Sterling Standards Library

Using EDIFACT

Version 5.4

Sterling Commerce
An IBM Company

Contents

Using EDIFACT with the Application

EDIFACT Overview

EDIFACT (Electronic Data Interchange For Administration, Commerce, and Transport) is the international EDI standard developed under the United Nations. EDIFACT has been adopted by the International Organization for Standardization (ISO) as the ISO standard ISO 9735. The EDIFACT standard provides a set of syntax rules to structure data, an interactive exchange protocol (I-EDI), and standard messages which allow multi-country and multi-industry exchange.

This table describes how the application supports EDIFACT:

Functionality	Details	Action You Need to Take to Implement EDIFACT
Services	◆ EDIFACT Enveloping service translates messages, determines which business processes need to run to apply EDI envelopes, and starts those business processes. This service improves performance of EDI EDIFACT by consolidating EnvelopeUNH, EnvelopeUNG, and EnvelopeUNB into a single service. ◆ EDIFACT Deenveloping service removes the envelopes from inbound messages.	Configure the EDIFACT services in the application Graphic Process Modeler (GPM). **Note:** As a way to help reduce the number of envelopes you need to create and use, the EDI Enveloping and EDI Deenvelope services support use of an asterisk (*) as a wildcard character in mandatory envelope fields for EDIFACT. By using wildcards, you can set up one set of envelopes that can be used for multiple trading partners. If certain trading partners have specific requirements, you can still have envelopes that pertain just to them, and the EDI Enveloping service chooses the envelope that is the best match. In other words, the envelope that has the most matches to specific fields in the data (for example Receiver ID, Receiver ID Qualifier), is the one selected.
Envelopes	◆ Inbound EDIFACT envelope wizards to implement inbound EDIFACT. ◆ Outbound EDIFACT envelope wizards to implement outbound EDIFACT.	◆ Create the appropriate EDIFACT envelopes for the EDIFACT messages you are sending and receiving.

Functionality	Details	Action You Need to Take to Implement EDIFACT
Predefined business processes	◆ EDIFACTDeenvelope business process extracts EDI documents from an EDIFACT interchange-level envelope and locates an associated contract, according to the content of the documents. EDIFACTDeenvelope then starts the appropriate business process. With additional configuration, EDIFACTDeenvelope can also reconcile TA1 and 997 reports and generate the interchange TA1 and functional group 997 acknowledgments. ◆ EDIFACTEnvelope business process applies an EDIFACT envelope to one or more EDIFACT messages and then uses the envelope data to translate and process them.	Configure the predefined business processes as necessary.
Map Editor	◆ Map Editor wizard enables you to generate a file layout for you using the EDIFACT message that you select. Included in the EDIFACT map are groups, segments, composite elements, and elements that are defined by EDIFACT. ◆ Properties dialog boxes that enable you to define and modify EDI map components for EDIFACT. ◆ 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. ◆ Additional mapping operations as needed.	◆ Download Map Editor. ◆ Create a map or maps to translate your EDIFACT messages.
Correlation Search	Search by EDIFACT information.	Use the EDI Correlation search functionality to search for EDIFACT correlation information.
Reports	Report by EDIFACT standard	Use the EDI Translation Detail report to search for details of inbound and outbound EDIFACT messages and easily create useful reports on your inbound and outbound EDIFACT messaging.

Prerequisite Knowledge

The audience using this software should be familiar with the application and the EDIFACT standard.

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

Using EDIFACT with the Application

To use EDIFACT with the application, you must complete the following tasks:

1. Create inbound and outbound EDIFACT envelopes.

2. Use the predefined EDIFACT business processes or create your own business processes.

3. Create any necessary code lists and maintain them as needed.

4. Create your EDIFACT map or maps, linking your proprietary file format to the appropriate EDIFACT message format.

5. To track EDIFACT messages, use the EDI Correlation Search functionality.

6. To report on inbound and outbound EDIFACT message flow, use the EDI Translation Detail report features.

EDIFACT Business Processes

Overview

To help you accomplish your business goals, the application provides two predefined enveloping business processes that are used by the application to implement EDIFACT processing: EDIFACTEnvelope and EDIFACTDeenvelope. These predefined business processes are initiated by services during EDIFACT processing and do not require you to modify them unless you want to use interchange and functional group CNTRL acknowledgments.

The following table lists business goals for the predefined EDIFACT business processes:

Business Process	Business Goals
EDIFACTDeenvelope	Extracts EDI documents from an EDIFACT interchange and locates an associated contract, according to the content of the envelopes. EDIFACTDeenvelope then starts the appropriate business process.
EDIFACTEnvelope	Applies an EDIFACT envelope to one or more EDIFACT messages and then uses the envelope data to translate and process them.

EDIFACTDeenvelope Business Process

The EDI Deenvelope business process includes the EDI Deenveloping service to support non-printable characters as EDI delimiters and perform the following steps:

1. Splits documents into interchanges and puts delimiters sent in the ISA (or UNA) into process data.

2. Starts the EDIFACT Deenveloping, EDIFACT Deenveloping, or CII Deenveloping business processes, as appropriate.

If exceptions occur when running an EDI Deenvelope business process, the application generates an EDI Compliance Report.

This table lists the configuration parameters for the EDIFACTDeenvelope business process:

Parameter	Default	Description
Document Tracking	False	When document tracking is enabled for a business process, tracking information is carried with the message throughout the process, and the tracking information is persisted about the message regardless of the persistence level you configured globally for the application.
Start Mode	Async	The business process can be configured to execute in synchronous mode and asynchronous mode.
Transaction	False	Indicates whether the transaction will be sent. False indicates that the transaction will not be sent.

Parameter	Default	Description
Set onfault processing	False	Onfault processing allows the process to immediately execute the application fault activity specified in the process, even if the process has not yet reached that step in the process. For example, if a process fails at step 3, but the on-fault activity is specified in step 7, if onfault processing is enabled, the process proceeds to the step 7 on-fault rather than halting at step 3.
Queue	4	The application enables you to set performance optimizations by queue, defining queue levels to allocate resources. This number indicates the previously allocated queue level that you want for this business process model for processing.
Persistence Level	Full	The level of data to retain for generating a status report that describes each step that the business process completes. System default indicates that, for the data, configuration is already defined in the application to retain data.
Recovery Level	Manual	The level of recovery for this business process if the business process should halt during execution. Manual requires you to resume or restart the business process manually.
Document Storage Type	System Default	The level of document storage for messages that process when the business process runs. System Default specifies to store messages in the file system or database, according to how you configured archiving and purging in the application.
Life Span	Life Span Days — 2 Life Span Hours — 0 Life Span Type — System Level Removal Method — Archive	The length of time, in days and hours, to retain the data in the application, along with the life span type and removal method.
Complete by Deadline	None Available **Note:** To set a deadline you must change it in the business process.	Complete by – The deadline time, in hours and minutes, by which the business process must complete process once it starts. ◆ First Notification: Hours and Minutes – Whether to receive notification before a business process deadline. ◆ Second Notification: Hours and Minutes – Whether to receive another notification before a business process deadline.
Event Reporting Level	None	The level of event reporting that is retrieved for this business process when it runs. Full specifies to generate events for the business process, including the business process start and end time, start and end times for all services or services running as a result of this business process, and any resulting errors and exceptions.

The following BPML code makes up the EDI deenveloping business process:

```
<process name="EDIFACTDeenvelope">
```

```
<rule name="ung_exists">
  <condition>UNG_I.DONE = 0</condition>
</rule>

<rule name="process_unh">
  <condition>UNG_FOUND = "false"</condition>
</rule>

<rule name="process_ung">
  <condition>UNG_FOUND = "true"</condition>
</rule>

<rule name="unh_exists">
  <condition>UNH_I.DONE = 0</condition>
</rule>

<rule name="contract">
  <condition>CONTRACT_FOUND = "YES"</condition>
</rule>

<rule name="generate_ack">
  <condition>GENERATE_CONTRL = "YES"</condition>
</rule>

<rule name="unh_ack">
  <condition>MessageType = "CONTRL"</condition>
</rule>

<sequence>
  <operation>
    <participant name="DeenvelopeUNB"/>
    <output message="Xout" >
      <assign to="." from="*"></assign>
    </output>
    <input message="Xin" >
      <assign to="." from="*"></assign>
    </input>
  </operation>

  <onFault code="Interchange-Non-Compliant">
    <choice>
      <select>
          <case ref="generate_ack" activity="generate_CONTRL_UNG" />
      </select>

      <sequence name="generate_CONTRL_UNG">
        <operation>
          <participant name="GenerateCONTRL"/>
          <output message="Xout">
            <assign to="." from="*"></assign>
          </output>
          <input message="Xin" >
            <assign to="." from="*"></assign>
          </input>
        </operation>
```

```
        <operation>
          <participant name="EDIEncoder"/>
          <output message="Xout">
            <assign to="." from="*"></assign>
          </output>
          <input message="Xin" >
            <assign to="." from="*"></assign>
          </input>
        </operation>
      </sequence>
    </choice>
</onFault>

<choice>
  <select>
    <case ref="process_ung" activity="deenvelope_ung"/>
    <case ref="process_unh" activity="deenvelope_unh"/>
  </select>

  <sequence name="deenvelope_ung">

    <sequence name="foreach_ung">
      <operation>
        <participant name="ForEachDocument"/>
        <output message="Xout">
          <assign to="." from="*"></assign>
          <assign to="ITERATOR_NAME">UNG_I</assign>
          <assign to="DOCUMENT_NAME_PREFIX">GROUP</assign>
        </output>
        <input message="Xin" >
          <assign to="." from="*"></assign>
        </input>
      </operation>

      <choice>
        <select>
          <case ref="ung_exists" activity="process_ung"/>
        </select>

        <sequence name="process_ung">

          <sequence name="deenvelope_ung_unh">
            <operation>
              <participant name="DeenvelopeUNG"/>
              <output message="Xout" >
                <assign to="." from="*"></assign>
              </output>
              <input message="Xin" >
                <assign to="." from="*"></assign>
              </input>
            </operation>

            <operation>
              <participant name="DeenvelopeUNH"/>
              <output message="Xout" >
                <assign to="." from="*"></assign>
```

```
        </output>
        <input message="Xin" >
          <assign to="." from="*"></assign>
        </input>
      </operation>

    <onFault code="Group-Non-Compliant">
      <assign to="BREAK">YES</assign>
    </onFault>

    <onFault code="Transaction-Non-Compliant">
      <assign to="BREAK">YES</assign>
    </onFault>
</sequence>

<sequence name="ung_foreach_unh">
  <operation>
    <participant name="ForEachDocument"/>
    <output message="Xout">
      <assign to="." from="*"></assign>
      <assign to="ITERATOR_NAME">UNH_I</assign>
      <assign to="DOCUMENT_NAME_PREFIX">$UNG_I.NAME</assign>
    </output>
    <input message="Xin" >
      <assign to="." from="*"></assign>
    </input>
  </operation>

  <choice>
    <select>
      <case ref="unh_exists" activity="ung_process_unh"/>
    </select>

    <sequence name="ung_process_unh">

      <sequence name="ung_process_document">
        <choice>
          <select>
            <case ref="unh_ack" activity="reconcile_ack_UNG" />
          </select>
          <sequence name="reconcile_ack_UNG">
            <operation>
              <participant name="ReconcileCONTRL" />
              <output message="Xout" >
                <assign to="." from="*"></assign>
              </output>
              <input message="Xin" >
                <assign to="." from="*"></assign>
              </input>
            </operation>
          </sequence>
        </choice>

        <choice>
          <select>
```

```
                             <case ref="contract"
activity="ung-unh_invoke_contract_workflow" />
                         </select>
                         <sequence name="ung-unh_invoke_contract_workflow">

        <!-- **************************************** -->
        <!-- Only send to child what is required -->
        <!-- **************************************** -->
        <operation name="release">
      <participant name="ReleaseService"/>
       <output message="Xout">
       <assign to="TARGET" from="string('message_to_child')"/>
       </output>
       <input message="Xin"/>
         </operation>

                         <assign to="message_to_child"
from="/ProcessData/*[not(boolean(@SCIObjectID))]"/>
                         <assign to="message_to_child/PrimaryDocument"
from="PrimaryDocument/@*"/>

                         <operation>
                           <participant name="InvokeBusinessProcessService" />
                           <output message="Xout">
                             <assign to="INVOKE_MODE">ASYNC</assign>
                             <assign to="." from="*"></assign>
                           </output>
                           <input message="Xin" >
                             <assign to="." from="*"></assign>
                           </input>
                         </operation>
                       </sequence>
                     </choice>
                   </sequence>

                   <repeat ref="ung_foreach_unh"/>

                 </sequence>
               </choice>

           </sequence>

           <repeat ref="foreach_ung"/>

         </sequence>
       </choice>
     </sequence>

     <choice>
       <select>
         <case ref="generate_ack" activity="generate_CONTRL_UNG" />
       </select>

       <sequence name="generate_CONTRL_UNG">
         <operation>
           <participant name="GenerateCONTRL"/>
```

```
            <output message="Xout">
              <assign to="." from="*"></assign>
            </output>
            <input message="Xin" >
              <assign to="." from="*"></assign>
            </input>
          </operation>
          <operation>
            <participant name="EDIEncoder"/>
            <output message="Xout">
              <assign to="." from="*"></assign>
            </output>
            <input message="Xin" >
              <assign to="." from="*"></assign>
            </input>
          </operation>
        </sequence>
      </choice>

   </sequence>


   <sequence name="deenvelope_unh">
     <sequence>
       <operation>
         <participant name="DeenvelopeUNH"/>
         <output message="Xout" >
      <assign to="." from="*"></assign>
         </output>
         <input message="Xin" >
      <assign to="." from="*"></assign>
         </input>
       </operation>

<onFault code="Transaction-Non-Compliant">
   <assign to="BREAK">YES</assign>
</onFault>
     </sequence>

     <choice>
       <select>
          <case ref="generate_ack" activity="generate_CONTRL_NO_UNG" />
       </select>

       <sequence name="generate_CONTRL_NO_UNG">
         <operation>
           <participant name="GenerateCONTRL"/>
           <output message="Xout">
             <assign to="." from="*"></assign>
           </output>
           <input message="Xin" >
             <assign to="." from="*"></assign>
           </input>
         </operation>
         <operation>
           <participant name="EDIEncoder"/>
```

```
            <output message="Xout">
              <assign to="." from="*"></assign>
            </output>
            <input message="Xin" >
              <assign to="." from="*"></assign>
            </input>
          </operation>
      </sequence>
   </choice>

   <sequence name="foreach_deenveloped_unh">
     <operation>
  <participant name="ForEachDocument"/>
  <output message="Xout">
    <assign to="." from="*"></assign>
    <assign to="ITERATOR_NAME">UNH_I</assign>
    <assign to="DOCUMENT_NAME_PREFIX">SET</assign>
  </output>
  <input message="Xin" >
    <assign to="." from="*"></assign>
  </input>
     </operation>

     <choice>
  <select>
    <case ref="unh_exists" activity="process_deenveloped_unh"/>
  </select>

  <sequence name="process_deenveloped_unh">

    <sequence name="process_document">
      <choice>
<select>
  <case ref="unh_ack" activity="reconcile_ack_NO_UNG" />
</select>
<sequence name="reconcile_ack_NO_UNG">
  <operation>
    <participant name="ReconcileCONTRL" />
    <output message="Xout" >
      <assign to="." from="*"></assign>
    </output>
    <input message="Xin" >
      <assign to="." from="*"></assign>
    </input>
  </operation>
</sequence>
      </choice>

      <choice>
        <select>
          <case ref="contract" activity="unh_invoke_contract_workflow" />
        </select>
        <sequence name="unh_invoke_contract_workflow">

        <!-- ************************************** -->
        <!-- Only send to child what is required -->
```

```
<!-- *************************************** -->
<operation name="release">
   <participant name="ReleaseService"/>
<output message="Xout">
  <assign to="TARGET" from="string('message_to_child')"/>
</output>
<input message="Xin"/>
   </operation>

   <assign to="message_to_child"
from="/ProcessData/*[not(boolean(@SCIObjectID))]"/>
   <assign to="message_to_child/PrimaryDocument" from="PrimaryDocument/@*"/>

<operation>
  <participant name="InvokeBusinessProcessService" />
  <output message="Xout">
    <assign to="INVOKE_MODE">ASYNC</assign>
    <assign to="." from="*"></assign>
  </output>
  <input message="Xin" >
    <assign to="." from="*"></assign>
  </input>
</operation>

      </sequence>
    </choice>
  </sequence>

  <repeat ref="foreach_deenveloped_unh"/>

</sequence>
    </choice>
  </sequence>


    </sequence>


  </choice>
  </sequence>
</process>
```

The following scenario shows how the EDIFACTDeenvelope business process can deenvelope EDIFACT documents:

1. The File System adapter collects a file from an EDIFACT collection folder and invokes the EDIFACTDeenvelope business process.

2. The EDI Deenveloping service determines that the file contains an EDIFACT document and starts the EDIFACTDeenvelope business process.

3. The EDIFACTDeenvelope business process removes the envelopes to make available the transaction set envelope for the EDIFACT document and determine the next action to perform.

4. For documents with CII Syntax Rule requirements, the EDIFACTDeenvelope business process parses the message group and uses the values indicated in the message group header to locate a matching envelope. The envelope then determines the next action to perform.

Note: If the EDIFACTDeenvelope business process does not locate a matching envelope, the business process stops. The application generates an EDI Compliance Report that describes the reason that the EDIFACTDeenvelope business process could not locate the envelope.

Before Using EDIFACTDeenvelope

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

1. Create the trading profile, including a contract.

2. Create necessary envelopes.

3. Create a business process that initiates the EDIFACTDeenvelope business processes.

   When creating a business process that enables inbound ST (or UNH) envelopes to translate transactions when running EDIFACTDeenvelope business processes, consider the following:

   ◆ Delimiters sent in the ISA or UNA segments are placed into process data in the application.

   ◆ The GPM makes available parameters that represent the delimiters, which you must configure to enable the envelopes to translate transaction.

   ◆ You must create an XPath expression that describes the location of envelopes and apply it to the Translation service.

EDIFACTEnvelope Business Process

The EDIFACTEnvelope business process is initiated when it is called by another business process or the EDI Enveloping service.

The EDIFACTEnvelope business processes performs the following primary activities:

1. Starting with one or more EDI documents, the EDIEnveloping services applies envelope properties at the document level to each document.

2. The EDIFACTEnvelope service takes one or more transaction sets and applies a functional group envelope.

3. The EDIFACTEnvelope service takes a functional group and applies the interchange-level envelope and searches for a matching contract.

4. Using the interchange-level envelope data, the service either looks up a contract or runs a business process.

This table lists the configuration parameters for the EDIFACTEnvelope business process:

Parameter	Default	Description
Document Tracking	False	When document tracking is enabled for a business process, tracking information is carried with the message throughout the process, and the tracking information is persisted about the message regardless of the persistence level you configured globally for the application.
Start Mode	Async	The business process can be configured to execute in synchronous mode and asynchronous mode.
Transaction	False	Indicates whether the transaction will be sent. False indicates that the transaction will not be sent.
Set onfault processing	False	Onfault processing allows the process to immediately execute the on-fault activity specified in the process, even if the process has not yet reached that step in the process. For example, if a process fails at step 3, but the on-fault activity is specified in step 7, if onfault processing is enabled, the process proceeds to the step 7 on-fault rather than halting at step 3.
Queue	4	The application enables you to set performance optimizations by queue, defining queue levels to allocate resources. This number indicates that previously allocated queue level that you want for this business process model for processing.
Persistence Level	System Default	The level of data to retain for generating a status report that describes each step that the business process completes. System default indicates that, for the data, configuration is already defined in the application to retain data.
Recovery Level	Manual	The level of recovery for this business process if the business process should halt during execution. Manual requires you to resume or restart the business process manually.
Document Storage Type	System Default	The level of document storage for messages that process when the business process runs. System Default specifies to store messages in the file system or database, according to how you configured archiving and purging in the application.
Life Span	Life Span Days — 2 Life Span Hours — 0 Life Span Type — System Level Removal Method — Archive	The length of time, in days and hours, to retain the data in the application, along with the life span type and removal method.

Parameter	Default	Description
Complete by Deadline	None Available **Note:** To set a deadline you must change it in the business process.	Complete by – The deadline time, in hours and minutes, by which the business process must complete process once it starts. ◆ First Notification: Hours and Minutes – Whether to receive notification before a business process deadline. ◆ Second Notification: Hours and Minutes – Whether to receive another notification before a business process deadline.
Event Reporting Level	Full	The level of event reporting that is retrieved for this business process when it runs. Full specifies to generate events for the business process, including the business process start and end time, start and end times for all services or services running as a result of this business processes, and any resulting errors and exceptions.

The following BPML code makes up the EDIFACTEnvelope business process:

```
<process name="EDIFACTEnvelope">

  <rule name="contract">
    <condition>CONTRACT_FOUND = "YES"</condition>
  </rule>

  <sequence>
    <sequence>
      <operation>
        <participant name="EnvelopeUNH" />
        <output message="Xout" >
          <assign to="." from="*"></assign>
        </output>
        <input message="Xin" >
          <assign to="." from="*"></assign>
        </input>
      </operation>
      <onFault code="Translation-Error-Some-Docs">
        <assign to="BREAK">YES</assign>
      </onFault>
      <onFault>
        <operation>
          <participant name="BPExceptionService"/>
          <output message="Xout" >
            <assign to="exceptionCode" from="'Translation Error'"/>
          </output>
          <input message="Xin"/>
          </operation>
      </onFault>
    </sequence>
    <operation>
      <participant name="EnvelopeUNG" />
      <output message="Xout" >
        <assign to="." from="*"></assign>
      </output>
```

```
      <input message="Xin" >
        <assign to="." from="*"></assign>
      </input>
    </operation>
    <operation>
      <participant name="EnvelopeUNB" />
      <output message="Xout" >
        <assign to="." from="*"></assign>
      </output>
      <input message="Xin" >
        <assign to="." from="*"></assign>
      </input>
    </operation>

    <choice>
      <select>
        <case ref="contract" activity="invoke_contract_workflow" />
      </select>
      <sequence name="invoke_contract_workflow">
        <operation>
          <participant name="InvokeBusinessProcessService" />
          <output message="Xout">
            <assign to="INVOKE_MODE">ASYNC</assign>
            <assign to="." from="*"></assign>
          </output>
          <input message="Xin" >
            <assign to="." from="*"></assign>
          </input>
        </operation>
      </sequence>
    </choice>

  </sequence>
</process>
```

Before Using the EDIFACTEnvelope Business Process

Before you use the EDIFACTDeenvelope business process, complete the following task:

1. Create inbound envelopes for your EDIFACT messages.
2. Create outbound envelopes for your EDIFACT messages.

EDIFACT Enveloping

EDIFACT Enveloping Overview

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 EDIFACT with your trading partners.

During the envelope creation process, you need to assign the envelope name, description, and parameters, such as unique identification numbers (control numbers).

Note: Each time a user adds, modifies, or deletes an envelope, the action is logged to create an audit trail based on the user who performed the event.

Each envelope type has a specific properties page for the envelope and other pages that enable you to specify additional requirements for the envelope. Properties for the following envelopes can be found in this documentation:

✦ EDIFACT Inbound Syntax 4 UNG UNE Envelope Properties, Group Level

✦ EDIFACT Inbound Syntax 4 UNB UNZ Envelope Properties, Interchange Level

✦ EDIFACT Inbound Syntax 4 UNH UNT Envelope Properties, Transaction Level

✦ EDIFACT Outbound Syntax 4 UNG UNE Envelope Properties, Group Level

✦ EDIFACT Outbound Syntax 4 UNB UNZ Envelope Properties, Interchange Level

✦ EDIFACT Outbound Syntax 4 UNH UNT Envelope Properties, Transaction Level

Document Lifespan

In previous releases, the document lifespan default was zero so that when the workflow expired, all associated documents were purged/archived with the workflow. Now the lifespan for ASC X12 documents awaiting acknowledgement is configurable, and the default is 30 days.

Note: You can change the default lifespan by editing the document.lifespan property in the enveloping.properties file. The document lifespan of the outbound document is automatically reset to zero after the acknowledgement for the document is received or if the user manually accepts the acknowledgement.

Envelope Structure

The application supports the use of many EDI protocols, including EDIFACT. The EDIFACT protocol has three levels of envelopes:

✦ Interchange (outermost) – Contains an interchange header and trailer, and all the data sent from one sender to one receiver in the same transmission.

✦ Functional group (middle) – Contains a group header and trailer that surrounds a group of transaction sets of the same type.

✦ Transaction set (innermost) – Contains the standard message surrounded by a header and trailer record.

This illustration shows the structure for the protocols that have 3 levels of envelopes:
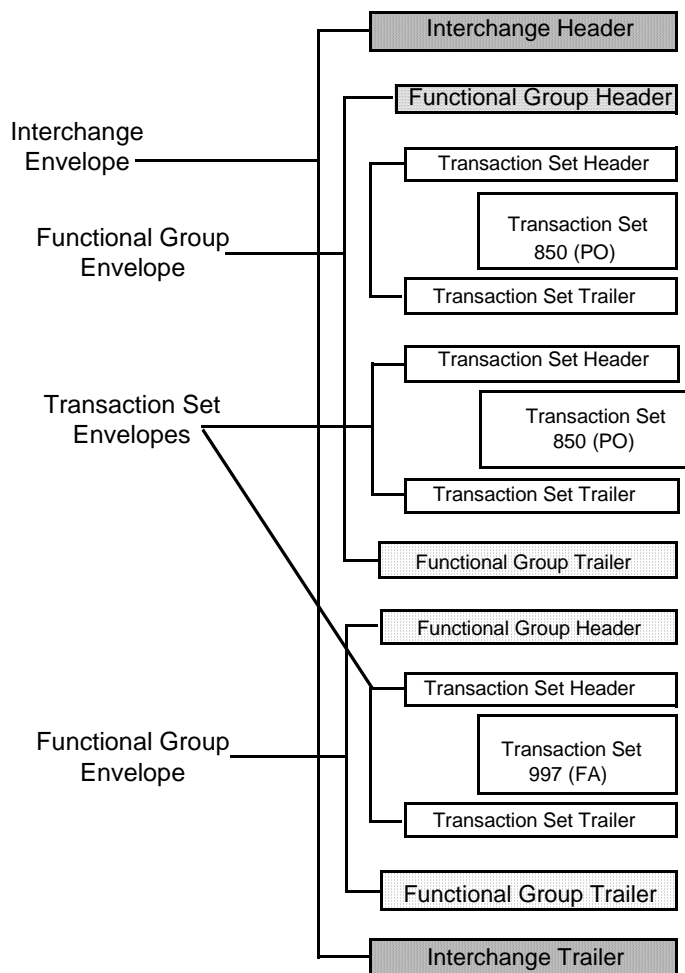
Base Envelopes

The application uses a concept of a *base envelope* as a starting point to create a new envelope. The base envelope maintains a link to the new envelope that inherited its properties. If you modify the base envelope, all related envelopes (that is, ones that inherited the base envelope properties) change, as well.

Note: If you plan to create many envelopes using base envelopes, it is recommended that you do not use the base envelopes in production. You may want to document how your base envelopes are linked to other envelopes.

The default envelope is the version of the document envelope that the application uses. You can specify a default envelope only if there are two or more versions of the same envelope. One version must be selected as the default.

Types of Envelopes

There are many types of envelopes provided in the application. Each envelope is defined by its envelope properties. These predefined enveloping and deenveloping business processes are available for you to incorporate into your own business processes. Envelope parameters specify whether the document is inbound or outbound:

✦ *Inbound envelopes* identify documents that come into the application so they can be properly routed. Inbound envelopes also give you the option to translate documents when you choose to check documents for compliance. By choosing to translate documents from within the envelope, you can reduce document processing time because you do not need to specify a separate Translation service step in the business process.

✦ *Outbound envelopes* identify documents so that they can be sent to and received by trading partners.

Using EDIFACT with the Sterling Community Manager (SCM)

If you are using the application with the Sterling Community Manager (SCM), please note the following:

✦ On the Document Envelopes list page, envelopes that are managed by SCM have **[SCM]** appended to their names.

✦ On the Document Envelopes summary page (that you access by selecting an envelope name from the envelope list), you are notified if the envelope is managed by SCM.

✦ When you attempt to edit an envelope managed by SCM, a warning dialog is displayed to indicate that you are editing an SCM resource.

✦ You cannot delete any of the versions of an envelope managed by SCM. The only way to delete these envelopes is through SCM.

✦ You cannot delete SCM-managed resources. For example, you cannot delete an envelope that is managed by SCM from the application.

✦ The import report indicates if an Import operation involving an SCM managed envelope fails.

✦ When application envelopes are converted to SCM, the naming convention is SCI_Envelope_[standard]_[envelope header tag]_[direction]_[Sponsor|Partner].

✦ Only one "next envelope" selection is supported for each envelope. Therefore, two or more agreements that result in conflicting "next envelope" chains will result in a conversion error.

✦ If you edit an envelope in the application user interface to chose a different "next envelope," a conversion error results if an agreement is created or updated using that envelope with a "next envelope" that is different than the envelope selected through the application user interface. In this scenario, any changes you make to the "next envelope" selection through the application user interface cannot be reverted through the execution of a program or update of an agreement in SCM.

✦ If you intend a change to the "next envelope" change to be temporary, you must revert it prior to the conversion of a new or updated agreement that uses that envelope, or you will receive a conversion error.

✦ If you intend a change to the "next envelope" to be permanent, you must terminate the original partner agreement in SCM. If the envelope changes involve sponsor answers, you may need to execute a new sponsor agreement, create a new partner program that uses the appropriate sponsor display step, and create a new partner agreement.

The SCM envelope converter automatically generates the envelope name for each envelope that is defined in the application based on fields used during inbound envelope lookup. Outbound envelopes have a name generation equal to the inbound equivalent. For example, the EDIFACT outbound interchange envelope has a name generation equal to the EDIFACT inbound interchange envelope. In addition to these fields, constant literals are also assigned as prefixes that are different for each envelope. The pattern used in the prefix is as follows:

```
{standard}_{direction}_{envelope start tag}
```

Envelope definitions created in the envelope converter on behalf of SCM are read-only in application (you cannot modify these envelopes).

Using Wildcards

As a way to help reduce the number of envelopes you need to create and use, the EDI Enveloping and EDI Deenveloping services support use of an asterisk (*) as a wildcard character in mandatory envelope fields for the EDIFACT standard. For optional fields, the wildcard value is leaving the field blank. With EDI Enveloping, for optional fields, the wildcard value is equivalent to leaving the field blank. If certain trading partners have specific requirements, you can create envelopes that pertain just to them, and the EDI Enveloping service chooses the envelope that is the best match. That is, the envelope that has the most matches to specific fields in the data (for example, Receiver ID and Receiver ID Qualifier), is the one selected.

Wildcards in Deenveloping (Inbound)

The EDI Deenveloping service receives data, parses the headers, and extracts the EDI information from it. The service searches the available envelopes for the closest match to the EDI data.

When setting up enveloping with wildcards, consider the following:

✦ The Deenveloping service selects the envelope that matches the EDI data most closely. For example, if an envelope has the same Transaction ID as the incoming data, and another envelope has wildcards in all mandatory fields, the envelope with the matching Transaction ID is used.

✦ Sender ID and Receiver ID have priority over other EDI fields. For example, two envelopes are found that have a field that matches one EDI field from the data. One envelope matches the value in the Receiver ID field from the data; the other matches the Transaction ID field. The envelope that has the matching Receiver ID is selected by the service and used for processing.

✦ If no envelopes are found that have an exact match to fields in the EDI data, and an envelope with wildcards does exist, the envelope with wildcards is used.

✦ If multiple envelopes have the same matching information (for example, two envelopes match the Sender ID and Receiver ID exactly) the service ends with an error.

✦ The wildcard character is an "all or nothing" parameter. That is, if using asterisk, you cannot enter other characters in the field with it. For example, you can not enter Me* to match MeaslesAnimalHaven and Medeira Foods.

Wildcards in Enveloping (Outbound)

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).

One option is to use the lightweight JDBC adapter to pull fields in from database tables. Whatever method you choose to retrieve the fields, the results must be given to the Correlation service, which places them into process data. Passing the data to the Correlation service must be the last step in the business process before running the EDI Encoder service. When the EDI Encoding service runs, it uses the same best-match process detailed in the Inbound section to determine which envelope to use. When the EDIFACT envelope service runs, any envelope field values set in the Correlation service overrides those values defined in the envelope definition.

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.

Creating an EDIFACT Envelope

You can create an EDIFACT envelope either by using the application or by using Sterling Community Manager (SCM) Integration. See the SCM Integration documentation for more information on creating envelopes using that product.

To create a new envelope using the application:

1. From the **Administration** menu, select **Trading Partner** > **Document Envelopes** > **Envelopes**.

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

3. On the Envelope Standards page, select **EDIFACT** and click **Next**:

4. Select the level of envelope you want to create, including inbound or outbound, and click **Next**.

5. Do one of the following:

   ◆ To create this envelope from a base envelope, select the Base envelope from the Select Envelope list, and click **Next**.

   Note: All the properties are pre-filled but you can update them as needed.

   ◆ To create a new envelope, select the Not Applicable from the Select Envelope list, 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.

8. Click **Finish** to add the envelope.

9. Click **Return** to continue.

Updating an EDIFACT Envelope

The application enables you to modify any information in your envelopes, except the name of the envelope.

Note: You can also edit EDIFACT envelopes details from SCM by either terminating the agreement and re-executing the program with Save-As or using the Modify response option in SCM.

To update envelope properties:

1. From the **Administration** menu, select **Trading Partner** > **Document Envelopes** > **Envelopes**.

2. Find the envelope using the Search or List function.

3. From the list of envelopes, click **source manager** next to the envelope you want to update.

4. Click **edit** next to the envelope you want to update.

5. Update envelope properties and settings as necessary, clicking **Next** to continue.

6. Click **Finish** to update the envelope.

7. Click **Return** to continue.

Modifying an SCM-Managed EDIFACT Resource

Resources created through SCM are centrally managed by SCM, and you are warned whenever you attempt to modify an SCM-managed EDIFACT resource. Deletion of an SCM Managed EDIFACT resource is not allowed from the application. The application user interface displays the following information to modify an SCM-managed EDIFACT resource:

✦ When you click **Go!** next to **List all configurations** in the Trading Partner Configurations page, all the EDIFACT resources that are managed by SCM have the keyword **[SCM]** appended to their names.

✦ When you click the name of the EDIFACT resource in the profiles list page, there is a line in the information summary page indicating whether the resource is managed by SCM.

✦ When you attempt to edit an EDIFACT resource managed by SCM, a warning dialog box is displayed with the options **OK** or **Cancel** and the following information:

```
Stop! This partner related data is now managed in Sterling Community Manager.
Please change the _____ information in Sterling Community Manager. Changes made
directly to Sterling Integrator are temporary and will be overwritten when updates
from Sterling Community Manager are absorbed. In case you have a critical need to
update it directly, please make sure that later on you make the same update in the
Partner _____information in Sterling Community Manager as soon as possible.
Do you still want to proceed and make a temporary change?
```

Note: Any SCM-managed resource in the application cannot be deleted. The delete option is not displayed in SCM for resources with [SCM] tag. The only way that you can delete SCM-managed resources

in the application is by deleting the SCM agreements corresponding to the resource update operation that was performed in SCM to create these resources in the application.

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 application 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.

Deleting an EDIFACT Envelope

Caution: Before deleting a base envelope, consider the impact on all related envelopes (envelopes that inherited the base envelope properties). You cannot delete an envelope that is managed by SCM using the application. To remove a SCM-managed resource, delete the agreement corresponding to the resource from SCM.

To delete an individual envelope or all versions of an envelope:

1. From the **Administration** menu, select **Trading Partner** > **Document Envelopes** > **Envelopes**.

2. Find the envelope using the Search or List function.

3. Next to the envelope you want to delete, click **source manager**.

4. In Envelope Source Manager page, do one of the following:

   To delete an individual envelope:

   ◆ Next to the version, under Delete, select the check box.

   ◆ Next to Delete Selected Versions, click **Go!**.

   To delete all versions of this envelope, next to Delete All Versions, click **Go!**.

EDIFACT Inbound Syntax 4 UNB UNZ Envelope Properties, Interchange Level

Note: An (*) asterisk indicates that a wildcard value can be used with that parameter (for mandatory fields, the wildcard value is an (*) asterisk). 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.

The following table describes EDIFACT inbound Syntax 4 UNB UNZ envelope properties at the interchange level:

Field	Description
* Sender ID	Interchange ID as it should appear on the interchange header segment sent to this company or division. Valid value is 12 standard characters. Required.
* Sender Organization	The sender organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Identification Code Qualifier (Sender ID)	Coded qualifier of the sender ID as it should appear on the interchange header segment sent to this company or division. Valid value is four standard characters. Optional.
* Interchange Sender Internal Identification	Internal interchange ID as it should appear on the interchange header segment sent to this company or division. Valid value is 12 standard characters. Optional.
* Interchange Sender Internal Sub-identification	Subinterchange ID as it should appear on the interchange header segment sent to this company or division. Valid value is 12 standard characters. Optional.
* Recipient ID	Interchange ID as it should appear on the interchange header segment received from this company or division. Valid value is 12 standard characters. Required.
* Receiver Organization	The receiver organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Identification Code Qualifier (Recipient ID)	Coded qualifier of the recipient ID as it should appear on the interchange header segment received from this company or division. Valid value is four standard characters. Optional.
* Interchange Recipient Internal Identification	Internal interchange ID as it should appear on the interchange header segment received from this company or division. Valid value is 12 standard characters. Optional.
* Interchange Recipient Internal Sub-identification	Subinterchange ID as it should appear on the interchange header segment received from this company or division. Valid value is 12 standard characters. Optional.

Field	Description
* Test Indicator	Mode of operation to use for this envelope. Required. Valid values are: ◆ None (default) ◆ Test ◆ Syntax Only Test ◆ Echo Request ◆ Echo Response ◆ Any Indicator (wildcard)
Use global control number	Whether this envelope definition must use a global control number to uniquely identify transactions. No indicates that the envelope definition must use its local global control number. Required. Valid values are: ◆ Yes (default) ◆ Yes (and generate name from data) ◆ No
* Global Group Reference Number	The global group reference number. This parameter is displayed only when **Use global control number** is set to **Yes**. The value entered here is the name of the global control number file. Required.
* Local Group Reference Number	The local group reference number. This parameter is displayed only when **Use global control number** is set to **No**. The value entered here is numeric from one to fourteen digits. Required.
Perform Control Number Sequence Checking	Performs an incremental sequence check. Required. Valid values are Yes and No (default).
Perform Duplicate Control Number Checking	Determine control number duplications. Required. Valid values are Yes and No (default).
If sequence or duplicate checking, EDI Post Processor should	How the EDI Post Processor service should perform sequence and duplicate checking. When multiple interchanges for the same trading partner are processed in parallel, subprocesses for transactions will always be a direct child to the process that did the deenveloping. If the interchange is out of sequence, the post processor re-runs until the time-out is reached. Required. Valid values: ◆ Process all interchanges in the EDI Sequence Check Queue (recommended and the default) ◆ Process only the current interchange
Maximum age of Control Number History Records in days	Maximum days to retain a history of control numbers to use for duplication determinations. Valid value is nine standard characters. Optional.
Retain Enclosing Envelope	Whether to copy the envelope segments (UNB and UNZ) into each group or message extracted from the interchange. Required. Valid values are Yes and No.
Business Process for Non-Compliant Documents	Business process to be used when non-compliant documents are encountered. Required.

Field	Description
Primary Name Format	Information to include when generating a name for a global number and finding the correct number to assign based on that name. Optional. Information options include: ◆ Test Indicator
* Global Interchange Control Reference	Interchange control reference number used globally. Select a previously defined number to reference. Valid value is nine standard characters. Required.
Handling of non-compliant Interchanges	Process to follow when non-compliant interchanges are encountered. Required. Valid values: ◆ Accept ◆ Reject (default)
Generate an acknowledgement	Generate an acknowledgement that notifies the sender that the receiver has received either a non-compliant interchange or an interpretable interchange transmission. Required. Valid values are Yes (default) and No.
Acknowledgement Detail Level	Level of detail to which acknowledgements are sent. If you select Segment level, the segment and message levels are acknowledged. Optional. Valid values are: ◆ Interchange Level ◆ Group Level ◆ Message Level ◆ Segment Level ◆ Data Element Level (default)
Send acknowledgement immediately	Send specified level of acknowledgement immediately after interchange transmission occurs. Valid values are Yes and No (default). Required.
Acceptor Lookup Alias format for generated CONTRLs	The format of accepter lookup alias for generated CONTRL messages. Optional. Valid values are: ◆ CONTRL (default) ◆ CONTRL_[syntax version] ◆ CONTRL_[test_indicator] ◆ CONTRL_[syntax version]_[test_indicator] ◆ [message type (first one if multiple)]
* Local Interchange Control Reference	Interchange control reference number used locally. Select a previously defined number to reference. Valid value is nine standard characters. Required.
Write S002 and S003 subelements	Whether to write the S002 (Interchange Sender) and S003 (Interchange Receiver) subelements. Valid values are Yes and No (default). Required.

Field	Description
Syntax 1,2,3 Action Response (UCI 0083)	This parameter is valid if you are using EDIFACT Syntax 3 or lower, and UNB 0031 is set to one (or Generate Acknowledgements is selected). Allows you to send receipt for UCI 0083. Optional. Valid values are: ◆ Acknowledged or Rejected (compliance based) (default) ◆ Received

EDIFACT Inbound Syntax 4 UNG UNE Envelope Properties, Group Level

Note: An (*) asterisk indicates that a wildcard value can be used with that parameter (for mandatory fields, the wildcard value is an (*) asterisk). 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.

The following table describes EDIFACT inbound Syntax 4 UNG UNE envelope properties at the group level:

Field	Description
* Sender ID	Coded identifier of the company or division sending the message group. Valid value is 12 standard characters. Required.
* Sender Organization	The sender organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Identification Code Qualifier (Sender ID)	Coded qualifier of the sender ID as it should appear on the interchange header segment sent to this company or division. Valid value is four standard characters. Optional.
* Recipient ID	Coded identifier of the company or division receiving the message group. Valid value is 12 standard characters. Required.
* Receiver Organization	The receiver organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Identification Code Qualifier (Recipient ID)	Coded qualifier of the recipient ID as it should appear on the interchange header segment sent to this company or division. Valid value is four standard characters. Optional.
Use global control number	Whether this envelope definition must use a global group number to uniquely identify transactions. No indicates that the envelope definition must use its own global group number. Required. Valid values are: ◆ Yes ◆ Yes - Generate name from data ◆ No
* Global Group Reference Number	Interchange control reference number used globally. Select a previously defined number to reference. Valid value is nine standard characters. Optional.
Use Security	Use previously checked in trusted certificates to verify a secured document transmission. Valid values are Yes and No. Required. **Note:** This option must be set to Yes when using secure AUTACK messages in EDIFACT.

EDIFACT Inbound Syntax 4 UNH UNT Envelope Properties, Transaction Level

Note: An (*) asterisk indicates that a wildcard value can be used with that parameter (for mandatory fields, the wildcard value is an (*) asterisk). 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.

The following table describes EDIFACT inbound Syntax 4 UNH UNT envelope properties at the transaction level:

Field	Description
* Sender ID (from group or interchange)	Interchange ID as it should appear on the transaction sent to this company or division. Valid value is 12 standard characters. Required.
* Sender Organization	The sender organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
Identification Code Qualifier (Sender ID) (from group or interchange)	Coded qualifier of the sender ID as it should appear on the from group or interchange header segment sent to this company or division. Valid value is four standard characters. Optional.
* Recipient ID (from group or interchange)	Interchange ID as it should appear on the transaction received from this company or division. Valid value is 12 standard characters. Required.
* Receiver Organization	The receiver organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
Identification Code Qualifier (Recipient ID) (from group or interchange)	Coded qualifier of the recipient ID as it should appear on the from group or interchange header segment sent to this company or division. Valid value is four standard characters. Optional.
* Message Type	Transaction message type as determined by the information type in the header of the message group, which includes the message or by the information type in the transaction message. Valid value is six standard characters. Required. If using AUTACK, create a separate AUTACK inbound transaction envelope with a message type of AUTACK so you can configure the trusted certificate you want to use when validating the digital signature.
* Message Version Number	Version number of the standard message. Valid value is three standard characters. Required.
* Message Release Number (Required for EDIFACT, optional for ODETTE)	Release number of the standard message. Valid value is three standard characters. Required for EDIFACT. Optional for ODETTE.
* Controlling Agency, Coded	Code identifying a controlling agency. This field works in conjunction with Message Group ID (UNG01) and Message Version Number (UNG07) to identify the controlling agency responsible for definition of message group ID in message version number. Optional.

Field	Description
* Association Assigned Code	Assigned associated code for the agency controlling the name of the sending EDI service provider and relaying the message group on the sending side. Valid value is alphanumeric code with six standard characters. Optional.
* Message Type Sub-Function Identification	Coded identifier of a sub-function of a message type. Valid value is six standard characters. Optional.
Use global control number	Whether this envelope definition must use a global control number to uniquely identify transactions. No indicates that the envelope definition must use its own global control number. Required. Valid values are: ◆ Yes ◆ Yes (and generate name from data) ◆ No
Test Indicator	Operation mode to use for this envelope. Required. Valid values are: ◆ None (default) ◆ Test ◆ Syntax Only Test ◆ Echo Request ◆ Echo Response ◆ Any Indicator (wildcard)
Perform Control Number Sequence Checking	Compare the sequence of control numbers in the data with the Control Number parameter for this envelope. Required. Valid values are Yes and No (default).
Perform Duplicate Control Number Checking	Determine control number duplications. Required. Valid values are Yes and No (default).
Maximum age of Control Number History Records in days	Maximum days to retain a history of control numbers to use for duplication determinations. Valid value is nine standard characters. Optional.
Retain Enclosing Envelope	Copy the envelope segments (UNB and UNZ) into each message extracted from the transaction. Required. Valid values are Yes (default) and No.
Batch transactions received within a functional group into one output document.	Put all similar transactions from a functional group into one output document. For example, all invoices would be put into one document. Valid values are Yes (default) and No. The default is to output each transaction to a separate document. Required.

Field	Description
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:
	◆ Message Type
	◆ Message Version and Release Number
	◆ Test Indicator
	Note: If you are using a generated global control number (**Use global control number** is set to Yes (and generate name from data)), the application 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).
	◆ 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:
	◆ Message Type
	◆ Message Version and Release Number
	◆ Test Indicator
	Note: If you are using a generated global control number (**Use global control number** is set to Yes (and generate name from data)), the application 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).
	◆ 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.

Field	Description
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:
	◆ Message Type
	◆ Message Version and Release Number
	◆ Test Indicator
	Note: If you are using a generated global control number (**Use global control number** is set to Yes (and generate name from data)), the application 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).
	◆ 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 Reference Number	Transaction group control reference number used globally. Select a previously defined number to reference. Valid value is nine standard characters. Required.
Handling of non-compliant Messages	Process to follow when non-compliant transactions are encountered. Required.
	Valid values are:
	◆ Accept compliant and non-compliant transactions
	◆ Accept compliant transactions, reject non-compliant transactions (default)
Compliance Check Document	Check the transaction set body for compliance. Valid values are Yes (default) and No. Required.
	Note: This only applies to data compliance, not to security/certificate compliance.
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.
Map Name	Which map to use to perform a compliance check (if **Compliance Check document** is set to Yes and **Map Name Mode** set to Specify). The map must already be checked in to the application. Optional.

Field	Description
Primary Name Format	Check boxes to instruct what information to include when generating and matching a name for the map. Optional. Select from: ◆ EDI Standard ◆ Sender ID Code Qualifier ◆ Sender ID ◆ Recipient ID Code Qualifier ◆ Recipient ID ◆ Message Type ◆ Message Version and Release Number ◆ Test Indicator **Note:** If you are using a generated map name (**Map Name Mode** is set to Generate from data), the application 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). ◆ 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. **Note:** Only occurs if **Generate an error if no matching map is found** is set to Yes.

Field	Description
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:
	◆ EDI Standard
	◆ Sender ID Code Qualifier
	◆ Sender ID
	◆ Recipient ID Code Qualifier
	◆ Recipient ID
	◆ Message Type
	◆ Message Version and Release Number
	◆ Test Indicator
	Note: If you are using a generated map name (**Map Name Mode** is set to Generate from data), the application 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).
	◆ 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.
	Note: Only occurs if **Generate an error if no matching map is found** is set to Yes.

Field	Description
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: ◆ EDI Standard ◆ Sender ID Code Qualifier ◆ Sender ID ◆ Recipient ID Code Qualifier ◆ Recipient ID ◆ Message Type ◆ Message Version and Release Number ◆ Test Indicator **Note:** If you are using a generated map name (**Map Name Mode** is set to Generate from data), the application 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). ◆ 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. **Note:** Only occurs if **Generate an error if no matching map is found** is set to Yes.
Generate an error if no generated map name exists in the system	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.
Translate transactions	Generate transactions according to the standard format from data groups in an EDI system and carry out reverse processing. By choosing to translate documents from within the envelope, you can reduce document processing time because you do not need to specify a separate Translation service step in the business process. Valid values are Yes and No (default).
Validate translation output	Validate translated output (that is, documents sent to trading partners). Valid values are Yes (default) and No. Required.

Field	Description
Determine the Business Process by	Select a business process that should run with messages as the primary document. Optional. Valid values are: ♦ Looking up contract dynamically ♦ Specifying a contract ♦ Specifying a business process ♦ Generating the business process name from the data
For each document	As specified in the Determine the Business Process. Required. Valid values are: ♦ Invoke the business process ♦ Set the business process name in the process data
Primary Name Format	Check boxes to instruct what information to include when generating and matching a name for the business process. Optional. Select from: ♦ EDI Standard ♦ Sender ID Code Qualifier ♦ Sender ID ♦ Recipient ID Code Qualifier ♦ Recipient ID ♦ (0065) Message Type ♦ Message Version and Release Number ♦ Test Indicator **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 application 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). ♦ 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. **Note:** Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Field	Description
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:

- ◆ EDI Standard

- ◆ Sender ID Code Qualifier

- ◆ Sender ID

- ◆ Recipient ID Code Qualifier

- ◆ Recipient ID

- ◆ (0065) Message Type

- ◆ Message Version and Release Number

- ◆ Test Indicator

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 application 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).

- ◆ 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.

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

Field	Description
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:
	◆ EDI Standard
	◆ Sender ID Code Qualifier
	◆ Sender ID
	◆ Recipient ID Code Qualifier
	◆ Recipient ID
	◆ (0065) Message Type
	◆ Message Version and Release Number
	◆ Test Indicator
	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 application 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).
	◆ 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.
	Note: 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.
Contract List	Select a previously created contract to associate with this envelope.
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	How to perform business process data extraction. Required.
	Valid values:
	◆ Determined by business process (default)
	◆ Extract to a file system directory
	◆ Extract to a mailbox

Field	Description
Data Extraction Directory	Directory for data extraction. Optional.
Data Extraction Filename	Filename for data extraction. Optional.
Data Extraction Mailbox	Mailbox for data extraction. Optional.
Data Extraction Mailbox Message Name	Mailbox message name for data extraction. Optional.
Error Business Process Mode	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. The generated name will end with FIN.
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: ◆ EDI Standard ◆ Sender ID Code Qualifier ◆ Sender ID ◆ Recipient ID Code Qualifier ◆ Recipient ID ◆ (0065) Message Type ◆ Message Version and Release Number ◆ Test Indicator **Note:** If you are using a generated error business process name (**Error Business Process Mode** is set to Generate from data), the application 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). ◆ 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. **Note:** Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Field	Description
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:

- EDI Standard

- Sender ID Code Qualifier

- Sender ID

- Recipient ID Code Qualifier

- Recipient ID

- (0065) Message Type

- Message Version and Release Number

- Test Indicator

Note: If you are using a generated error business process name (**Error Business Process Mode** is set to Generate from data), the application 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).

- 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.

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

Field	Description
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:
	◆ EDI Standard
	◆ Sender ID Code Qualifier
	◆ Sender ID
	◆ Recipient ID Code Qualifier
	◆ Recipient ID
	◆ (0065) Message Type
	◆ Message Version and Release Number
	◆ Test Indicator
	Note: If you are using a generated error business process name (**Error Business Process Mode** is set to Generate from data), the application 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).
	◆ 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.
	Note: 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 **Error Business Process Mode** is set to Generate from 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.
Use Security	Use previously checked in trusted certificates to verify a secured document transmission. Valid values are Yes and No (default). Required.
	Note: This option must be set to Yes when using secure AUTACK messages in EDIFACT.

Field	Description
Scope of Security Application	Used for AUTACK security only. Security header and message body referenced in the digital certificate. Required. Valid values are:
	◆ 1 - Security header and message body. Use this option if you want the AUTACK embedded in the transaction that it is securing.
	◆ 3 - Whole related message, package, group, or interchange. Use this option if you want the AUTACK to reference the data transaction or the group it is securing. When selecting this option, you have two options for certificate parameters:
	- Use a certificate in USC segment or generate a certificate name
	- Use a specified certificate only
Security Certificate	Select a previously checked in system certificate to associate with this envelope. Required.

EDIFACT Outbound Syntax 4 UNB UNZ Envelope Properties, Interchange Level

Note: An (*) asterisk indicates that a wildcard value can be used with that parameter (for mandatory fields, the wildcard value is an (*) asterisk). 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.

The following table describes EDIFACT outbound Syntax 4 UNB UNZ envelope properties at the interchange level:

Field	Description
* Syntax Identifier	Coded identifier of syntax rules. Valid value is four standard characters. Required.
* Syntax Version Number	Version number of syntax rules. Valid value is one standard characters. The current value is 4. Required.
* Service Code List Directory Version Number	Version number of the service code list directory. Optional.
* Character Encoding, Coded	Coded character encoding used in the interchange. Optional.
* Sender ID	Coded identifier of the company or division sending the interchange. Valid value is 12 standard characters. Required.
* Sender Organization	The sender organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Identification Code Qualifier (Sender ID)	Coded qualifier of the sender ID as it should appear on the interchange header segment sent to this company or division. Valid value is four standard characters. Optional.
* Interchange Sender Internal Identification	Internal interchange ID as it should appear on the interchange header segment sent to this company or division. Valid value is 12 standard characters. Optional.
* Interchange Sender Internal Sub-identification	Internal interchange sub ID as it should appear on the interchange header segment sent to this company or division. Valid value is 12 standard characters. Optional.
* Recipient ID	Coded identifier of the company or division receiving the interchange. Valid value is 12 standard characters. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Receiver Organization	The receiver organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Identification Code Qualifier (Recipient ID)	Coded qualifier of the recipient ID as it should appear on the interchange header segment sent to this company or division. Valid value is three standard characters. Optional.
* Interchange Recipient Internal Identification	Internal interchange ID as it should appear on the interchange header segment received from a company or division. Valid value is 12 standard characters. Optional.

Field	Description
* Interchange Recipient Internal Sub-identification	Internal interchange sub ID as it should appear on the interchange header segment received from a company or division. Valid value is 12 standard characters. Optional.
Use global control number	Whether this envelope definition must use a global control number to uniquely identify transactions. No indicates that the envelope definition must use its own global control number. Optional. Valid values are: ◆ Yes (default) ◆ Yes (and generate name from data) ◆ No
* Recipient Reference/Password	Reference or password as agreed between the communicating trading partners. Optional.
* Recipient Reference/Password Qualifier	Coded qualifier and password of the receiver as it should appear on the message group header sent to this company or division. Valid value is four standard characters. Optional.
* Application Reference	Reference identifier of the application used by the sender. Valid value is 15 standard characters. Optional.
* Processing Priority Code	Code determined by the sender requesting processing priority for the interchange. Optional.
* Acknowledgement Request	Request an acknowledgement that notifies the sender that the receiver has received an interpretable interchange transmission. Optional.
* Acknowledgement overdue after (hours)	Amount of time, in hours, within which you must receive an acknowledgement. Valid value is four numeric characters. Optional.
* Acknowledgement overdue after (minutes)	Amount of time, in minutes, within which you must receive an acknowledgement. Valid value is four numeric characters. Optional.
* Interchange Agreement Identifier	Identification by name or code of the type of agreement under which the interchange takes place. Optional.
* Test Indicator	The structural level in the envelope containing the test indicator is a test. Optional.
Primary Name Format	Information to include when generating a name for a primary global number and finding the correct number to assign based on that name. Information includes: ◆ Test Indicator Check the appropriate information to use in the format. Optional.
* Global Interchange Control Reference	Interchange control reference number used globally. Select a previously defined number to reference. Valid value is nine standard characters. Required. **Note:** Only displayed if Use Global Control Number is set to Yes.
* Local Interchange Control Reference	Interchange control reference number used locally. Select a previously defined number to reference. Valid value is fourteen standard characters. Required. Default is 1. **Note:** Only displayed if Use Global Control Number is set to No.

Field	Description
Limit Interchange Size	Maximum size allowed for an interchange before a single interchange is split into multiple interchanges that are each under the maximum size limitation.
	Valid values are:
	◆ No
	◆ Use smallest transaction limit (default)
	Note: The **Use smallest transaction limit** setting is specific to the limitation imposed by the standard.
	◆ Specify size limit
	◆ Use global default in enveloping.properties
	Required.
	Note: If you select **Use transaction set default in enveloping.properties**, you need to specify the size limit directly (in the **Maximum Interchange Size** parameter) or by setting properties in the enveloping.properties property file, which contains a property for each supported standard and more specific properties for each message type (for example, INVOIC for the EDIFACT standard) that can override the standard property.
Invoke Business Process for Each Interchange	Whether a business process is invoked for each interchange generated. Valid values are Yes and No (default). Required.
	Note: If **Invoke Business Process for Each Interchange** is set to **No**, the service places the name of the business process specified in the envelope into process data using the key WFD_NAME. This business process is then invoked by the Invoke Business Process Service. If **Invoke Business Process For Each Interchange** is set to **Yes**, the service directly bootstraps an instance of the business process specified in the envelope, and does not set WFD_NAME in process data.
Maximum Interchange Size	Specify the maximum size of each individual interchange. Required.
	Note: Only displayed if you selected **Specify size limit** for the **Limit Interchange Size** parameter. If you choose to specify a maximum value, the value you specify is checked to make sure it is at least 173 bytes based on the header and trailer size for EDIFACT standard.
	Note: For performance reasons, the size is estimated for some components of the interchange (for example, control numbers) when determining the size. For this reason, the actual enforced size limit is slightly smaller than the value you specify.
Specify Non-Standard delimiters	Whether delimiters other than the default delimiters are associated with the Syntax Identifier, UNOA, or UNOB. Valid values are Yes and No (default). Required.
* UNA Segments	Incorporate UNA segments in this envelope. Valid values are Yes and No (default). Required.
Stream Segments	Insert an end-of-line character after each segment record. Valid values are Yes and No (default). Required.
Data Element Separator	EDI delimiter to use to separate elements. Optional.

Field	Description
Segment Terminator	EDI delimiter to use to indicate the end of segments. Optional.
Component Element Separator	EDI delimiter used to separate components. Optional.
Release Character	Character to use to release an EDI delimiter. Optional.
Repetition Separator	Character to use to separate a group of repeated elements. Optional.
Decimal Separator	Character to use to identify different multi details. Optional.
Encode Document	Whether to specify character encoding for the document. Required. Valid values are Yes and No (default).
Encoding	Type of character encoding for the message group that includes this header. This field is used with the Encoding field. Valid value is six standard characters. Required only if Encode Document is set to Yes.
Determine the Business Process by	Select a business process that should run with extracted EDIFACT outbound Syntax 4 UNB/UNZ messages as the primary document. Required. Valid values: ◆ Looking up contract dynamically ◆ Specifying a contract ◆ Specifying a business process ◆ Generating the business process name from the data
Business Process List	Select a business process to run when this envelope is processed. Optional.
Contract List	Select a previously created contract to associate with this envelope. Optional.

Field	Description
Primary Name Format	Information to include when generating a name for a primary global number and finding the correct number to assign based on that name. Information includes:

<div style="margin-left: 2em;">

- EDI Standard

- Sender ID Code Qualifier

- Sender ID

- Recipient ID Code Qualifier

- Recipient ID

- (0065) Message Type

- Message Version and Release Number

- Test Indicator

Check the appropriate information to use in the format. Optional.

Note: If you are using a generated map name (**Generating the business process name from the data** is set to Generate from data), the application 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).

- 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.

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

</div>

Field	Description
First Backup Name Format	Information to include when generating a name for a backup global number and finding the correct number to assign based on that name. Information includes:
	◆ EDI Standard
	◆ Sender ID Code Qualifier
	◆ Sender ID
	◆ Recipient ID Code Qualifier
	◆ Recipient ID
	◆ (0065) Message Type
	◆ Message Version and Release Number
	◆ Test Indicator
	Check the appropriate information to use in the format. Optional.
	Note: If you are using a generated map name (**Generating the business process name from the data** is set to Generate from data), the application 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).
	◆ 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.
	Note: Only occurs if **Generate an error if no generated business process name exists in the system** is set to Yes.

Field	Description
Second Backup Name Format	Information to include when generating a name for a secondary backup global number and finding the correct number to assign based on that name. Information includes: ◆ EDI Standard ◆ Sender ID Code Qualifier ◆ Sender ID ◆ Recipient ID Code Qualifier ◆ Recipient ID ◆ (0065) Message Type ◆ Message Version and Release Number ◆ Test Indicator Check the appropriate information to use in the format. Optional. **Note:** If you are using a generated map name (**Generating the business process name from the data** is set to Generate from data), the application 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). ◆ 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. **Note:** 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	Generate an error message if it cannot locate the specified business process name. Valid values are Yes (default) and No.
Extraction Options	How the messages is extracted from the envelope. Required. Valid values are: ◆ Determined by a business process (default) ◆ Extract to a file system directory ◆ Extract to a mailbox
Data Extraction Directory	Directory for data extraction. Optional.
Data Extraction Filename	Filename for data extraction. Optional.
Data Extraction Mailbox	Mailbox for data extraction. Required.
Data Extraction Mailbox Message Name	Mailbox message name for data extraction. Optional.

EDIFACT Outbound Syntax 4 UNG UNE Envelope Properties, Group Level

Note: An (*) asterisk indicates that a wildcard value can be used with that parameter (for mandatory fields, the wildcard value is an (*) asterisk). 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.

The following table describes EDIFACT outbound Syntax 4 UNG UNE envelope properties at the group level:

Field	Description
Next Envelope	Specifies envelope to use after this envelope. Required.
Create Next Envelope	Adds group level envelope to use as next envelope or edits an existing envelope. Optional.
* Sender ID	Interchange ID as it should appear on the message group header sent to this company or division. Valid value is 12 standard characters. Required.
* Sender Organization	The sender organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Identification Code Qualifier (Sender ID)	Application Group Sender ID Qualifier that identifies the content of the (Group) Sender ID. Valid value is four standard characters. Optional.
* Recipient ID	Interchange ID as it should appear on the message group header received from this company or division. Valid value is 12 standard characters. Required.
* Receiver Organization	The receiver organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Identification Code Qualifier (Recipient ID)	An Application Group Recipient ID Qualifier that identifies the content of the (Group) Recipient ID. Valid value is four standard characters. Optional.
Use global control number	Whether this envelope definition must use a global control number to uniquely identify message group headers. No indicates that the envelope definition must use its own local control number. Required. Valid values are: ◆ Yes (default) ◆ Yes (and generate name from data) ◆ No
Controlling Agency, Coded	Code identifying a controlling agency. This field works in conjunction with Message Group ID (UNG01) and Message Version Number (UNG07) to identify the controlling agency responsible for definition of message group ID in message version number. Optional.
Application Password	Password used to access the application of the receiver. Valid value is 15 standard characters. Optional.

Field	Description
Use Security	Turns on AUTACK security for the group. The AUTACK inbound transaction envelope will store the certificate reference or the way the certificate can be generated from the data used in validating the digital signature for the group. Use previously checked in trusted certificates to verify a secured document transmission. Valid values are Yes and No (default). Required. **Note:** This option must be set to Yes when using secure AUTACK messages in EDIFACT.
Security Certificate	The security certificate. Default is B2BHttp. Required. **Note:** Only displayed if Use Security is set to Yes.
Security Algorithm	The algorithm for security purposes. Default is MD5 with RSA. Required. **Note:** Only displayed if Use Security is set to Yes.
Use of Algorithm	How the algorithm should be used. Default is 1-Owner Hashing. Required. **Note:** Only displayed if Use Security is set to Yes.
Security Service	The service provided by the selected security. Default is Non-repudiation of origin. Required. **Note:** Only displayed if Use Security is set to Yes.
Scope of Security Application	The scope of the security application (what it checks). Default is 3 - Whole related message, package, group or interchange. Required. **Note:** Only displayed if Use Security is set to Yes.
Response Type	The type of response. Default is 1- No acknowledgement. Required. **Note:** Only displayed if Use Security is set to Yes.
Filter Function	The type of filter used. Default is 2 - Hexadecimal Filter. Required. **Note:** Only displayed if Use Security is set to Yes.

EDIFACT Outbound Syntax 4 UNH UNT Envelope Properties, Transaction Level

Note: An (*) asterisk indicates that a wildcard value can be used with that parameter (for mandatory fields, the wildcard value is an (*) asterisk). 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.

The following table describes EDIFACT outbound Syntax 4 UNH UNT envelope properties at the transaction level:

Field	Description
Sender ID	Coded identifier of the company or division sending the transaction. Valid value is 12 standard characters. Required.
* Sender Organization	The sender organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Identification Code Qualifier (Sender ID)	Coded qualifier of the sender ID as it should appear on the interchange header segment sent to this company or division. Valid value is four standard characters. Optional.
Recipient ID	Coded identifier of the company or division receiving the interchange. Valid value is 12 standard characters. Required.
* Receiver Organization	The receiver organization for an SCM-managed envelope. Required for SCM-managed envelopes; Optional for envelopes managed by the application.
* Identification Code Qualifier (Recipient ID)	Coded qualifier of the recipient ID as it should appear on the interchange header segment sent to this company or division. Valid value is three standard characters. Optional.
Next Envelope	Envelope to apply after this envelope. Required.
Create Next Envelope	Identifies if group envelopes are to be generated. ◆ If next envelope is UNG/UNE then group envelopes are to be generated (that is, UNB, UNG, UNH, UNT, UNE, UNZ). ◆ If UNB/UNZ then there are no group envelopes (that is, UNB, UNH, UNT, UNZ). Optional. Valid values are: ◆ UNG/UNE Syntax 4 ◆ UNB/UNZ Syntax 4
Control number handling	How to handle control numbers. Required. Valid values are: ◆ Global (default) ◆ Global (and generate name from data) ◆ Local ◆ Always start with 1

Field	Description
* Message Type	Transaction message type as determined by the information type in the header of the message group, which includes the message or by the information type in the transaction message. Valid value is six standard characters. Required.
* Message Version Number	Version number of the standard message. Valid value is three standard characters. Required.
* Message Release Number (Required for EDIFACT, optional for ODETTE)	Release number of the standard message. Valid value is three standard characters. Required for EDIFACT and AUTACK. Optional for ODETTE.
* Controlling Agency, Coded	Agency controlling the code representing the name of the sending EDI service provider and relaying the message group on the sending side. Valid value is alphanumeric code with three standard characters. Optional.
* Association Assigned Code	Assigned associated code for the agency controlling the name of the sending EDI service provider and relaying the message group on the sending side. Valid value is alphanumeric code with six standard characters. Optional.
* Code List Directory Version Number	Previously defined version number of a code list to implement in this message. Valid value is numeric code with six limited standard characters. Optional.
* Message Type Sub-Function Identification	Coded identifier of a sub-function of a message type. Valid value is six standard characters. Optional.
* Common Access Reference	Key used for referencing related all subsequent transfers of data to the same business case or file. Optional.
* Sequence Of Transfers	Number assigned by the sender indicating the transfer sequence of a message related to the same topic. The message could be an addition or a change to an earlier transfer related to the same topic. Valid value is two standard characters. Optional.
* First And Last Transfer	First and last message in a sequence of messages related to the same topic. Valid value is one standard character. Optional.
* Message Subset Identification	Coded identifier of a message subset by its identifier, version, release, and source. Valid value is 15 standard characters. Optional.
* Message Subset Version Number	Version number of the message subset. Valid value is three standard characters. Optional.
* Message Subset Release Number	Release number of the message subset. Valid value is three standard characters. Optional.
Message Subset Controlling Agency, Coded	Agency controlling the code representing the name of the message subset. Valid value is alphanumeric code with three standard characters. Optional.
* Message Implementation Guideline Identification	Coded identifier of a message implementation guideline by its identifier, version, release, and source. Valid value is 14 standard characters. Optional.
* Message Implementation Guideline Version Number	Version number of a message implementation guideline. Valid value is three standard characters. Optional.
* Message Implementation Guideline Release Number	Release number of a message implementation guideline. Valid value is three standard characters. Optional.

Field	Description
*Message Implementation Controlling Agency, Coded	Agency controlling the code representing the name of the message implementer. Valid value is alphanumeric code with three standard characters. Optional.
* Scenario Identification	Coded identifier of a message by its identifier, version, release, and source. Valid value is 14 standard characters. Optional.
* Scenario Version Number	Version number of a message scenario. Valid value is three standard characters. Optional.
* Scenario Release Number	Release number of a message scenario. Valid value is three standard characters. Optional.
Message Scenario Controlling Agency, Coded	Agency controlling the code representing the message scenario. Valid value is alphanumeric code with three standard characters. Optional.
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 document with the service it requires. Valid value must be at least one limited standard character. Required. **Note:** For CONTRL documents, the ALA format must be entered in the format as defined by the Inbound UNB/UNZ envelope for the **Accepter Lookup Alias for generated CONTRLs** parameter.
Limit Interchange Size	Maximum size allowed for an interchange before a single interchange is split into multiple interchanges that are each under the maximum size limitation. Valid values are: ◆ No ◆ Specify size limit ◆ Use transaction set default in enveloping.properties (default) Required. **Note:** If you select **Use transaction set default in enveloping.properties**, you need to specify the size limit directly (in the **Maximum Interchange Size** parameter) or by setting properties in the enveloping.properties property file, which contains a property for each supported standard and more specific properties for each message type that can override the standard property.
Maximum Interchange Size	Specify the maximum size of each individual interchange. Required. **Note:** Only displayed if you selected **Specify size limit** for the **Limit Interchange Size** parameter. If you choose to specify a maximum value, the value you specify is checked to make sure it is at least 173 bytes based on the header and trailer size for EDIFACT standard. **Note:** For performance reasons, the size is estimated for some components of the interchange (for example, control numbers) when determining the size. For this reason, the actual enforced size limit is slightly smaller than the value you specify.
Specify Non-Standard Delimiters	Whether you want to specify non-standard delimiters. Valid values are Yes and No (default). Required.
Data Element Separator	EDI delimiter to use to separate elements. Optional.

Field	Description
Segment Terminator	EDI delimiter to use to indicate the end of segments. Optional.
Component Element Separator	EDI delimiter used to separate components. Optional.
Release Character	Character to use to release an EDI delimiter. Optional.
Repetition Separator	Character to use to separate a group of repeated elements. Optional.
Decimal Separator	Character to use to identify different multi details. Optional.
Stream Segments	Insert an end-of-line character after each segment record. Valid values are Yes (stream segments) and No (default, do not stream segments). Required.
Primary Name Format	Information to include when generating a name for a global number and finding the correct number to assign based on that name. Optional. Information includes: ◆ Message Type ◆ Message Version and Release Number ◆ Test Indicator Check each that you want to use in the format.
First Backup Name Format	Check boxes to indicate what information to include when generating a name for a global number and finding the correct number to assign based on that name. Optional. Information includes: ◆ Message type ◆ Message version and release number ◆ Test Indicator Check each that you want to use in the format.
Second Backup Name Format	Check boxes to indicate what information to include when generating a name for a global number and finding the correct number to assign based on that name. Optional. Information includes: ◆ Message type ◆ Message version and release number ◆ Test Indicator Check each that you want to use in the format.
Global Message Reference Number	Message control reference number used globally. Select a previously defined number to reference. Valid value is nine standard characters. Required.
Local Message Reference Number	Message control reference number used locally. Select a previously defined number to reference. Valid value is nine standard characters. Required.
Map Name Mode	Mode in which to specify the map to use to perform a compliance check. Valid values are: ◆ Specify ◆ Generate from data
Map Name	Select the map to be used to perform a compliance check. The map must already be checked in. Optional.

Field	Description
Primary Name Format	Information to include when generating a name for a map and finding the correct number to assign based on that name. Information may include some or all of the following: ♦ EDI Standard ♦ Sender ID Code Qualifier ♦ Sender ID ♦ Recipient ID Code Qualifier ♦ Recipient ID ♦ Message Type ♦ Message Version and Release Number ♦ Test Indicator Check all that you want to use in the format. Optional.
First Backup Name Format	Information to include when generating a name for a map and finding the correct number to assign based on that name. Information may include some or all of the following: ♦ EDI Standard ♦ Sender ID Code Qualifier ♦ Sender ID ♦ Recipient ID Code Qualifier ♦ Recipient ID ♦ Message Type ♦ Message Version and Release Number ♦ Test Indicator Check all that you want to use in the format. Optional.
Second Backup Name Format	Information to include when generating a name for a map and finding the correct number to assign based on that name. Information may include some or all of the following: ♦ EDI Standard ♦ Sender ID Code Qualifier ♦ Sender ID ♦ Recipient ID Code Qualifier ♦ Recipient ID ♦ Message Type ♦ Message Version and Release Number ♦ Test Indicator Check all that you want to use in the format. Optional.

Field	Description
Generate an error if no generated map name exists in the system	Generate an error message if it cannot locate the specified map name. Valid values are Yes and No. Required.
Validate translation input	Whether to validate the translation input. Valid values are Yes (default) or No. Required.
Validate translation output	Whether to validate the translation output. Valid values are Yes (default) or No. Required.
Use Security	Use previously checked in trusted certificates to verify a secured document transmission. Valid values are Yes and No (default). Required. **Note:** This option *must* be set to Yes when using secure AUTACK messages in EDIFACT.
Scope of Security Application	Security header and message body referenced in the digital certificate. Required. Valid values are: ◆ 1 - Security header and message body (default) ◆ 3 - Whole related message, package, group, or interchange **Note:** Only displayed if Use Security is set to Yes.
Security Certificate	Select a previously checked in trusted certificate to associate with this envelope. Required. **Note:** Only displayed if Use Security is set to Yes.
Security Algorithm	Strength of the algorithm used to encrypt data and provides a key length your trading partner can support. Required. Valid values are: ◆ MD5 with RSA – Message digest algorithm that verifies the integrity of data through a 128-bit message digest. Default. ◆ SHA1 with RSA – Secure hash algorithm that verifies the integrity of data through a 160-bit (20 byte) message digest. Transmission may be a bit slower that MD5; however, the larger message digest size offers a more secure data transference. **Note:** Only displayed if Use Security is set to Yes.
Use of Algorithm	Defines when the algorithm is used. Required. Valid values are: ◆ 1 - Owner Hashing (default) ◆ 6 - Owner Signing **Note:** Only displayed if Use Security is set to Yes.
Security Service	Whether the message sender must sign the message with a digital certificate. Required. **Note:** Only displayed if Use Security is set to Yes.
Response Type	Whether you required an acknowledgement for transmitted digital certificates. Required. **Note:** Only displayed if Use Security is set to Yes.
Filter Function	The function of the filter. Default is Hexadecimal filter. Required.

Creating EDIFACT Maps

Overview

The Map Editor enables you to map EDIFACT documents, and generates a file layout for you using the components and message types that you select. The Map Editor-generated EDIFACT map consists of groups, records, composites, and fields that are comparable to parameters that are defined by EDIFACT.

**Note:** You need to download the Map Editor component to use it.

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 EDIFACT Standards Release messages loaded into the standards database through the Map Editor.

EDIFACT Components in the Map Editor

The following table lists the components that make up the EDIFACT layout in the Map Editor, the icons that represent the components, and descriptions of the components. For information about adding a map component to a layout, see *Map Editor Basics*.

Component	Icon	Description
EDIFACT root element		The *EDIFACT root element* represents the message that the application is mapping. At the EDIFACT file root element, you define the message type and encoding. It is a group and can contain groups and records.
Group		A *group* is a looping structure that contains a sequence or an implicit group of repeating field tags (in Map Editor a group is related segments 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.

Component	Icon	Description
Segment		An *EDIFACT segment* contains a field tag (in Map Editor, an EDIFACT segment is a group of related elements or composite data elements that combine to communicate useful data). An EDIFACT segment can occur once or can repeat multiple times. **Note:** If an EDIFACT segment 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.
Element		An *element* is a field or a group of EDIFACT components (the smallest piece of information defined by the standard) that define an EDIFACT element. An element can have different meanings depending on the context. In other data formats in the Map Editor, an element is not considered to have useful meaning except in the larger context of the segment that contains it. However, elements used in the Map Editor to represent EDIFACT fields and components contain useful and discrete information. **Note:** If an element 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 element* is an element with the ability to loop (occur more than once) within a particular EDIFACT segment. To enable a single element to repeat multiple times within an EDIFACT element, the occurrence of the element must be separated by a start and end delimiter. The use of start and end delimiters help the translator determine where elements and components are defined within a field tag. When a field has a link performed against it, a red check mark appears over the element icon. When a field contains an extended rule or a standard rule, an asterisk appears to the right of the element icon.

Using Transaction XREF with EDIFACT

If you need the ability to cross-reference the application data to generated transaction, perform the following:

1. The Map Editor uses the **TransactionCrossReferenceTable**, with possible values SenderID, ReceiverID, MessageType, Identifier.

2. The translator populates these values during translation.

3. After each round of translation completes, the EDI Enveloping service can query for the values. When it performs the enveloping, it will locate the standard-specific values for the same four parameters.

4. Based on these two sets of values for identifiers (application specific and standard specific), the EDI Enveloping service populates a TransactionCrossReferenceTable

5. You can build a custom application that queries this table to uniquely link an application file and generate an EDIFACT transaction.

Creating an EDIFACT Map

When you create a new map, you can either manually create an EDI layout or you can use a wizard that creates a layout for you based on an EDI standard. The wizard saves you time and effort and minimizes the risk of having an invalid standard format.

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).

To create an EDIFACT map:

1. From the 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 Integrator** is selected in the **What kind of map are you creating** list.

3. If you are translating from EDIFACT, in the Input Format window select the **Create a New Data Format Using this Syntax** option.

4. Then, select **Delimited EDI** and click **Customize**. If you are translating from another format, select that format and continue to the next page of the wizard.

5. If you want to use a HIPAA transaction, select the **HIPAA transaction** check box and click **Next**. Otherwise, just click **Next**.

6. If you want to import code lists from the database, select the **Import code list** check box and click **Next**. Otherwise, just click **Next**.

Note: If you select the **Import code list** check box, a Use Code standard rule is created on fields for which the element name matches a code list table name.

7. Select the ODBC data source that contains the EDI standards database (the default is **Sterling Integrator Standards**) and click **Next**.

8. Select the standards agency (**[E] EDIFACT**), the version of the standard, and the transaction set you want to use and click **Next**.

9. Click **Finish** to load the transaction set.

10. Click **Next** to select the Output Format.

11. If you are translating to EDIFACT, in the Output Format window (select the **Create a New Data Format Using this Syntax** option.

12. Then, select **Delimited EDI** and click **Customize**. If you are translating from another format, select that format and continue to the next page of the wizard.

13. If you want to use a HIPAA transaction, select the **HIPAA transaction** check box and click **Next**. Otherwise, just click **Next**.

14. If you want to import code lists from the database, select the **Import code list** check box and click **Next**. Otherwise, just click **Next**.

Note: If you select the **Import code list** check box, a Use Code standard rule is created on fields for which the element name matches a code list table name.

15. Select the ODBC data source that contains the EDI standards database (the default is **Sterling Integrator Standards**) and click **Next**.

16. Select the standards agency (**[E] EDIFACT**), the version of the standard, and the transaction set you want to use and click **Next**.

17. Click **Finish** to load the transaction set. The Map Editor displays the new map in the Map Editor window.

18. In the Map Editor, select **File > Save** to save the map. Do not use spaces or apostrophes in the map name.

Note: To compile the map, select **File > Compile**. A progress dialog box displays and updates during the compilation process. If the map contains a large number of objects, you may be prompted that you should save the map in .MAP format.

19. To save a map as a .map file, select File > Save As and then select Source Maps (*.map) from the Save as type list.

Note: Prior to opening an .mxl (XML-formatted) file, the Map Editor verifies that you have the Microsoft XML Core Services (MSXML) 4.0 installed on the same computer as Map Editor. If you do not have the Microsoft XML Core Services (MSXML) 4.0 installed, the Map Editor cannot save or load .mxl source files.

EDIFACT Reporting

The EDI Translation Detail report enables you to access EDI data (for a particular standard like EDIFACT) for a specified time period. It further allows you to organize the report by selecting criteria such as direction of the transmission, sender and receiver identifier, envelope name, unique identifier for the envelope level, EDI standard used, and translation success or failure, among other parameters. This report allows you to quickly and easily access information about EDI translation through the system.

To run a report containing details on inbound/outbound EDIFACT messages:

1. From the **Operations** menu, select **Reports**.

2. Use the Report Configuration wizard to create a new **EDI Transaction Detail** report.

3. On the **Detail Filtering** page in the **Standard** parameter, clear the checkboxes for all standards *except* **EDIFACT**.

4. Filter by any other desired criteria and click **Finish** at the end of the Report Configuration Wizard to generate the report.

Searching for EDIFACT 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 application has some predefined correlations and enables you to define more so that you can easily and efficiently monitor and track your activities, such as EDIFACT message flow.

To search for EDIFACT correlations:

1. From the **Administration** menu, select **Business Process** > **Monitor** > **Advanced Search** > **EDI Correlation**.

2. In the Search Option area, specify any combination of the following search criteria, as appropriate. To search for EDIFACT messages, in the **Interchange Level Options Standard** box, type **EDIFACT**.

3. Click **Go!** to display the EDI correlation records that match your search criteria.

4. In the EDI Correlation Interchange Results page, click ▢ **info** in the Detail column for the AS2 interchange for which you want to view details.

5. In the EDI Correlation Interchange/Group/Transaction Detail Results page, click ▢ **info** to the right of Document Correlations for additional information about the EDIFACT messages.