E08
1234	E09
1235	E10
1236	E11
1237	E12
1238	E13
1239	E14
1240	E15
1241	E16
1242	E17
1243	E18
1244	E19
1245	E20
1246	E21
1247	E22
1248	E23

Table 8. SWIFT error codes for message syntax and semantic rules (E00 - E99) (continued)

Translation Status Report Error Number	SWIFT Error Code
1249	E24
1250	E25
1251	E26
1252	E27
1253	E28
1254	E29
1255	E30
1256	E31
1257	E32
1258	E33
1259	E34
1260	E35
1261	E36
1262	E37
1263	E38
1264	E39
1265	E40
1266	E41
1267	E42
1268	E43
1269	E44
1270	E45
1271	E46
1272	E47
1273	E48
1274	E49
1275	E50
1276	E51
1277	E52
1278	E53
1279	E54
1280	E55
1281	E56
1282	E57
1283	E58
1284	E59
1285	E60
1286	E61

Table 8. SWIFT error codes for message syntax and semantic rules (E00 - E99) (continued)

Translation Status Report Error Number	SWIFT Error Code
1287	E62
1288	E63
1289	E64
1290	E65
1291	E66
1292	E67
1293	E68
1294	E69
1295	E70
1296	E71
1297	E72
1298	E73
1299	E74
1300	E75
1301	E76
1302	E77
1303	E78
1304	E79
1305	E80
1306	E81
1307	E82
1308	E83
1309	E84
1310	E85
1311	E86
1312	E87
1313	E88
1314	E89
1315	E90
1316	E91
1317	E92
1318	E93
1319	E94
1320	E95
1321	E96
1322	E97
1323	E98
1324	E99

Code Word Validation in Generic Fields

The following table contains the Translation Status report error numbers and the corresponding SWIFT error codes for code word validation in generic fields. For more information on a specific SWIFT error code, see the SWIFT User Handbook.

Table 9. SWIFT error codes for word validation in generic fields

Translation Status Report Error Number	SWIFT Error Code
1325	K00
1326	K01
1327	K02
1328	K03
1329	K04
1330	K05
1331	K06
1332	K07
1333	K08
1334	K09
1335	K10
1336	K11
1337	K12
1338	K13
1339	K14
1340	K15
1341	K16
1342	K17
1343	K18
1344	K19
1345	K20
1346	K21
1347	K22
1348	K23
1349	K24
1350	K25
1351	K26
1352	K27
1353	K28
1354	K29
1355	K30
1356	K31
1357	K32
1358	K33

Translation Status Report Error Number	SWIFT Error Code
1359	K34
1360	K35
1361	K36
1362	K37
1363	K38
1364	K39
1365	K40
1366	K41
1367	K42
1368	K43
1369	K44
1370	K45
1371	K46
1372	K47
1373	K48
1374	K49
1375	K50
1376	K51
1377	K52
1378	K53
1379	K54
1380	K55
1381	K56
1382	K57
1383	K58
1384	K59
1385	K60
1386	K61
1387	K62
1388	K63
1389	K64
1390	K65
1391	K66
1392	K67
1393	K68
1394	K69
1395	K70
1396	K71

Table 9. SWIFT error codes for word validation in generic fields (continued)

Translation Status Report Error Number	SWIFT Error Code
1397	K72
1398	K73
1399	K74
1400	K75
1401	K76
1402	K77
1403	K78
1404	K79
1405	K80
1406	K81
1407	K82
1408	K83
1409	K84
1410	K85
1411	K86
1412	K87
1413	K88
1414	K89
1415	K90
1416	K91
1417	K92
1418	K93
1419	K94
1420	K95
1421	K96
1422	K97
1423	K98
1424	K99

Header Validation Error Codes

The following table contains the Translation Status report error numbers and their corresponding SWIFT error codes for header validations that are supported by Sterling B2B Integrator .

Table 10. SWIFT header validation error codes (H and U)

Translation Status Report Error Number	Description of How This Validation is Handled and Enforced in Sterling B2B Integrator	SWIFT Error Code
1425	Enforced by the envelope map.	H01

Table 10. SWIFT header validation error codes (H and U) (continued)

Translation Status Report Error Number	Description of How This Validation is Handled and Enforced in Sterling B2B Integrator	SWIFT Error Code
1426	Inbound. Sterling B2B Integrator generates this error.	H02
1427	Inbound and outbound. Sterling B2B Integrator generates this error.	H03
1434	Not generated or enforced by Sterling B2B Integrator.	H10
1439	Not generated or enforced by Sterling B2B Integrator.	H15
1444	Not generated or enforced by Sterling B2B Integrator.	H20
1445	Outbound. Sterling B2B Integrator generates this error.	H21
1449	Enforced by the envelope map.	H25
1450	Enforced by the envelope map.	H26
1454	Inbound and outbound. Sterling B2B Integrator generates this error.	H30
1464	Inbound and outbound. Sterling B2B Integrator generates this error.	H40
1474	Translator Error Code 1474 (inbound and outbound)—the Sterling B2B Integrator generates this error	H50
1475	Outbound. Sterling B2B Integrator generates this error.	H51
1476	Not generated or enforced by Sterling B2B Integrator.	H52
1479	Not generated or enforced by Sterling B2B Integrator.	H55
1504	Enforced by the envelope map.	H80
1505	Enforced by the envelope map.	H81
1522	Inbound and outbound. Sterling B2B Integrator generates this error.	H98
1523	Enforced by the envelope map.	H99
1524	Enforced by the envelope map.	U00
1525	Enforced by the envelope map.	U01
1526	Enforced by the envelope map.	U02
1527	Not generated or enforced by Sterling B2B Integrator.	U03
1531	Outbound. Sterling B2B Integrator generates this error.	U07
1532	Inbound and outbound. Sterling B2B Integrator generates this error.	U08
1533	Inbound and outbound. Sterling B2B Integrator generates this error.	U09

SWIFT System Messages Error Codes

The following table contains the Translation Status report error numbers and the corresponding SWIFT error codes for system messages. For more information on a specific SWIFT error code, see the SWIFT User Handbook.

Table 11. SWIFT error codes for system messages

Translation Status Report Error Number	SWIFT Error Code
BlockDataUnknown, 25	V02
BlockDataUnknown, 25	V67
MandatoryDataMissing, 100	V03
InsufficientRepeats, 101	V03
TooManyRepeats, 102	V02
TooManyRepeats, 102	V67
IncorrectDataFormat, 110]	V07
DataNotMinLength, 111	V08
DataExceedsMaxLength, 112	V10
InvalidDate, 113	V04
TooManyComponents, 120	V08
TooManyCompositeElements, 121	V08
UnsupportedDataType, 122	V07
DataConversionError, 123	V07
MandatoryBlockMissing, 300	V02
MandatoryGroupMissing, 301	V03
IncorrectBlockSequence, 305	V02
IncorrectBlockSequence, 305	V67
ChoiceBlockNoMatches, 317	V02
ChoiceBlockTooManyMatches, 318	V02
ChoiceBlockTooManyMatches, 317	V67
ChoiceGroupNoMatches, 319	V03
ChoiceGroupTooManyMatches, 320	V02
ChoiceGroupTooManyMatches, 320	V67

Custom Error Codes

The following table contains the Translation Status report error numbers and the corresponding SWIFT custom error codes. These error codes are not part of the standard SWIFT validation (for example, BICPlus validation). For more information on a specific SWIFT error code, see the SWIFT User Handbook.

Table 12. SWIFT custom error codes

Translation Status Report Error Number	SWIFT Error Code
5901	BICPlusIBAN BIC clearing code validation.
5902	Delivered Market Practice MT536 Delta statement rule.

Table 12. SWIFT custom error codes (continued)

Translation Status Report Error Number	SWIFT Error Code
5903	BICPlusIBAN Bank ID validation.
5904	BICPlusIBAN BIC/IBAN combination validation.
5905	SEPA Routing operational readiness status validation.
5906	SEPA Routing adherence status validation.
5907	BICPlusIBAN BIC validation.

SWIFT MX Message Error Codes

The following table contains the Translation Status report error numbers and the corresponding SWIFT error codes for MX messages. For more information on a specific SWIFT error code, see the SWIFT User Handbook.

Table 13. SWIFT error codes for MX messages

Translation Status Report Error Number	SWIFT Error Code
1624	Invalid BIC Code
1625	Invalid BEI Code
1626	Invalid IBAN format or invalid check digits
1627	Invalid Country Code
1628	Invalid active currency
1629	Invalid active or historic currency
1630	Invalid Currency Amount
1631	Invalid BIC or BEI
1632	Invalid Language code
1633	AccruedInterestAmountSignRule
1634	BalanceAtSafekeepingPlaceRule
1635	BalanceForAccountOrSubAccountDetailsRule
1636	Party1Rule
1637	StreetNameAndOrPostOfficeBoxRule
1638	ValueRule
1639	AccumulationPeriodRule
1640	AddressRule
1641	AmountAndOrRateRule
1642	AmountSignRule
1643	BalanceForSubAccountRule
1644	BeneficiaryRule
1645	BulkCashSettlementDetails1Rule
1646	BulkCashSettlementDetails2Rule
1647	BulkCashSettlementDetails3Rule
1648	BulkCashSettlementDetails4Rule
1649	BulkCashSettlementDetails5Rule

Table 13. SWIFT error codes for MX messages (continued)

Translation Status Report Error Number	SWIFT Error Code
1650	BulkCashSettlementDetails6Rule
1651	CalculationBasisRule
1652	CashSettlementDateRule
1653	ClientReferenceRule
1654	CommercialAgreementRule
1655	CorporateRule
1656	CurrencyPredefinedPeriodsRule
1657	CurrencyPriceChangeRule
1658	CurrencyUserDefinedPeriodsRule
1659	CurrentYearRule
1660	CurrentYearSubscriptionDetailsRule
1661	CurrentYearTypeRule
1662	DateOrDateCodeRule
1663	DeliverersCustodianDetailsRule
1664	DeliverersCustodianRule
1665	DeliverersIntermediaryDetailsRule
1666	DesignationRule
1667	ExceptionalCashFlowIndicatorRule
1668	ExchangeConversionRule
1669	MultipleSwitchExecutionRule
1670	ExtendedPartyRole2Rule
1671	ExtendedPartyRoleRule
1672	FinancialInstrumentQuantity1Rule
1673	FloorAmountRule
1674	FutureSettlementDateRule
1675	GoodTIllOrderRule
1676	InvestmentAccountDetailsRule
1677	InvestmentAccountIdentificationRule
1678	InvestmentAccountRule
1679	InvestorRule
1680	MailingIndicatorRule
1681	MessageNameAndReferenceGuideline
1682	MessageNameRule
1683	NameAndAddress1Rule
1684	NetAmountGrossAmountRule
1685	OrderOriginatorEligibility1Rule
1686	OrderOriginatorEligibility3Rule
1687	OrderOriginatorEligibility4Rule
1688	OrderPriceAndForeignExchangeRule

Table 13. SWIFT error codes for MX messages (continued)

Translation Status Report Error Number	SWIFT Error Code
1689	OrderPriceAndLimitOrderRule
1690	OtherCodeRule
1691	OtherReferenceRule
1692	OtherStatusRule
1693	ISARule
1694	PendingAdditionalInformation
1695	PhysicalDeliveryDetailsRule
1696	PhysicalTransferDetailsRule
1697	PorfolioRule
1698	PreviousOrOtherReferenceOrAccountApplicationIdentificationRule
1699	PreviousReferenceRule
1700	PriceDetailsRule
1701	ReceiversCustodianDetailsRule
1702	ReceiversCustodianDetailsRule
1703	ReceiversIntermediaryDetailsRule
1704	ReferenceRule
1705	ReferredPlacementAgentRule
1706	RegistrationAddressIndicator1Rule
1707	RegistrationAddressIndicator2Rule
1708	RelatedPartiesDetailsRule
1709	RelatedReferenceRule
1710	RequestedSettlementCurrencyRule
1711	RoundingDirectionAndModulusRule
1712	RoundingMethodRule
1713	RoundingRule
1714	SettlementCurrency1Rule
1715	SettlementCurrency2Rule
1716	SettlementCurrency3Rule
1717	SettlementCurrency4Rule
1718	SettlementCurrency5Rule
1719	StatisticsCurrency1Rule
1720	StatisticsCurrency2Rule
1721	StopPriceAndStopOrderRule
1722	TaxRule
1723	TotalAmountYearToDateRule
1724	TotalNumberRule
1725	TotalRedemptionAmountRule
1726	TotalUnitsNumberRule
1727	TransactionOnAccountOrSubAccountDetailsRule

Table 13. SWIFT error codes for MX messages (continued)

Translation Status Report Error Number	SWIFT Error Code
1728	TransactionOnSubAccountRule
1729	TransactionTypeLegExecutionIdentificationRule
1730	TransactionTypeLegIdentificationRule
1731	TypeAmountRateRule
1732	UserDefinedRule
1733	BilateralBalanceRule
1734	CreditDebitIndicator1Rule
1735	EntryAmountCreditDebitIndicator1Rule
1736	EntryAmountCreditDebitIndicator2Rule
1737	InstructedAmountCreditDebitIndicator1Rule
1738	InstructedAmountCreditDebitIndicator2Rule
1739	InstructedAmountCurrencyRule
1740	PaymentInstructionStatusRule
1741	ReturnCriteriaAndOrSearchCriteriaRule
1742	SettlementAmountCreditDebitIndicator1Rule
1743	SettlementAmountCreditDebitIndicator2Rule
1744	SettlementAmountCurrencyRule
1745	TransactionCreditDebitIndicatorRule
1746	TransferValueDateRule
1747	OrderOriginatorEligibilityGuideline
1748	NomineeAccountServicerRule
1749	TotalSubscriptionAmountRule
1750	AccountIdentificationRule
1751	BilateralBalanceRule
1752	BilateralLimitCounterparty1Rule
1753	BilateralLimitCounterparty2Rule
1754	BilateralLimitCounterparty3Rule
1755	BilateralLimitCounterparty4Rule
1756	BilateralLimitRule
1757	CashAccountIdentificationGuideline
1758	ChargeRule
1759	CounterpartyIdentification1Rule
1760	CounterpartyIdentification2Rule
1761	CreditAccountRule
1762	CreditDebitIndicator1Rule
1763	CreditDebitIndicatorGuideline
1764	CurrencyGuideline
1765	DebitAccountRule
1766	DebitCreditIndicatorGuideline

Table 13. SWIFT error codes for MX messages (continued)

Translation Status Report Error Number	SWIFT Error Code
1767	DeliverersIntermediaryGuideline
1768	EntryAmountCreditDebitIndicator1Rule
1769	EntryAmountCreditDebitIndicator2Rule
1770	ExchangeConversionRule
1771	ExchangeOrConversionRateRule
1772	FloorAmountRule
1773	InstructedAmountCreditDebitIndicator1Rule
1774	InstructedAmountCreditDebitIndicator2Rule
1775	InstructedAmountCurrencyRule
1776	IssuerAndOrMessageNameRule
1777	LimitCriteriaGuideline
1778	MailingIndicatorRule
1779	MandatoryValueRule
1780	MemberIdentificationRule
1781	ModifyStandingOrderRules
1782	PaymentInstructionStatusRule
1783	PaymentMessageTypeRule
1784	ReceiversIntermediaryGuideline
1785	ReturnCriteriaAndOrSearchCriteriaRule
1786	SearchAndReturnCriteriaAndStatementReportRule
1787	SettlementAmountCreditDebitIndicator1Rule
1788	SettlementAmountCreditDebitIndicator2Rule
1789	SettlementAmountCurrencyRule
1790	StreetNameAndOrPostOfficeBoxRule
1791	TaxRule
1792	TransactionCreditDebitIndicatorRule
1793	TransferValueDateRule
1794	AvailabilityAndTypeRule
1795	CertificateIdentificationAndOrTaxTypeRule
1796	DomainAndProprietary1Rule
1797	DomainAndProprietary2Rule
1798	FamilyAndSubFamilyRule
1799	ReturnReasonRule
1800	StatusAndBookingDateRule
1801	ERISAEligibilityRule
1802	LimitedPartnershipRule
1803	LinkedMessageReferenceRule
1804	NameRule
1805	NoSpecifiedReasonRule

Table 13. SWIFT error codes for MX messages (continued)

Translation Status Report Error Number	SWIFT Error Code
1806	OrderOriginatorEligibility2Rule
1807	SidePocketOrderRule
1808	UnitizedRule
1809	DetailedReportParameterRule
1810	ISINIdentificationRule
1811	ParameterRule
1812	ReportParameterRule1
1813	ReportParameterRule2
1814	ReportParameterRule3
1909	AccountIdentificationGuideline
1910	CallDataVersusCallableIndicatorRule
1911	ConvertibleBondVersusConvertibleIndicatorRule
1912	CounterpartyReferenceRule
1913	FinancialInstrumentIdentificationGuideline
1914	PutDateVersusPutableIndicatorRule
1915	SettlementPlaceRule
1916	SocialSecurityNumberGuideline
1917	SpecificPreviousYearsGuidelines
1918	CoexistenceIssuerSchemeNameRule
1919	CreditDebitIndicatorGuideline
1920	DescriptionUsageRule
1921	ISINGuideline
1922	ReferenceGuideline
1923	SupplementaryDataRule
1924	EffectiveSettlementDateRule
1925	ExtensionRule
1926	RequestedSettlementDateRule
4811	AmendmentIndicatorTrueRule
4812	AmendmentIndicatorFalseRule
4817	SettlementMethodAgentRule
4818	SettlementMethodClearingRule
4836	InstructedReimbursementAgentAccountRule
4837	InstructingReimbursementAgentAccountRule
4838	ThirdReimbursementAgentAccountRule
4839	ThirdReimbursementAgentRule
4861	GroupCancellationAndNumberOfTransactionsRule
4871	GroupCancellationAndReasonRule
4874	SettlementMethodCoverRule
4875	SettlementMethodCoverAgentRule

Table 13. SWIFT error codes for MX messages (continued)

Translation Status Report Error Number	SWIFT Error Code
4876	ReturnReasonRule
4897	ForwardBalanceAndAvailabilityRule
4903	DomainOrPropertiesRule
4904	GroupCancellationTrueAndPaymentInformationRule
4905	GroupCancellationFalseAndPaymentInformationRule
4912	MessageOrGroupCaseRule
4913	GroupOrPaymentInformationCaseRule
4914	MessageOrTransactionCaseRule
4915	GroupOrTransactionCaseRule
4916	PaymentInformationOrTransactionCaseRule
4917	MessageOrPaymentInformationCaseRule
4953	TradeMinusRule
4954	NonWorkingDayAdjustmentRule
4955	DealingCutOffTimeFrameRule
4956	ReportParameterRule1
4991	DescriptionPresenceRule
4992	OtherIdentificationPresenceRule
4993	ISINPresenceRule
5003	ReturnCriteriaAndOrSearchCriteriaRule
5004	ReturnCriteriaAndOrSearchCriteriaRule
5005	PaymentInstructionStatusRule
5006	TransferValueDateRule
5007	CreditDebitIndicatorRule
5008	ReturnCriteriaAndOrSearchCriteriaRule
5009	SearchAndReturnCriteriaAndStatementReportRule
5010	ReturnCriteriaAndOrSearchCriteriaRule
5012	BilateralBalanceRule
5013	BilateralLimitCounterpartyRule
5014	BilateralLimitCounterparty2Rule
5015	CounterpartyIdentification1Rule
5016	CounterpartyIdentification2Rule
5017	BilateralLimitCounterparty3Rule
5018	BilateralLimitCounterparty4Rule
5019	SettlementAmountCurrencyRule
5020	InstructedAmountCurrencyRule
5021	InstructedAmountCreditDebitIndicator1Rule
5022	InstructedAmountCreditDebitIndicator2Rule
5023	SettlementAmountCreditDebitIndicator1Rule
5024	SettlementAmountCreditDebitIndicator2Rule

Table 13. SWIFT error codes for MX messages (continued)

Translation Status Report Error Number	SWIFT Error Code
5025	EntryAmountCreditDebitIndicator1Rule
5026	EntryAmountCreditDebitIndicator2Rule
5027	TransactionCreditDebitIndicatorRule
5028	CreditAccountRule
5029	DebtorAccountRule
5031	ReturnCriteriaAndOrSearchCriteriaRule
5060	MessageOrReportPaginationRule
5061	MessageOrStatementPaginationRule
5062	MessageOrNotificationPaginationRule
5064	PartialHoldingUnitsRule
5065	PartialHoldingUnits2Rule
5066	PartialHoldingUnits3Rule
5067	HoldingCertificateNumberRule
5069	AmountRule
5070	UnitRule
5071	DividendPercentageRule
5072	NewPriceReportRule
5073	CompletePriceCancellation1Rule
5074	CompletePriceCancellation2Rule

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing

Legal and Intellectual Property Law

IBM Japan Ltd.

1623-14, Shimotsuruma, Yamato-shi

Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be

incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

J46A/G4

555 Bailey Avenue

San Jose, CA 95141-1003

U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© IBM 2011. Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 2011.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium and the Ultrium Logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Connect Control Center®, Connect:Direct®, Connect:Enterprise®, Gentran®, Gentran®:Basic®, Gentran:Control®, Gentran:Director®, Gentran:Plus®, Gentran:Realtime®, Gentran:Server®, Gentran:Viewpoint®, Sterling Commerce™, Sterling Information Broker®, and Sterling Integrator® are trademarks or registered trademarks of Sterling Commerce®, Inc., an IBM Company.

Other company, product, and service names may be trademarks or service marks of others.

IBM ®

Printed in USA