

IBM Sterling Gentran:Server for Windows



Application Integration Tutorial for Inbound Purchase Orders

Version 5.3.1

IBM Sterling Gentran:Server for Windows



Application Integration Tutorial for Inbound Purchase Orders

Version 5.3.1

Note

Before using this information and the product it supports, read the information in "Notices" on page 91.

This edition applies to the 5.3.1 version of IBM Sterling Gentrans:Server for Microsoft Windows and to all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright IBM Corporation 1996, 2012.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Chapter 1. Getting Started 1

About this Tutorial	1
Mapping Overview	2
Application File Format.	3
EDI File Format	4
Default Delimiter Values	5
The Application Integration Subsystem	5
Customizing Global Display Options	7
Customizing Global Display of Links	7
Customizing Global Confirmation Options	8
The Application Integration Map-building Process.	8
How to Use the Tutorial	9
Tips and Recommendations	10

Chapter 2. Inbound Purchase Order . . . 15

About the Inbound Purchase Order Tutorial	15
Getting Started on your Inbound Purchase Order.	15
Analysis of the Application File Format for your Inbound PO	16
Application File Definitions for a Purchase Order.	17
Application File Header Layout for a Purchase Order	17
Application File Detail Layout for a Purchase Order	18
Application File Summary Layout for a Purchase Order	20
Analysis of the Customer EDI File for a Purchase Order	20
Inbound PO Map Components for ANSI X12	21
Inbound PO Map Components for EDIFACT	21
Inbound PO Map Components for TRADACOMS	22
Reconcile your Inbound PO Application File and the Customer EDI File	23
Inbound PO Header Mapping Operations	24
Inbound PO Detail Mapping Operations.	25
Cross-reference Tables	26

Chapter 3. The Export Map 29

Creating the Export Map	29
Loading the ORDHDR Message - TRADACOMS only	31
Activating EDI Map Components	31
Verifying EDI Delimiters for Inbound POs	32
Application File Format	33
About the Application File Format in Inbound POs	33
Creating the POHEADER Record	33
Creating the PODetailGroup.	34
PODetailGroup Subordinate Records	34
Creating the POSUMMARY Record	36
The Application Side of the Map	36
Application Fields	37
About Application Fields in Inbound POs	37
Creating the POHEADER Record Fields	37
Creating the PODETAIL Record Fields	38

Creating the PODESC Record Fields	39
Creating the POALLCHG Record Fields - ANSI X12 and EDIFACT	41
Creating the POSUMMARY Record Fields	42

Chapter 4. POHEADER Record 45

About the POHEADER Record	45
Mapping the CUSTNUMHDR Field	45
PODATEHDR Field	46
Changing the Properties of the DTM02 Element	47
Creating the TEMPDATES Temporary Storage Segment	47
Creating the PODATE Temporary Storage Element	47
Using an Extended Rule to Extract the Date/Time.	48
Mapping the PODATEHDR Field	48
PONUMHDR Field.	49
Mapping the PONUMHDR	49
Changing the Maximum Length of the BGM05 Element - EDIFACT	49
Setting up the Document Name for the Inbound PO	49
Mapping the POTYPECODE Field	50

Chapter 5. Shipping and Billing 51

Shipping and Billing Information for ANSI X12 and EDIFACT	51
About Shipping and Billing Information for ANSI X12 and EDIFACT	51
Temporary Storage Segments for Shipping and Billing	51
Temporary Storage Elements for Shipping and Billing	52
Changing the Maximum Use of the N3 Segment - ANSI X12	53
Changing the Max Length of NAD20 - EDIFACT	54
Creating an Extended Rule to Extract Shipping And Billing Information	54
Mapping Shipping Information for ANSI X12 and EDIFACT	55
Date Segments and Elements for ANSI X12 and EDIFACT	56
Temporary Storage for Date Segments and Elements	56
Creating a Temporary Storage Date Segment	57
Creating a Temporary Storage Date Element	57
Using an Extended Rule to Extract Delivery and Cancel by Dates	58
Mapping the Delivery Date Information.	58
NUMLINEITEMS	59
Shipping and Billing Information for TRADACOMS	60
About Shipping and Billing Information for TRADACOMS	60

Mapping the Shipping Information for TRADACOMS	60	DESC Field	77
Mapping Customer Information for TRADACOMS	61	POALLCHG Record for ANSI X12 and EDIFACT.	78
Mapping the DELDATE Field for TRADACOMS	61	About the POALLCHG Record	78
Mapping the NUMLINEITEMS Field for TRADACOMS	62	Mapping the CUSTNUMAC Field	78
Chapter 6. PODetailGroup Records	63	Mapping the PODATEAC Field	79
PODETAIL Record	63	Mapping the PONUMAC Field	80
About the PODETAIL Record	63	ACIND Field	80
Mapping the CUSTNUMDET Field	63	Mapping the ACMETHAND Field.	82
Mapping the PODATEDET Field	64	ACTOTAMT Field	82
Mapping the PONUMDET Field	65	Chapter 7. POSUMMARY Record	85
Mapping the CUSTPROCEDURE/UPCCODE Fields	66	About the POSUMMARY Record	85
CONSUMERUNITS Field—TRADACOMS	67	Mapping the CUSTNUMSUM Field	85
UOM Field	69	Mapping the PODATESUM Field	86
UNITPR Field	70	Mapping the PONUMSUM Field	87
QTYORD Field	72	Mapping the NUMLINEITEMSUM Field	87
PODESC Record.	74	Chapter 8. Finalize the Map	89
About the PODESC Record	74	Compiling the Map for the Inbound PO.	89
Mapping the CUSTNUMDESC Field	74	Printing the Inbound PO Report	90
Mapping the PODATEDESC Field	75	Testing the Inbound PO Translation Object	90
Mapping the PONUMDESC Field	76	Notices	91

Chapter 1. Getting Started

About this Tutorial

This tutorial for the Sterling Gentran:Server® Application Integration subsystem references the ANSI X12, EDIFACT, and TRADACOMS standards.

Any sections in this tutorial that do not apply to all three standards are marked with the applicable standard name. If a section heading does not include a standard name, then the section applies to all three standards. In some tasks, the steps and information apply to all three standards but a specific detail may be different, such as a field name.

For example: In the "What is the name of the form" box, enter one of the following:

- **PET_850** (for ANSI X12)
- **PET_EORD** (for EDIFACT)
- **PET_TORD** (for TRADACOMS)

This tutorial includes an inbound mapping example (purchase order) and an outbound mapping example (invoice). It covers EDI to flat file translation only. The purpose of this tutorial is to break the mapping process into concise and understandable steps, using examples that allow you to understand and master the tasks and processes required to create your own maps.

This guide assumes that you are familiar with using a PC and with Microsoft Windows functions, including:

- The terminology used to describe mouse and cursor actions.
- The terminology used to describe Windows-specific attributes, including dialog boxes, icons, windows, and buttons.
- Using the Windows Program Manager.

This guide also assumes that you are familiar with EDI concepts, EDI standard structure, your internal application format, data mapping concepts, and with the Sterling Gentran:Server product.

Additional Information

To access reference information, see the *IBM® Sterling Gentran:Server for Microsoft Windows Application Integration User Guide*.

For additional information on the various tasks described in this guide, refer to the *IBM Sterling Gentran:Server for Microsoft Windows User Guide*.

For XML mapping information, see the tutorial in the *IBM Sterling Gentran:Server for Microsoft Windows XML User Guide*.

Mapping Overview

A map is a set of instructions you define in the Sterling Gentran:Server Application Integration subsystem that defines the relationship between your application file and the EDI standards, and defines how the system should translate data.

For inbound mapping, the system translates your trading partner's EDI standard formatted business document to your file format. To translate inbound data, you need to create an export map in Sterling Gentran:Server Application Integration (because the system is exporting to your application file).

For outbound mapping, the system translates your file format to EDI standard formats, so you can send documents to your partners. To translate outbound data, you need to create an import map and a system import map in the Sterling Gentran:Server Application Integration subsystem. You need to create an import map because the system is importing from your application file.

The import map defines how to move data from your application file (flat file definition), which may contain multiple documents, to the EDI standard-formatted documents that your partners expect to receive from you. You need to create a system import map to determine which trading relationship (established in Partner Editor) corresponds to each document in the application file, so the system knows which import map to use to process the document.

Translation Objects

The map you create is the source from which you compile a translation object. A translation object is a set of instructions the translator uses to convert a file from one format to another. Every function performed during translation requires a translation object. Translation objects control all the functions of Sterling Gentran:Server EDI translation. Translation objects can belong to one of the following categories:

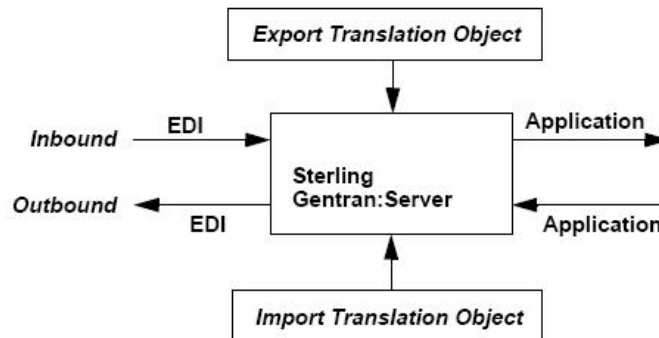
- system translation objects
- user translation objects

System translation objects control the creation and separation of interchanges, functional groups, and transaction sets. They are also used to generate and reconcile functional acknowledgements. User translation objects control data entry, importing, exporting, document turnaround, and creating printed reports. All required system translation objects are automatically installed with the Sterling Gentran:Server system.

The Application Integration subsystem enables you to generate import, export, and turnaround translation objects. Once you generate a translation object using the Application Integration subsystem, you must register the translation object with Sterling Gentran:Server. In addition, you must set up relationships for all trading partners from which you receive or to which you send data. During the process of establishing a trading relationship, you need to specify the name of the appropriate translation object that the translator uses to translate the data. See the *IBM Sterling Gentran:Server for Microsoft Windows User Guide* for more information about trading partners.

Inbound and Outbound Translation Process

The following diagram shows a high-level view of the inbound EDI to application and outbound application to EDI translation processes.






When you create an export map, you need to define to Sterling Gentran:Server the EDI standard format in which your partners sends documents and your application file format (how the data needs to be formatted for your application to process it). When you create an import map, you must define to Sterling Gentran:Server your application file format (your data that you will send to your partners) and the EDI standard format in which your partners expect to receive the documents.

Application File Format

If you are creating an import or export map, you must define your application to the Application Integration subsystem. In Sterling Gentran:Server terminology, your application file is also referred to as a fixed-format file or a positional file. Your application file must contain all the information that you need to either extract from your partner's document (if the map is inbound) or send to your partner (if the map is outbound).

This table describes the map components that you use to define your application file.

Component	Icon	Description
Group		This is a looping structure that contains related records and/or groups that repeat in sequence until either the group data ends or the maximum number of times that the loop is allowed to repeat is exhausted. If you create a group that is subordinate to another group (a subgroup), this corresponds to a nested looping structure (a loop within a loop). The application (positional) file is a group and therefore, it is visually represented the same way as other groups and subgroups in the Application Integration subsystem.
Record		Contains a group of related fields. A record can occur once or can repeat multiple times.
Field		This is the smallest piece of information defined in the application file. A field is the application map component that is mapped (linked) to a corresponding EDI element. When an element contains a standard rule a black asterisk appears to the right of the element icon.

Notes:

- When a field has a mapping operation performed against it, a red checkmark is displayed over the field icon.
- When a field contains a standard or extended rule a black asterisk appears to the right of the element icon.
- When a group contains an extended rule, a yellow asterisk appears to the right of the group icon.

Before you define your application file format, you should obtain a layout of the necessary records, fields, and groups. Each map component is arranged sequentially in the order that it is most logical for the system to process. Therefore, each level of your application file must be created sequentially. For example, your application file contains records and groups. The records contain fields and the groups contain records and/or subgroups. This means that you must create records and groups before you create the subordinate fields.

EDI File Format

The EDI file must contain all the information that you expect to receive from your partner (if the map is inbound) or need to send to your partner (if the map is outbound).

The Application Integration subsystem generates an EDI file for you, based on the standard (agency), version, transaction set, and release (for TRADACOMS only) that you selected. The system includes all the groups, segments, composites, and elements that are defined by the standards agency for the version of the document you selected. If you are creating an import or export map, you typically need to customize the system-generated EDI file by modifying the properties of the map components and using specialized Sterling Gentran:Server functions to manipulate the EDI file structure.

The specific EDI map components that you use depends on the type of map you are creating. This includes the standard, version, and transaction set (document) selected, and which groups, segments, composites, and elements your company requires. We recommend that you determine which map components you are using before generating or defining an EDI file.

The system activates all of the groups, segments, composites, and elements that are defined as mandatory by the standard. The system does not enable you to deactivate the mandatory groups, segments, composites, and elements. By default, Sterling Gentran:Server displays active map components with a black font, and inactive map components with a grey (dimmed) font.

When translating data, the system does not process groups, segments, composites, and elements (or records and fields) that are not activated. Therefore, you must activate the groups, segments, composites, and elements that are not defined as mandatory by the standard, but that you have determined that you need to use in mapping.

If you want to use a specialized version of an EDI standard that is not available in the Sterling Gentran:Server standards database, it may be appropriate for you to define the EDI file yourself.

This table describes the map components Sterling Gentran:Server uses to define the EDI file.

Component	Icon	Description
Group		This is a looping structure that contains related segments and/or groups that repeat in sequence until either the group data ends or the maximum number of times that the loop is allowed to repeat is exhausted. Groups are defined by the EDI standards. A group that is subordinate to another group is a subgroup (this corresponds to a nested looping structure – a loop within a loop). The EDI file is a group and is visually represented the same way as other groups and subgroups in the Application Integration subsystem.
Segment		Contains a group of related elements or composite data elements that combine to communicate useful data. Segments are defined by the EDI standards. A segment can occur once or can repeat multiple times.
Composite		This is a data element that contains two or more component data elements or subelements. Composites are defined by the EDI standards that use them (EDIFACT, TRADACOMS, and certain ANSI X12 standards).
Element		This is the smallest piece of information defined by the EDI standards. An element is the EDI map component that is mapped (linked) to a corresponding application field to move data to and from the EDI file.

Notes:

- When an element has a mapping operation performed against it, a red checkmark is displayed over the element icon.
- When a field contains a standard or extended rule a black asterisk appears to the right of the element icon.
- When a group contains an extended rule, a yellow asterisk appears to the right of the group icon.

Default Delimiter Values

The default delimiter values for all three standards are shown in the following table:

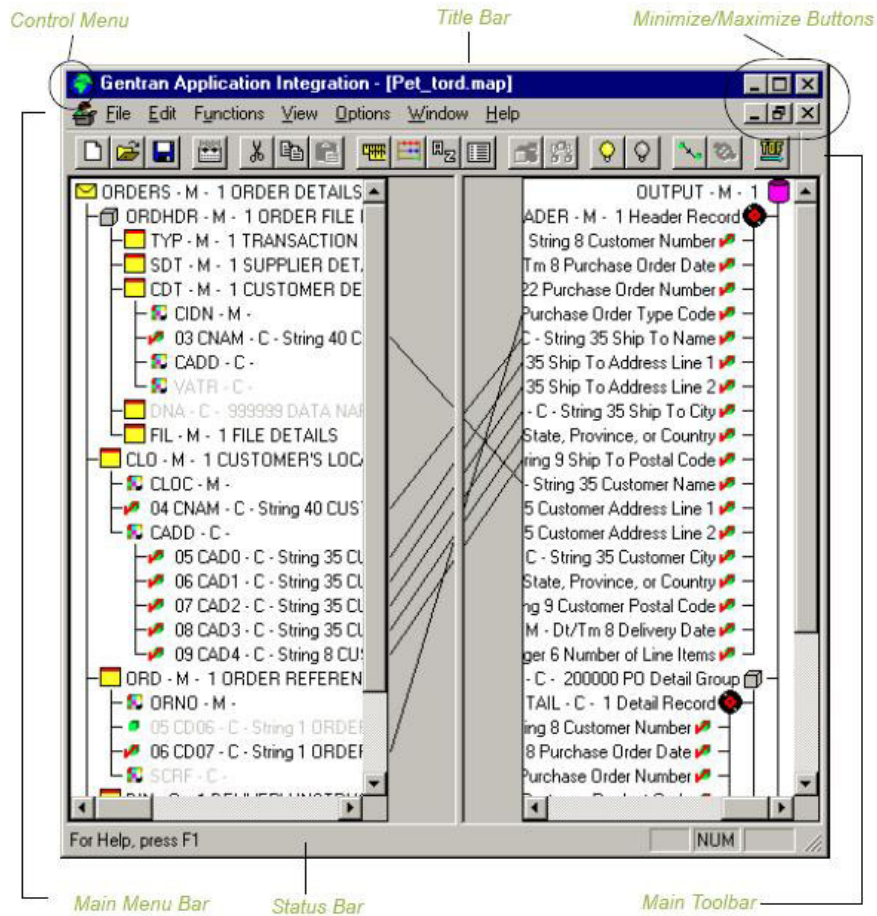
Box	ANSI X12 Default Char.	ANSI X12 Default Hex	EDIFACT Default Char.	EDIFACT Default Hex	Tradacoms Default Char.	Tradacoms Default Hex
Tag Delimiter					= (equal sign)	0x3D
Element Delimiter	* (asterisk)	0x2A	+ (plus)	0x2B	+ (plus sign)	0x2B
Segment Delimiter	~ (tilde)	0x7E	' (single quote)	0x27	' (single quote)	0x27
Sub Element	: (colon)	0x3A	: (colon)	0x3A	: (colon)	0x3A
Release Character			? (question mark)	0x3F	? (question mark)	0x3F
Decimal Separator			. (period)	0x2E	. (full stop)	0x2E

The Application Integration Subsystem

The Application Integration subsystem enables you to translate your application files to EDI standard formats for documents you send to your partners (outbound mapping) and to translate EDI standard formats to your application format for documents that you receive from your partners (inbound mapping).

Application Integration Window

The following diagram illustrates the Sterling Gentran:Server - Application Integration Window. (This is a TRADACOMS example.)



This table describes the Application Integration window components.

Part	Function
Control Menu	Allows you to control the Application Integration window by moving, maximizing, or closing it.
Title Bar	Contains the name of the application. You can use this to move the position of the EDI Director window on your desktop.
Minimize	Reduces the window to an icon.
Maximize	Enlarges the window to its maximum size.
Main Menu Bar	Contains drop-down menus. Unavailable items are dimmed.
Main Toolbar	Enables you access to some of the most common operations in the Application Integration subsystem. Unavailable items are dimmed. The Main Toolbar is a dockable toolbar, so you can affix it to any edge of the client window.
Status Bar	Displays status information about a selection, command, or process; defines menu items as you highlight each item in the menu; and indicates any current keyboard-initiated mode for typing (for example, CAP for the Caps Lock key).

Customizing Global Display Options

The Sterling Gentran:Server Application Integration subsystem allows you to customize the display of maps in several different ways. The global display options that are established apply to all maps.

About this task

Note: You can set or change global map display options at any time.

Use this procedure to customize global display options for all maps.

Procedure

1. Select **Options > Preferences**. The Preferences dialog box enables you to set global defaults for Sterling Gentran:Server.
The Preferences dialog box is displayed.
2. To turn on the default display of group, record (segment), and field (element) descriptions, select the appropriate options.

Notes:

- You typically want all descriptions displayed for reference. But depending on the size of your monitor, it may be easier to see the entire map if the descriptions are not displayed.
 - You may want to experiment with shrinking the size of the font for the map before you turn off the display of descriptions.
3. Click **OK** to save changes and exit the Preferences dialog box.

Customizing Global Display of Links

Mapping Links are the visual lines that connect a field/element on the Input side of the map to a field/element on the Output side of the map.

About this task

Use this procedure to customize the global display of mapping links.

Procedure

1. Select **Options > Preferences**.
The Preferences dialog box is displayed. The Tree tab is displayed by default.
2. Click the **Links** tab. The Links tab enables you to set the default display of mapping links.
3. Select one of the following settings:
 - **Show no links** - Do not display mapping links.
 - **Show links to or from the currently selected element** - Display only the mapping links for the currently selected field.
 - **Show links to or from all visible elements** - Display all the mapping links.
We recommend that you select this option.
The option you select is the default for all maps.
4. Click **OK** to save changes and exit the Preferences dialog box.

Customizing Global Confirmation Options

The Confirmations tab on the Preferences dialog allows you specify when you want confirmation messages displayed. Depending on how you set these options, you may receive either more or fewer confirmation messages than described in this tutorial.

About this task

Use this procedure to set the confirmation options.

Procedure

1. Select **Options > Preferences**.
The system displays the Preferences dialog box (Tree tab displayed by default).
2. Select the **Confirmation** tab.
The system displays the confirmation options.
3. Set the global confirmation options by either selecting the Confirm everything check box (displays all confirmation messages) or by selecting individual confirmation messages by action performed.
4. Click **OK** to save the confirmation options.
The system sets the confirmation options you selected.

The Application Integration Map-building Process

The following table provides an overview of the map-building process.

Stage	Description
1	Prepare and analyze. Obtain a layout of your application file and determine how it corresponds with the EDI standard you are using. Determine how you move data to or from each application field.
2	Set global defaults (first time only). The first time you use the Application Integration subsystem, you should establish the default date format that the system uses. Please see the <i>IBM Sterling Gentran:Server for Microsoft Windows User Guide</i> for more information on setting the global date default.
3	Create, save, and name a new map. See the topics on creating maps for more information. <ul style="list-style-type: none">• For outbound invoices, create both a system import map and an import map.• For inbound purchase orders, create an export map.
4	Activate the appropriate EDI groups, segments, and elements. See the topic on activating EDI map components for more information.
5	Define your application file format. See the topic on analyzing the application file format for more information.
6	Map the appropriate data for each application field. Specific mapping tasks begin with mapping or defining the header of the translation object.

Stage	Description
7	Print the mapping report. Validate and review the map, and make modifications as needed. See the topic on printing the report for more information.
8	Compile the translation object. See the topic on compiling the map for more information.
9	Register the translation object with Sterling Gentran:Server. See the <i>IBM Sterling Gentran:Server for Microsoft Windows User Guide</i> for more information on registering translation objects.
10	Create the appropriate trading relationship. Establish the appropriate trading relationship in Sterling Gentran:Server for your trading partners. The trading relationship that you will use in this tutorial is provided in the tutorial folders. You need to import the partner profile into Sterling Gentran:Server, before you begin this tutorial. See the <i>IBM Sterling Gentran:Server for Microsoft Windows User Guide</i> for more information on trading partners.
11	Testing the translation object. Obtain test data from your partners and process the data. Verify acknowledgement processing (if applicable). Verify communications with your network.

How to Use the Tutorial

Tutorial basics for ANSI X12

For these tutorials, we have provided test data for the translation objects you create. The following table describes how to locate that test data:

Translation Object Type	Test Data Name	Default Path Location
Inbound (export)	PETTEST.INT	C:\GENSRVNT\TUTORIAL\PETTEST.INT
Outbound (system import and import)	PET_810.TXT	C:\GENSRVNT\TUTORIAL\PET_810.TXT

Tutorial basics for EDIFACT

For these tutorials, we have provided test data for the translation objects you create. The following table describes how to locate that test data:

Translation Object Type	Test Data Name	Default Path Location
Inbound (export)	PETTSTE1.INT	C:\GENSRVNT\TUTORIAL\PETTESTE1.INT
Outbound (system import and import)	PET_EINV.TXT	C:\GENSRVNT\TUTORIAL\PET_EINV.TXT

Tutorial basics for TRADACOMS

For these tutorials, we have provided test data for the translation objects you create. The following table describes how to locate that test data:

Translation Object Type	Test Data Name	Default Path Location
Inbound (export)	PETTSTT1.INT	C:\GENSRVNT\TUTORIAL\PETTSTT1.INT
Outbound (system import and import)	PET_TINV.TXT	C:\GENSRVNT\TUTORIAL\PET_TINV.TXT

Tips and Recommendations

The following information includes tips and recommendations that you can use while you are going through the tutorials and building your maps. You might want to print these pages to keep available through the process.

Map names

You allocate a file name to a map when you save the map for the first time or use Save As to save a copy of the map under a different name. The name should define the map in a meaningful way. The default filename extension is .MAP (it must have a MAP extension to open in the Map Editor). Using this extension distinguishes map files from other types of files, such as translation objects. The naming conventions in Sterling Gentran:Server include:

The source map (map created in Sterling Gentran:Server) should be named using the file extension .MAP (for example, PET_810.MAP, PET_EINV.MAP, or PET_TINV.MAP).

The compiled translation object (translation object that is the result of compiling the source map) should be named using the same filename that was allocated to the source map, but with the file extension .TPL (for example, PET_810.TPL, PET_EINV.TPL, or PET_TINV.TPL).

We recommend that you use these naming conventions to distinguish source maps and compiled translation objects from other types of files. We also recommend that you use the same file name (different file extension) for a source map and the translation object that was compiled from it.

Be careful not to overlay the source map with the compiled translation object. Use the .TPL file extension to distinguish the translation object.

Document names and references

We recommend that you set up a Document Name for each map to make the identification of a document created by this map easier in Sterling Gentran:Server. This allows you to differentiate between documents in the document browsers in Sterling Gentran:Server. Document names can be up to 255 characters in length.

To set up a Document Name, you must select an element or field in the map that contains data that distinguishes the documents translated by this map. (If you choose a non-string element or field, you need to convert it to a string first.) Then, the Name field of the document browsers that contain this document contain the data from the selected element or field. You can choose any element or field as the Document Name, but we strongly recommend that you select an element or field that occurs only once in the document and has a data type of String. In this tutorial, the document name for the map is the purchase order number.

You can also establish a Document Reference for each map, which also assists you in differentiating between documents. When you set up the Reference in the Application Integration subsystem, the RefData field of the document browsers that contain this document in Sterling Gentran:Server contains the data from that mapped field. In this tutorial, the document reference for the import map in this tutorial is the purchase order number.

Link function

The Link function allows you to map a field or element from the Input side of the map to a field or element on the Output side of the map. When you click the element, the cursor changes to a Link Arrow. The link between the two fields is visually represented with a line connecting the two fields.

You can only map fields and elements that are at the same level. This means that you cannot map to a group from a segment that does not repeat.

Important: If you select **Confirm when I link options at different levels**, you will get a warning when an element should not be linked. Do not ignore this warning. If you link elements when you see this warning, your output may be incorrect.

When compiling

Each rule in the map is compiled when you compile the translation object after you complete the map. However, the system allows you to compile each rule individually so that you can verify its accuracy after you create it.

The Compile function will compile the rule interactively and allows you to correct any generated errors. Errors or warnings that occur during the compilation process are displayed in the Compile Errors box (or in the Errors list).

Entries in dialog boxes

Names

Do not use spaces or dashes (-) in the Name box. You can use the underscore (_) to separate words.

Each application field must have a unique name. It is useful to tag the end of the fields that occur in multiple records with a suffix that identifies the record that contains it. For example, HDR for the Header record, DET for the detail record, DESC for the description record, AC for the allowance/charge record, SUM for the summary record.

Dates:

- For ANSI X12 and EDIFACT, dates are in the MMDDYY format. For example, 120511 would be December 5, 2011.
- For Tradacoms, dates are in the DD/MM/YY format. For example, 05/12/11 would be 5 December 2011.

Descriptions:

This box is used to provide a brief explanation and allows you to differentiate it from similar fields.

Tags:

A record is recognized by the system as:

```
<TAG>[Field_1][Field_2]
```

The record TAG for each record enables the system to recognize that record and then determine the mapping requirements.

For temporary storage, use a value of \$\$\$\$. The system does not read a segment with a value of \$\$\$ in the Tag box. Therefore, it does not flag this temporary storage segment as an error during compliance checking.

We recommend that you define the record tag on the Positional Record Properties dialog box, instead of defining fields with the purpose of explicitly containing the record tag.

Looping Tab:

The Looping section allows you to specify the minimum and maximum number of times an element can repeat, and whether it is mandatory or conditional. Alter this section only if the element is a repeating element. Otherwise, accept the defaults.

Minimum Usage (Looping):

This is the minimum number of times the loop must be repeated. For a conditional loop, the minimum usage should always be 0. For a mandatory loop, the minimum usage should be 1 or greater.

Position tab:

To indicate the exact position of the field in the record, type the starting position of the field in the Start Pos box. For example, you should specify field start positions if you are only using a few fields but you want them positioned exactly in the record.

In this tutorial, add the fields sequentially in the record and then use the Auto Position function.

Auto Position:

This automatically calculates the start position in the record of each field based on the criteria that each field is positioned directly after the previous field and is of the length specified in the Max Length box.

It is only valid to use the Auto Position function if you define a record tag on the Positional Record Properties dialog box, and if you define every field in the record in the sequence that each field occurs, as you are doing in this tutorial.

Positional Field Editor:

If you press **Enter** after typing the information for the last field in the record, a blank field is created. Click **Delete** and then **Yes** to delete the blank field.

Accumulators

There is only one set of accumulators for each map. This means that accumulator 0, whether it is used in the Primary Accumulator or Alternate Accum field is the same accumulator with the same contents. If you assign calculations to accumulator 0 at the beginning of the map and then use accumulator 0 again later in the map, the content of that accumulator will be the result of the earlier calculation. Any additional calculations you assign to that accumulator will be performed on the contents resulting from an earlier calculation.

Chapter 2. Inbound Purchase Order

About the Inbound Purchase Order Tutorial

This part of the tutorial defines the step-by-step tasks that are necessary to create a map that will be used to translate purchase orders that you receive from your trading partner.

These tasks include the following:

- Mapping analysis
- Creating the new map
- Activating groups, segments, and elements
- Activating EDI map components
- Verifying EDI Delimiters
- Defining your application file format
- Mapping your data
- Finalizing the map

There are tasks and information for ANSI X12, EDIFACT, and TRADACOMS in this tutorial. Some tasks and information may not apply to all three standards, and the heading will indicate this. If the heading does not specify which standard it applies to, then it applies to all three.

Getting Started on your Inbound Purchase Order

This tutorial is meant to be completed in its entirety. We recommend that you complete each task sequentially without referring to the completed map that is provided with the installation program. You can then verify your completed map with the model we have provided.

Completing each task in this tutorial in sequential order as if it were your own scenario not only teaches individual mapping tasks, but also provides a logical approach to methods that you use when you create your own map.

Inbound Mapping Scenario

This tutorial sets up the following scenario:

- Your company is MWT Manufacturing Co., which manufactures a wide variety of supplies for pet stores.
- MWT Manufacturing Co. sells pet supplies directly to large pet supply retail chains.
- Pet Zone, a large pet supply retailer, is the trading partner sending a purchase order to your company.

You need to create a map that enables your company to translate the purchase orders that you receive from Pet Zone. Once the purchase orders are translated into your application file format, they can be processed through your Order Processing System, and your company can ship the goods to Pet Zone.

You have an existing application file layout from your order entry department that defines the information your system needs to process the order. And, from discussions with the Pet Zone, you know the data content of the orders that your partner is sending you.

About Mapping Analysis

The first step in creating a map is to analyze your mapping requirements. This is the most important step in creating a successful map. If your analysis is complete, you have all the information you need to create the map in an efficient and logical manner. If you omit this critical step and proceed directly to creating the map, you may have problems creating your map and the map may be invalid due to oversights and omissions.

This table describes the steps used in mapping analysis for translating both inbound and outbound data.

Step	Action
1	Analyze your application file format.
2	Analyze your partner's EDI file format.
3	Correlate your application and the EDI file formats.
4	Create the map.

Analysis of the Application File Format for your Inbound PO

The first step of mapping analysis is analyzing your application file format because this is probably the component that is the most familiar to you. In this tutorial, your application file format contains all the information that you need to extract from the purchase order your partner sends you, so that your system can correctly process the purchase order, and your company can fill the order correctly.

You must define your application file format to the Application Integration subsystem. If your company has an existing application file format, as in this tutorial, you should obtain the record layout from the appropriate person. If you do not have an existing application file format, you need to create one by determining which fields are necessary to process the data correctly, and then grouping the fields logically under records.

Purchase orders are divided into three parts:

- Header (POHEADER)
- Detail information (PODetailGroup)
- Summary (POSUMMARY)

Header record:

The POHEADER contains the purchase order type code, shipping and billing information, delivery and cancel by date information, and the number of line items in the purchase order. The header record is mandatory and occurs once for each purchase order translated.

Detail information group:

The PODetailGroup contains all the line item information for the order, including the item, unit of measure, price, quantity ordered, description, and the allowance and charge information.

In addition, the detail information is split up into records:

- Detail (PODETAIL)△Contains line item code number, unit of measure, price, and quantity ordered. The PODETAIL record occurs once per PODetailGroup.
- Description (PODESC)△Contains descriptive information about the line item ordered. The PODESC record can occur up to five times per PODetailGroup, so multiple descriptive terms can be used for each item. For example, if the line item is a dog collar, the description fields may be red, large, and leather.
- Allowance and charge (POALLCHG)—for ANSI X12 and EDIFACT△Contains the allowance or charge indicator, method of handling code, and total allowance or charge amount. The POALLCHG record can occur up to 10 times (for ANSI X12) or 5 times (for EDIFACT) per PODetailGroup, so multiple allowances and/or charges can be used for each item.

Summary record:

The POSUMMARY contains the total number of line items in the purchase order. This record occurs once for each purchase order translated.

Application File Definitions for a Purchase Order

The tables in the following topics contain the minimum information necessary to define your application file format to the Application Integration subsystem. Each record contains a group of logically-related application fields. These fields contain the information from the EDI file that your system needs to process the purchase order. The group contains logically-related application records.

Each record contains a key (consisting of three fields) that ensures that the same identifying information is located at the beginning of each record. Your application contains a record key so that each record can be identified individually. The record key for this purchase order consists of:

- your partner’s customer number
- the purchase order date
- purchase order number

Application File Header Layout for a Purchase Order

The application file header layout for this tutorial is shown in the table below. This occurs once per document.

Field Name	Description	Data Type	Length (min - max)
CUSTNUMHDR	Customer Number (mandatory)	String	1 - 8
PODATEHDR	Purchase Order Date (mandatory)	Dt/Tm MMDDYY or DD/MM/YY	6 or 8

Field Name	Description	Data Type	Length (min - max)
PONUMHDR	Purchase Order Number (mandatory)	String	1 - 22
POTYPECODE	Purchase Order Type Code	String	0 - 2
SHIPTONAME	Ship To Name	String	0 -35
SHIPTOADD1	Ship To Address Line 1	String	0 - 35
SHIPTOADD2	Ship To Address Line 2	String	0 - 35
SHIPTOCITY	Ship To City	String	0 - 35 0 - 30 (for ANSI)
SHIPTOSTATE	X12 = Ship To State ED/TC = Ship To State, Province, or Country	String	0 - 2 0 - 35 (for TRADACOMS)
SHIPTOPOSTALCODE	Ship To Postal Code	String	0 - 9
BILLTONAME	Bill To Name	String	0 - 35
CUSTNAME	Customer Name		
BILLTOADD1	Bill To Address Line 1	String	0 - 35
CUSTADD1	Customer Address Line 1		
BILLTOADD2	Bill To Address Line 2	String	0 - 35
CUSTADD2	Customer Address Line 2		
BILLTOCITY	Bill To City	String	0 - 2
CUSTCITY	Customer City		0 - 35 (for TRADACOMS)
BILLTOSTATE	Bill To State	String	0 - 2
CUSTSTATE	Customer State, Province, or Country		0 - 35 (for TRADACOMS)
BILLTOPOSTALCODE	Bill To Postal Code	String	0 - 9
CUSTPOSTALCODE	Customer Postal Code		
DELDATE	Delivery Date (mandatory)	Dt/Tm MMDDYY or DD/MM/YY	6 or 8
CANCELDATE (ANSI X12 and EDIFACT)	Cancel By Date (mandatory)	Dt/Tm (MMDDYY)	6
NUMLINEITEMS	Number of Line Items (mandatory)	Number (N0)	1 - 6

Application File Detail Layout for a Purchase Order

Each PODetailGroup can occur up to:

- ANSI X12: 100,000 times per document
- EDIFACT: 200,000 times per message

- TRADACOMS: 999,999 times per message

The application file **detail record** layout for this tutorial is shown in the table below. This occurs once per PODetailGroup.

Field Name	Description	Data Type	Length (min - max)
CUSTNUMDET	Customer Number (mandatory)	String	1 - 8
PODATEDET	Purchase Order Date (mandatory)	Dt/Tm MMDDYY or DD/MM/YY	6 or 8
PONUMDET	Purchase Order Number (mandatory)	String	1 - 22
CUSTPROCEDURE	Customer Product Code (mandatory)	String	1 - 30
UPCCODE	UPC Code (mandatory)	String	1 - 30
CONSUMERUNITS TRADACOMS only	Consumer Units in Trading Unit	Number (N0)	0 - 15
UOM	Unit of Measure	String	0 - 7
UNITPR	Unit Price	X12/ED: Number (N2)△ TC: Number (R2)	0 - 14
QTYORD	Quantity Ordered (mandatory)	Number (N0)	1 - 9

The application file **description record** layout for this tutorial is shown in the table below. This occurs up to five times per PODetailGroup.

Field Name	Description	Data Type	Length (min - max)
CUSTNUMDESC	Customer Number (mandatory)	String	1 - 8
PODATEDESC	Purchase Order Date (mandatory)	Dt/Tm MMDDYY or DD/MM/YY	6 or 8
PONUMDESC	Purchase Order Number (mandatory)	String	1 - 22
DESC	Description	String	0 - 80 0 - 81 (for TRADACOMS)

The application file **allowance/charge record** layout for this tutorial is shown in the table below. This occurs up to 10 times per PODetailGroup for ANSI X12 documents or up to 5 times per PODetailGroup for EDIFACT messages. It does not apply to TRADACOMS.

Field Name	Description	Data Type	Length (min - max)
CUSTNUMAC	Customer Number (mandatory)	String	1 - 8
PODATEAC	Purchase Order Date (mandatory)	Dt/Tm (MMDDYY)	6
PONUMAC	Purchase Order Number (mandatory)	String	1 - 22
ACIND	Allowance or Charge Indicator	String	0 - 1
ACMETHAND	Allowance or Charge Method of Handling Code	String	0 - 2
ACTOTAMT	Allowance or Charge Total Amount	Number (N2)	0 - 9

Application File Summary Layout for a Purchase Order

The application file summary record layout for this tutorial is shown in the table below. This occurs once per document.

Field Name	Description	Data Type	Length (min - max)
CUSTNUMSUM	Customer Number (mandatory)	String	1 - 9
PODATESUM	Purchase Order Date (mandatory)	Dt/Tm MMDDYY or DD/MM/YY	6 or 8
PONUMSUM	Purchase Order Number (mandatory)	String	1 - 22
NUMLINEITEMSUM	Number of Line Items (mandatory)	Number (N0)	1 - 6

Analysis of the Customer EDI File for a Purchase Order

To analyze the EDI file, you must first determine what your trading partners are sending you. You and your partners need to agree on which standard, version, and transaction set or message that you are using.

In this tutorial, we are using:

- ANSI X12 standard, version 003030, transaction set 850 (purchase order)
- EDIFACT standard, version 92.1, message ORDERS (purchase order message)

- TRADACOMS standard, version ANA001, message ORDERS (purchase order message), release 9

It is important to know the information that your partners are sending you within the purchase order, and what data is contained in each element used. It is also helpful to review the EDI standards that you are using before analyzing the EDI file. Obtain an EDI standards manual for each standard and version that you are using. Standards manuals are available from EDI standard agencies.

The TRADACOMS standard uses a header with each interchange (ORDHDR for the ORDERS message). The header message contains information that pertains to each message in the interchange. Sterling Gentran:Server copies the header segments to the beginning of each document, to ensure that all the relevant information is contained in the document. Therefore, you will need to load the header message (ORDHDR) into the ORDERS EDI file, so that all the segments and elements are available for use.

After discussing with your partners what they are sending, determining which segments and elements your company requires, and reviewing the EDI standard, you can list the map components that you need to make available for use (activate).

Note: When you activate a conditional group or segment, the mandatory subordinate segments and elements are automatically activated.

Inbound PO Map Components for ANSI X12

This table lists the ANSI X12 map components that are required for this tutorial.

Groups	Segment	Element
	DTM	DTM02
N1	N1 N3 N4	<ul style="list-style-type: none"> • N102 • N302 • N401 • N402 • N403
PO1 (already activated)	PO1	<ul style="list-style-type: none"> • PO102 • PO103 • PO104 • PO107
	PID	PID:205
	ITA:2	ITA:207

Inbound PO Map Components for EDIFACT

This table lists the EDIFACT map components that are required for this tutorial. The Composite column is provided so you will know where to locate the appropriate element.

Groups	Segment	Group	Segment	Composite	Element
	BGM segment (already active)				<ul style="list-style-type: none"> • BGM05 element • BGM06 element

Groups	Segment	Group	Segment	Composite	Element
	DTM segment (already active)			C507	DTM02 element
NAD△group	NAD segment			C080 C059 C059	<ul style="list-style-type: none"> • NAD10 element • NAD16 element • NAD17 element • NAD19 element • NAD20 element • NAD21 element
LIN group	LIN segment			C212	LIN03 element
	IMD:2 segment			C273	IMD:206 element
	QTY:3 segment			C186	QTY:303 element
	MOA:5 segment			C516	MOA:502 element
	ALC group	ALC:2 segment			ALC:204 element
		MOA group	MOA:8 segment	C516	MOA:802 element
	CNT segment			C270	

Inbound PO Map Components for TRADACOMS

The majority of the elements you activate will be mapped to your application file. However, there are three elements (SDT SIDN SID0, CDT CIDN CID0, and OLD:2 SPRO SPR1) that you activate because your partner will send you data to be compliant with the TRADACOMS standard. You do not need this information in your application file; therefore, you will not need to map these three elements.

This table lists the TRADACOMS map components that are required for this tutorial. The composite column is informational only, so you will know where to locate the appropriate element.

Group	Segment	Composite	Element
ORDHDR (already active)	SDT (already active)	SIDN	SID0
	CDT (already active)	CIDN	CID0
			CNAM
		CADD	<ul style="list-style-type: none"> • CAD0 • CAD1 • CAD2 • CAD3 • CAD4

Group	Segment	Composite	Element
ORDERS	CLO (already active)	CLOC	CLO0
			CNAM
		CADD	<ul style="list-style-type: none"> • CAD0 • CAD1 • CAD2 • CAD3 • CAD4
	ORD (already active)	ORNO (already active)	<ul style="list-style-type: none"> • ORN0 • ORN2
			CD07
	DIN		LDAT
OLD (already active)	OLD:2	SPRO	SPR1
		CPRO	CPR1
		UNOR	<ul style="list-style-type: none"> • UNO0 • CD04
		OQTY	OQT0
		OUCT	OUC0
		TDES	<ul style="list-style-type: none"> • TDE0 • TDE1
ORDTRLR	MHD		
	OFT		
	MTR		

Reconcile your Inbound PO Application File and the Customer EDI File

To reconcile your application file format with the EDI file, you must identify each application field with its corresponding element in the EDI file and select a method for mapping it. To map information to a field, you use linking, standard rules, extended rules, or a combination of all three.

- Linking (simple mapping) enables you to map a field or element from the input side of the map to a field or element on the output side of the map. The link between two map components (fields) is visually represented with a line connecting the two fields.
- Standard rules give you access to mapping operation functions that are more complex than simple linking, but less involved than extended rules.
- Extended rules enable you to use a Sterling Gentran:Server proprietary programming language to perform virtually any mapping operation you require.

Additionally, you may determine that you need to establish temporary storage (work) areas for the map to handle items such as Ship To and Bill To name and address information, which is extracted using extended rules from a group in the EDI data.

We recommend that you determine which mapping operations are required on a field-by-field basis for your application file.

Inbound PO Header Mapping Operations

This table describes the header mapping operations (by application field) required for this tutorial. This occurs once per document.

Field Name	Mapping Operations for ANSI X12	Mapping Operations for EDIFACT	Mapping Operations for TRADACOMS
CUSTNUMHDR	Extract Application ID from Partner record	Extract Application ID from Partner record	Extract Application ID from Partner record
PODATEHDR	Link from BEG05	Link from DTM02 using 4 (PODATE) code	Link from ORD ORNO ORN2
PONUMHDR	Link from BEG03	Link from BGM05	Link from ORD ORNO ORN0
POTYPCODE	Link from BEG02	Link from BGM06	Link from ORD CD07
SHIPTONAME	Link from N101 using ST (ShipTo) Code	Link from NAD10 using ST (ShipTo) code	Link from CLO CNAM
SHIPTOADD1	Link from N301 using ST (ShipTo) Code	Link from NAD16 using ST (ShipTo) code	Link from CLO CADD CAD0
SHIPTOADD2	Link from N302 using ST (ShipTo) Code	Link from NAD17 using ST (ShipTo) code	Link from CLO CADD CAD1
SHIPTOCITY	Link from N401 using ST (ShipTo) Code	Link from NAD19 using ST (ShipTo) code	Link from CLO CADD CAD2
SHIPTOSTATE	Link from N402 using ST (ShipTo) Code	Link from NAD20 using ST (ShipTo) code	Link from CLO CADD CAD3
SHIPTOPOSTALCODE	Link from N403 using ST (ShipTo) Code	Link from NAD21 using ST (ShipTo) code	Link from CLO CADD CAD4
BILLTONAME CUSTNAME	Link from N101 using BT(BillTo) Code	Link from NAD10 using BT (BillTo) code	Link from CDT CNAM
BILLTOADD1 CUSTADD1	Link from N301 using BT(BillTo) Code	Link from NAD16 using BT (BillTo) code	Link from CDT CADD CAD0
BILLTOADD2 CUSTADD2	Link from N302 using BT(BillTo) Code	Link from NAD17 using BT (BillTo) code	Link from CDT CADD CAD1
BILLTOCITY CUSTCITY	Link from N401 using BT(BillTo) Code	Link from NAD19 using BT (BillTo) code	Link from CDT CADD CAD2
BILLTOSTATE CUSTSTATE	Link from N402 using BT(BillTo) Code	Link from NAD20 using BT (BillTo) code	Link from CDT CADD CAD3
BILLTOPOSTALCODE CUSTPOSTALCODE	Link from N403 using BT(BillTo) Code	Link from NAD21 using BT (BillTo) code	Link from CDT CADD CAD4
DELDATE	Link from DTM02 using 001 Code	Link from DTM02 using 117 (DELDATE) code	Link from DIN LDAT
CANCELDATE	Link from DTM02 using 002 Code	Link from DTM02 using 153 (CANCELDATE) code	Not applicable.
NUMLINEITEMS	Link from CTT01	Link from CNT02 using 2 (NUMLINEITEMS) code	Link from OTR LORD

Inbound PO Detail Mapping Operations

Each PODetailGroup can occur up to:

- ANSI X12: 100,000 per document
- EDIFACT: 200,000 per message
- TRADACOMS: 999,999 per message

This table describes the **detail record** mapping operations (by application field) required for this tutorial. This occurs once per PODetailGroup.

Field Name	Mapping Operations for ANSI X12	Mapping Operations for EDIFACT	Mapping Operations for TRADACOMS
CUSTNUMDET	Extract Application ID from Partner record	Extract Application ID from Partner record	Extract Application ID from Partner record
PODATEDET	Extended rule to map from BEG05	Link from DTM02 using 4 (PODATE) code	Link from ORD ORNO ORN2 using extended rule
PONUMDET	Extended rule to map from BEG03	Link from BGM05	Link from ORD ORNO ORN0 using extended rule
CUSTPROCEDURE	Link from P0107	Link from LIN03	Link from OLD:2 CPRO CPR1
UPCCODE	Extract Partner xref by partner item from subtable PRODCODE	Extract Partner xref by partner item from subtable PRODCODE	Extract Partner xref by partner item from subtable PRODCODE
CONSUMERUNITS	Not applicable.	Not applicable.	Link from OLD:2 UNOR UNO0
UOM	Link from P0103	Extended rule to link from QTY:303	Link from OLD:2 UNOR CD04
UNITPR	Link from P0104	Link to MOA:502 using 146 (UNITPRICE) code	Link from OLD:2 OUCT OUC0
QTYORD	Link from P0102	Link from QTY:302 using 21 (QTYORD) code	Link from OLD:2 OQTY OQT0

This table describes the **description record** mapping operations (by application field) required for this tutorial. This occurs up to five times per PODetailGroup.

Field Name	Mapping Operations for ANSI X12	Mapping Operations for EDIFACT	Mapping Operations for TRADACOMS
CUSTNUMDESC	Extract Application ID from Partner record	Extract Application ID from Partner record	Extract Application ID from Partner record
PODATEDESC	Extended rule to map from BEG05	Link from DTM02 using 4 (PODATE) code	Link from ORD ORNO ORN2 using extended rule
PONUMDESC	Extended rule to map from BEG03	Link from BGM05	Link from ORD ORNO ORN0 using extended rule
DESC	Link from PID:205	Link from IMD:206	Link from OLD:2 TDES TDE0 and OLD:2 TDES TDE1 after concatenating them with an extended rule

This table describes the **allowance/charge record** mapping operations (by application field) required for this tutorial. This occurs up to 10 times per

PODetailGroup for ANSI X12 documents or up to 5 times per PODetailGroup for EDIFACT messages. It does not apply to TRADACOMS.

Field Name	Mapping Operations for ANSI X12	Mapping Operations for EDIFACT	Mapping Operations for TRADACOMS
CUSTNUMAC	Extract Application ID from Partner record	Extract Application ID from Partner record	Not applicable.
PODATEAC	Extended rule to map from BEG05	Link from DTM02 using 4 (PODATE) code	Not applicable.
PONUMAC	Extended rule to map from BEG03	Link from BGM05	Not applicable.
ACIND	Link from ITA:201	Link from ALC:201	Not applicable.
ACMETHAND	Link from ITA:204	Link from ALC:204	Not applicable.
ACTOTAMT	Link from ITA:207	Link from MOA:802	Not applicable.

Summary mapping operations

This table describes the **summary** mapping operations (by application field) required for this tutorial. This occurs once per document.

Field Name	Mapping Operations for ANSI X12	Mapping Operations for EDIFACT	Mapping Operations for TRADACOMS
CUSTNUMSUM	Extract Application ID from Partner record	Extract Application ID from Partner record	Extract Application ID from Partner record
PODATESUM	Link from BEG05	Link from DTM02 using 4 (PODATE) code	Link from ORD ORNO ORN2
PONUMSUM	Link from BEG03	Link from BGM05	Link from ORD ORNO ORN0
NUMLINEITEMSUM	Link from CTT01	Link from CNT02 using 2 (NUMLINEITEMS) code	Link from OTR LORD

Cross-reference Tables

You must determine if you need code tables to convert map data. If you need code tables, you must decide whether to create a code table in Sterling Gentran:Server that can only be used with this map or to add the table to either the partner relationship or the internal system partner in Partner Editor.

In the Inbound Purchase Order tutorial, you need to map information from a cross-reference table to translate your partner's code for a purchased item to your code for the same item. A cross-reference table is used when you and your partner each reference an item by a different name (or number).

In the Outbound Invoices tutorial, you need to extract information from a lookup table to be inserted into the outbound data. A lookup table is used when you want to supplement data before it is sent to your partner.

The cross-reference table below is included in the tutorial data with the PETZONE partner (INVATCO.LKP).

My Value	Partner Value	Description	Reference Data
12345678901234	555666411	Large Dog Collar	Red
23456789012345	555666817	Spring Breeze Cat Litter	10 lb or 5 kg Bag
34567890123456	555666908	Small Beef Basted Rawhide Bones	Bag of 12
45678901234567	555666321	X-Large Food Bowl	Black
56789012345678	555666287	FatKat Kitty Bed	Blue with White Trim
67890123456789	555666413	Catnip Mouse Toy	

Chapter 3. The Export Map

Creating the Export Map

The first task in creating a new map is to define the format used for the input and output sides of the map.

About this task

In this example, you are creating an export map so Sterling Gentran:Server can translate a purchase order that your partner sends to your application file. The input side of the map must reflect the purchase order that you will receive. Instead of defining the purchase order to the system yourself, you want Sterling Gentran:Server to use the standards database on your system to build the purchase order according to the standard. The output side of the map must reflect your application file.

Use this procedure to create the export map for the Inbound Purchase Order.

Procedure

1. Select **File > New**.

The system displays the New Map Wizard.

2. Select **Export** as the type of map.
3. For the name of the form, enter one of the following:
 - **PET_850** (for ANSI X12)
 - **PET_EORD** (for EDIFACT)
 - **PET_TORD** (for TRADACOMS)

The system adds the .MAP extension.

4. Verify that your name appears correctly and click **Next**.

Note: You need to complete the format of the Input side of the map. This is the format of the data that is translated by the Sterling Gentran:Server system.

5. To create a new data format using this syntax list, select **Delimited EDI** and click **Customize**.
6. Click **Next**.
7. From the ODBC data source list, select **Gentran Standards** (if you defined your EDI standards data source with another name, select that name) and click **Next**.
8. For ANSI X12, select the following and click **Next**:
 - **[X] X12 standards agency**
 - **[003030] ANSI X12 VERSION 3 RELEASE 3 version**
 - **[850] PURCHASE ORDER transaction set**
9. For EDIFACT, select the following and click **Next**:
 - **[E] EDIFACT standards agency**
 - **[092001] EDIFACT 1992 RELEASE 1 (92.1) version**
 - **[ORDERS] PURCHASE ORDER MESSAGE transaction set**
10. For TRADACOMS, select the following and click **Next**:
 - **[A] TRADACOMS standards agency**

- [ANA001 TRADACOMS UNITED KINGDOM version
 - [ORDERS] ORDER DETAILS transaction set
 - 9 release
11. Click **Finish**.
The system loads the selected transaction set.
 12. Click **Next**.
 13. To create a new data format using this syntax list, select **Positional** and click **Next**.
 14. Click **Finish** to create the new map (this may take a few seconds).
The system displays the new map in the Application Integration Window.

Note: After you finish creating and saving a new map, you need to define the Input and Output sides of the map. The steps you take are different, depending on whether the map is an Import, System Import, Export, or Turnaround map.

15. Select **Edit > Details**.
The Translation Object Details dialog box should look similar to the following. (This is an ANSI X12 example; some details may be different for EDIFACT and TRADACOMS.)

Summary		
Author	Description	Translation Object Function
Sterling	PET_850	Export

Flags

System Use Configurable Trimming Gentran:Server for Windows 2.x Compatible Rule Execution

Version Control

Major version: 0 Minor version: 0 Compiled on: _____

EDI Associations

	Input	Output
Agency	X12	
Version	003030 ANSI X12 VERSION	
Transaction	850 PURCHASE ORDE	
Release	0	
F Group	PD	

16. In the Description box, enter one of the following:
 - **PET X 3030 850 Export** (for ANSI X12)
 - **PET E 92.1 ORDERS Export** (for EDIFACT)
 - **PET T ANA001 9 ORDERS Export** (for TRADACOMS)
17. Complete the Major and Minor boxes to designate a version number for this translation object. Type **1** as the version number in the Major box and accept the **0** as the release number in the Minor box.

Note: Whenever any major changes are made to your map, increase the version to the next real number (for example, 2.0, 3.0). If you make a minor change to the map, increase the version by a percentage point (for example, 2.1, 2.2).

18. Click **OK** to save the translation object details.
19. Select **File > Save As**.
20. Verify that the File name box contains one of the following:

- **pet_850.map** (for ANSI X12)
 - **pet_eord.map** (for EDIFACT)
 - **pet_tord.map** (for TRADACOMS)
21. Click **Save** to save the map.

Note: After you complete each section of this tutorial or when you make any changes to the map, you should save the map.

Loading the ORDHDR Message - TRADACOMS only

The TRADACOMS standard uses a header with each interchange (ORDHDR for the ORDERS message). The header message contains information that pertains to each message in the interchange. Sterling Gentran:Server copies the header segments to the beginning of each document, to ensure that all the relevant information is contained in the document. Therefore, you will need to load the header message (ORDHDR) into the ORDERS EDI file, so that all the segments and elements are available for use.

About this task

Use this procedure to load the ORDHDR message into the ORDERS EDI file.

Procedure

1. Right-click **ORDERS ORDER DETAILS EDI** and select **Create Sub > Transaction set from standards**.
The system displays the Select Data Source dialog box.
2. Select the **Machine Data Sources** tab.
3. Select **Gentran Standards** (if you defined your EDI standards data source with another name, select that name) and click **OK**.
The system displays the Insert Transaction Set from Standards dialog box.
4. From the Agency list, select **[A] TRADACOMS** as the EDI standard you will use for the inserted message.
5. From the Version list, select **[ANA001] TRADACOMS UNITED KINGDOM** as the version of the standard you will use. The versions that are available depend on which standard you selected.
6. From the Transaction list, select **[ORDHDR] ORDER FILE HEADER** as the transaction set (message) you will use. The transactions that are available depend on which standard you selected.
7. From the Release list, select **9** as the message version you will use. The message versions that are available depend on which message (transaction) you selected.
8. Click **OK** to load the ORDHDR message into the ORDERS EDI file as a subordinate group.

Activating EDI Map Components

About this task

Use this procedure to activate the required conditional groups, segments, and elements.

Procedure

1. Double-click the groups and segments to open them. Opening all the necessary segments and groups before you start to activate allows you to scroll down the map and activate the segments and elements sequentially. See the following for details:
 - Inbound PO Map Components for ANSI X12
 - Inbound PO Map Components for EDIFACT
 - Inbound PO Map Components for TRADACOMS

Note: When you activate a conditional group or segment, the mandatory subordinate segments, composites, and elements are automatically activated.

2. Click **Activate** on the Main Toolbar.
3. Click each inactive group that you need. This activates the groups.

Note: If you accidentally click a group, segment, or element that you did not mean to activate, right-click on the object and select **Deactivate**.

4. Click each inactive segment that you need. This activates the segments.
5. Click each inactive element that you need. This activates the elements.
6. Once you have activated all necessary groups, segments, and elements, click **Activate** on the Main Toolbar to turn activation mode off.
7. Select **View > Show Active Only**.

Verifying EDI Delimiters for Inbound POs

If you are using an EDI standard that contains composite elements or subelements, you must verify that Sterling Gentran:Server is specifying the correct EDI delimiters. Delimiters are flags that you define to the system as separating specific EDI components. Delimiters are necessary for all variable field-length standards because the data is compressed and the leading zeroes and trailing blanks are removed. Because the fields vary in length, the system needs a flag to determine where one element ends and another begins. For example, an element delimiter marks the beginning of a new element.

About this task

Although verifying EDI delimiters in Sterling Gentran:Server is mandatory only if you are using a standard with composite elements or sub-elements, we recommend that you perform this task regardless of which standard you use.

Use this procedure to verify EDI delimiters.

Procedure

1. Right-click one of the following and select **Properties**:
 - **850 PURCHASE ORDER EDI** (for ANSI X12)
 - **ORDERS ORDER DETAILS EDI** (for EDIFACT)
 - **ORDERS ORDER DETAILS EDI** (for TRADACOMS)The system displays the File Properties dialog box.
2. Select the **Delimiters** tab.
3. Verify that the Specify defaults check box is selected.

4. Verify the required delimiters for the standard. If the delimiters differ from the default in the table below, type either the character or the hexadecimal value in the correct box.
5. Click **OK** to exit the File Properties dialog box.

Application File Format

About the Application File Format in Inbound POs

If either side of your map is positional (fixed-format), you must define your application file format to Sterling Gentrans:Server.

In this tutorial, the output side of the map is positional. Your application file must contain all the information that you need to extract from your partner's document.

Before you define your application file format, obtain a layout of the necessary groups, records, and fields. The groups contain related records and the records contain related fields. Each map component is arranged sequentially in the order that it is most logical for the system to process. Therefore, each level of your application file format must be created sequentially. This means that you must create records and groups before you create the subordinate fields and records.

Creating the POHEADER Record

The first record in your application file is generally the header record. The first record is subordinate to the icon in the application side (output) of the map.

About this task

The POHEADER (header) record contains the purchase order type code, shipping and customer billing information, delivery and cancel by date information, and the number of line items in the purchase order. This header record is mandatory and occurs once for each purchase order translated.

We recommend that you include a header record in each application file. The header record is typically mandatory and repeats only once (not a looping structure).

Use this procedure to create the first application record (POHEADER).

Procedure

1. Right-click the Positional File icon (Output for an inbound map) and select **Create Sub > Record**.
The Positional Record Properties dialog box is displayed.
2. In the Name box, type **POHEADER** as the record identification code.
3. In the Desc box, type **PO Header Record** as the description of the record.
4. On the **Tag** tab, do the following:
 - In the Tag box, type **HDR** as the record identification code <TAG>.
 - In the Position box, accept the default of **1** as the starting column position of the tag in the data record.
5. On the **Looping** tab, type **1** in the Min Usage box to make the header record mandatory. This is the minimum number of times the record must repeat. If you want the record to repeat (loop), type the maximum number of times it can repeat in the Max Usage box.

Note: If the Min Usage box contains **0**, the record is conditional. If the Min Usage box contains **1** or greater, the record is mandatory.

We recommend that you always accept the default of 1 so it remains a record that only occurs once.

6. Click **OK** to create the POHEADER record.

Creating the PODetailGroup

The PODetailGroup (detail information) contains all line item information for the order, including the item, unit of measure, price, quantity ordered, description, and the allowance and charge information.

About this task

Each PODetailGroup can occur up to 100,000 times (for ANSI X12), 200,000 (for EDIFACT), or 999,999 times (for TRADACOMS) per order translated. The information within the PODetailGroup is further segregated by related records that repeat in sequence until either the data ends or it reaches the maximum number of times that the loop is allowed to repeat.

Use this procedure to create the PODetailGroup.

Procedure

1. Click the **POHEADER** and select **Edit > Insert > Group**.
The Group Properties dialog box is displayed.
2. In the Name box, type **PODetailGroup** to indicate that this group contains all of the detail records. This is the segment loop identification code.
3. In the Description box, type **PO Detail Group** as the loop description.
4. Select the **Looping** tab.
5. Accept the default of **0** in the Minimum Usage box.
6. In the Maximum Usage box, enter one of the following to indicate the maximum number of times the group of records can repeat.
 - **100000** (for ANSI X12)
 - **200,000** (for EDIFACT)
 - **999999** (for TRADACOMS)
7. Click **OK** to create the PODetailGroup.

PODetailGroup Subordinate Records

In this tutorial, the PODetailGroup contains the following subordinate records:

- **PODETAIL**
- **PODESC**
- **POALLCHG.** (for ANSI X12 and EDIFACT)

After you create the first subordinate record (PODETAIL), you can define the subsequent records at the same level (PODESC and POALLCHG).

Creating the PODETAIL Record

About this task

Use this procedure to create the PODETAIL record.

Procedure

1. Click the **PODetail** Group and select **Edit > Create Sub > Record**.
The Positional Record Properties dialog box is displayed.
2. In the Name box, type **PODETAIL** as the record identification code.
3. In the Desc box, type **PO Detail Record** as the description of the record.
4. On the **Tag** tab, do the following:
 - In the Tag box, type **DET** as the record identification code <TAG>.
 - In the Position box, accept the default of **1** as the starting column position of the tag in the data record.
5. Click **OK** to create the **PODETAIL** record.

Creating the PODESC Record About this task

Use this procedure to create the PODESC record.

Procedure

1. Click the **PODETAIL** record and select **Edit > Insert > Record**.
The system displays the Positional Record Properties dialog box.
2. In the Name box, type **PODESC** the record identification code.
3. In the Desc box, type **Description Record** as a description of the record.
4. On the **Tag** tab, do the following:
 - In the Tag box, type **DES** as the record identification code <TAG>.
 - In the Position box, accept the default of **1** as the starting column position of the tag in the data record.
5. On the **Looping** tab, do the following:
 - Accept the default of **0** in the Min Usage box.
 - In the Max Usage box, type **5** as the maximum amount of times the record can repeat.
6. Click **OK** to create the PODESC record.

Creating the POALLCHG Record - ANSI X12 and EDIFACT About this task

Use this procedure to create the POALLCHG record:

Procedure

1. Click the **PODESC** record and select **Edit > Insert > Record**.
The system displays the Positional Record Properties dialog box.
2. In the Name box, type **POALLCHG** the record identification code.
3. In the Desc box, type **Allowance/Charge Record** as a description of the record.
4. On the **Tag** tab, do the following:
 - In the Tag box, type **ACG** as the record identification code <TAG>.
 - In the Position box, accept the default of **1** as the starting column position of the tag in the data record.
5. On the **Looping** tab, do the following:
 - Accept the default of **0** in the Min Usage box.
 - In the Max Usage box, type **10** (for ANSI X12) or **5** (for EDIFACT) as the maximum amount of times the record can repeat.

- Click **OK** to create the POALLCHG record.

Creating the POSUMMARY Record

The final record you need to create (POSUMMARY) is at the same level as the POHEADER record and the PODetailGroup. The POSUMMARY (summary) record contains the total number of line items in the purchase order. The POSUMMARY record occurs once for each purchase order translated.

About this task

Use this procedure to create the POSUMMARY record.

Procedure

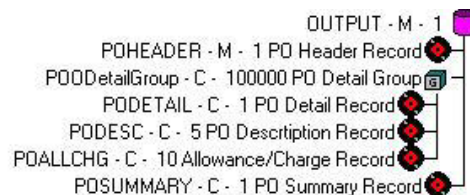
- Click the **PODetailGroup** and select **Edit > Insert > Record**.
The system displays the Positional Record Properties dialog box.

Note: Select Insert to create equal records/groups and select Create Sub to create subordinate groups/records/fields.
- In the Name box, type **POSUMMARY** as the record identification code.
- In the Desc box, type **PO Summary Record** as the description of the record.
- On the **Tag** tab, do the following:
 - In the Tag box, type **SUM** as the record identification code <TAG>.
 - In the Position box, accept the default of **1** as the starting column position of the tag in the data record.
- On the **Looping** tab, do the following:
 - Accept the default of **0** in the Min Usage box.
 - Accept the default of **1** in the Max Usage box so the summary record only occurs once (not a looping structure).
- Click **OK** to create the POSUMMARY record. See the results below.

The Application Side of the Map

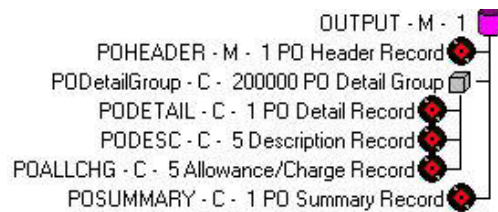
ANSI X12

The following example shows the ANSI X12 application (output) side of the map.



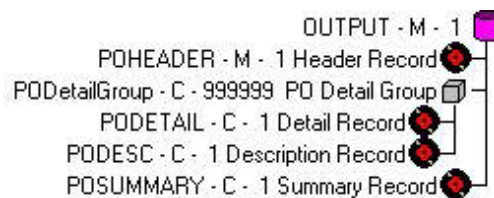
EDIFACT

The following example shows the EDIFACT application (output) side of the map.



TRADACOMS

The following example shows the TRADACOMS application (output) side of the map.



Application Fields

About Application Fields in Inbound POs

Each record you create contains a group of logically-related application fields. These fields define the structure and content of the data that your system needs to process the purchase order.

The easiest way to add application fields to a record is to use the Positional Field Editor. Generally, you create the fields for the first record in the application file (POHEADER), and then proceed with each sequential record.

See the following for details of the fields:

- Application File Header Layout for Inbound POs
- Application File Detail Layout for Inbound POs
- Application File Summary Layout for Inbound POs

Creating the POHEADER Record Fields

About this task

Use this procedure to create the application fields for the POHEADER record.

Procedure

1. Right-click the **POHEADER** application record and select **Edit Fields**.
2. If the field you are creating is the first field in the record, click **New**.
3. In the Name box, type **CUSTNUMHDR**, which is the name of the first field in the POHEADER record.

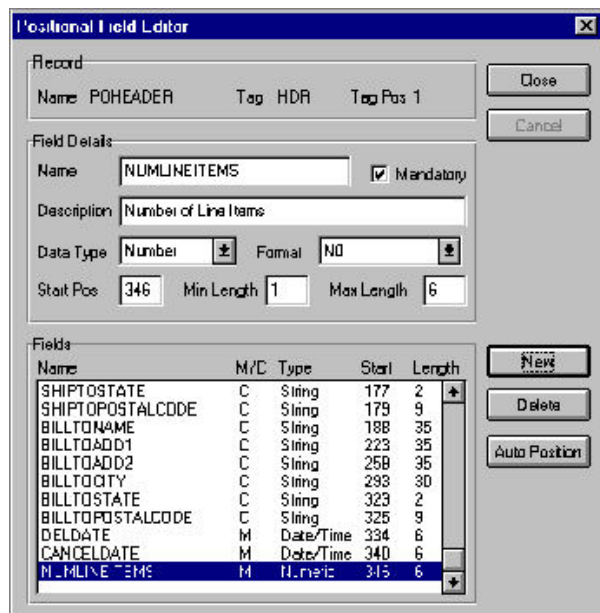
4. Select the **Mandatory** check box to identify it as a required field.
5. In the Description box, type **Customer Number**.
6. From the Data Type list, select the appropriate type.

Note: Select **String** if you want to specify that this field must contain alphanumeric data.

If the Data Type is Number or Dt/Tm, select the appropriate format from the Format list.

7. In the Min Length box, type **1** as the minimum field length. If the minimum length of the field is **0**, then accept that default in the Min Length box.
8. In the Max Length box, select the maximum length of the field from the list.
9. To add the field and to create the next field positioned after it, press **Enter**.
10. Create the rest of the fields according to Application File Header Layout for Inbound POs .
11. After adding the last field, click **Auto Position** to automatically position the fields in the record. Click **Yes** to acknowledge the warning message that fields are sequenced in order.

The Positional Field Editor dialog box should look similar to the following. (This is an ANSI X12 example; some details may be different for EDIFACT and TRADACOMS.) Only the last eleven fields in the record are currently visible in the Fields list:



12. When you are finished with the POHEADER record, click **Close**.

Creating the PODETAIL Record Fields

About this task

Use this procedure to create the application fields for the PODETAIL record.

Procedure

1. Right-click the **PODETAIL** application record and select **Edit Fields**.
2. If the field you are creating is the first field in the record, click **New**.

3. In the Name box, type **CUSTNUMDET**, which is the name of the first field in the PODETAIL record.
4. Select the **Mandatory** check box to identify it as a required field.
5. In the Description box, type **Customer Number**.
6. From the Data Type list, select the appropriate type.

Note: Select **String** if you want to specify that this field must contain alphanumeric data.

If the Data Type is **Number** or **Dt/Tm**, select the appropriate format from the Format list.

7. In the Min Length box, type **1** as the minimum field length. If the minimum length of the field is **0**, then accept that default in the Min Length box.
8. In the Max Length box, type **8** as the maximum length of the field.
9. To add the field and to create the next field positioned after it, press **Enter**.
10. Create the rest of the fields according to Application File Detail Layout for Inbound POs .
11. After adding the last field, click **Auto Position** to automatically position the fields in the record. Click **Yes** to acknowledge the warning message that fields are sequenced in order.

The Positional Field Editor dialog box should look similar to the following. (This is an ANSI X12 example; some details may be different for EDIFACT and TRADACOMS.)

The screenshot shows the 'Positional Field Editor' dialog box. At the top, it displays 'Record Name: PODETAIL', 'Tag: DET', and 'Tag Pos: 1'. The 'Field Details' section includes: Name: QTYORD, Description: Quantity Ordered, Data Type: Number, Format: NO, Start Pos: 121, Min Length: 1, and Max Length: 9. The 'Fields' section contains a table of existing fields:

Name	M/C	Type	Start	Length
CUSTNUMDET	M	String	4	8
PODATEDET	M	Date/Time	12	6
PONUMDET	M	String	18	22
CUSTPROCDE	M	String	40	30
UPCCODE	M	String	70	30
UDM	C	String	100	7
UNITPR	C	Numeric	107	14
QTYORD	M	Numeric	21	9

12. When you are finished with the PODETAIL record, click **Close**.

Creating the PODESC Record Fields

About this task

Use this procedure to create the application fields for the PODESC record.

Procedure

1. Right-click the **PODESC** application record and select **Edit Fields**.
2. If the field you are creating is the first field in the record, click **New**.
3. In the Name box, type **CUSTNUMDESC**, which is the name of the first field in the PODESC record.
4. Select the **Mandatory** check box to identify it as a required field.
5. In the Description box, type **Customer Number**.
6. From the Data Type list, select the appropriate type.

Note: Select **String** if you want to specify that this field must contain alphanumeric data.

If the Data Type is **Number** or **Dt/Tm**, select the appropriate format from the Format list.

7. In the Min Length box, type **1** as the minimum field length. If the minimum length of the field is **0**, then accept that default in the Min Length box.
8. In the Max Length box, type **8** as the maximum length of the field.
9. To add the field and to create the next field positioned after it, press **Enter**.
10. Create the rest of the fields according to Application File Detail Layout for Inbound POs .
11. After adding the last field, click **Auto Position** to automatically position the fields in the record. Click **Yes** to acknowledge the warning message that fields are sequenced in order.

The Positional Field Editor dialog box should look similar to the following. (This is an ANSI X12 example; some details may be different for EDIFACT and TRADACOMS.)

Positional Field Editor

Record
Name PODESC Tag DES Tag Pos 1

Field Details
Name: DESC Mandatory
Description: Description
Data Type: String Format: X
Start Pos: 40 Min Length: 0 Max Length: 80

Name	M/C	Type	Start	Length
CUSTNUMDESC	M	String	4	8
PODATEDESC	M	Date/Time	12	6
PDNUMDESC	M	String	18	22
DESC	M	String	40	81

12. When you are finished with the PODESC record, click **Close**.

Creating the POALLCHG Record Fields - ANSI X12 and EDIFACT

About this task

Use this procedure to create the application fields for the POALLCHG record.

Procedure

1. Right-click the **POALLCHG** application record and select **Edit Fields**.
2. If the field you are creating is the first field in the record, click **New**.
3. In the Name box, type **CUSTNUMAC**, which is the name of the first field in the POALLCHG record.
4. Select the **Mandatory** check box to identify it as a required field.
5. In the Description box, type **Customer Number**.
6. From the Data Type list, select the appropriate type.

Note: Select **String** if you want to specify that this field must contain alphanumeric data.

If the Data Type is **Number** or **Dt/Tm**, select the appropriate format from the Format list.

7. In the Min Length box, type **1** as the minimum field length. If the minimum length of the field is **0**, then accept the default in the Min Length box.
8. In the Max Length box, type **8** as the maximum length of the field.
9. To add the field and create the next field positioned after it, press **Enter**.
10. Create the rest of the fields according to Application File Detail Layout for Inbound POs .
11. After adding the last field, click **Auto Position** to automatically position the fields in the record. Click **Yes** to acknowledge the warning message that fields are sequenced in order.

The Positional Field Editor dialog box should look similar to the following. (This is an ANSI X12 example; some details may be different for EDIFACT and TRADACOMS.)

Name	M/C	Type	Start	Length
CUSTNUMAC	M	String	4	8
PODATEAC	M	Date/Time	12	6
PONUMAC	M	String	18	22
ACOND	C	String	40	1
ACMETHAND	C	String	41	2
ACTOTAMT	C	Numeric	43	9

12. When you are finished with the POALLCHG record, click **Close**.

Creating the POSUMMARY Record Fields

About this task

Use this procedure to create the application fields for the POSUMMARY record.

Procedure

1. Right-click the **POSUMMARY** application record and select **Edit fields**.
2. If the field you are creating is the first field in the record, click **New**.
3. In the Name box, type **CUSTNUMSUM**, which is the name of the first field in the POSUMMARY record.
4. Select the **Mandatory** check box to identify it as a required field.
5. In the Description box, type **Customer Number**.
6. From the Data Type list, select the appropriate type.

Note: Select **String** if you want to specify that this field must contain alphanumeric data.

If the Data Type is **Number** or **Dt/Tm**, select the appropriate format from the Format list.

7. In the Min Length box, type **1** as the minimum field length. If the minimum length of the field is **0**, then accept that default in the Min Length box.
8. In the max Length box, type **8** as the maximum length of the field.
9. To add the field and to create the next field positioned after it, press **Enter**.
10. Create the rest of the fields according to Application File Summary Layout for Inbound POs .
11. After adding the last field, click **Auto Position** to automatically position the fields in the record. Click **Yes** to acknowledge the warning message that fields are sequenced in order.

The Positional Field Editor dialog box should look similar to the following. (This is an ANSI X12 example; some details may be different for EDIFACT and TRADACOMS.):

Positional Field Editor

Record
 Name: POSUMMARY Tag: SUM Tag Pos: 1

Field Details
 Name: NUMLINEITEMSUM Mandatory
 Description: Number of Line Items
 Data Type: Number Format: NO
 Start Pos: 40 Min Length: 1 Max Length: 6

Fields

Name	M/C	Type	Start	Length
CUSTNUMSUM	M	String	4	8
PODATESUM	M	Date/Time	12	6
PONUMSUM	M	String	18	22
NUMLINEITEMSUM	M	Number	40	6

Buttons: Close, Cancel, Next, Delete, Auto Position

12. When you are finished with the POSUMMARY record, click **Close**.

Chapter 4. POHEADER Record

About the POHEADER Record

The POHEADER record contains the following:

- purchase order type code
- shipping and customer billing information
- delivery information
- cancel by date information (for ANSI X12 and EDIFACT)
- the number of line items in the purchase order

This header record is mandatory and occurs once for each purchase order translated.

Mapping the CUSTNUMHDR Field

The Application Integration subsystem allows you to use selected information from the Sterling Gentran:Server Partner database in your maps via a standard rule. You can map information from your trading partner's profile in the Partner Editor to a selected element or field in the map. The information that you can use in the map includes any field in a table (location, lookup, or cross-reference) or data from the partner record.

About this task

For this map, you need to pull information from the partner definition to populate the Customer Number fields on the application side of the map. Since the Customer Number is a part of the record key for each record, it appears five times in the application file format, once in each record (CUSTNUMHDR, CUSTNUMDET, CUSTNUMDESC, CUSTNUMAC, and CUSTNUMSUM).

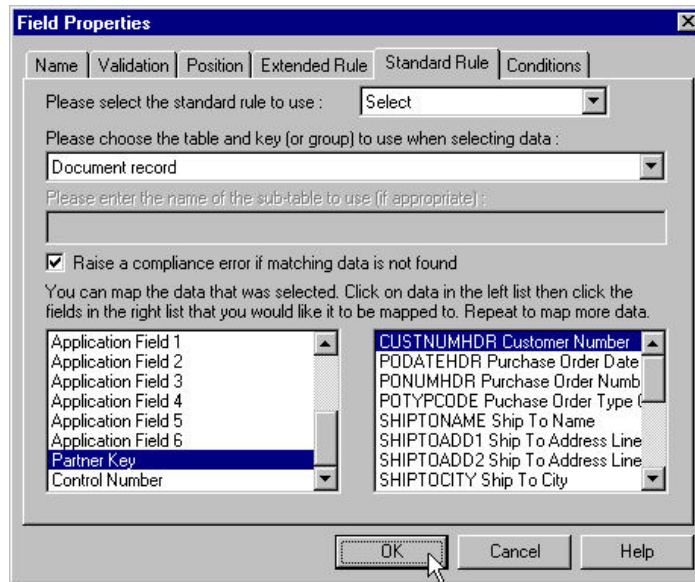
Your partner's customer number is part of the record key for each record in the application file format. It identifies which trading partner sent the purchase order. This customer number is already part of the partner definition for this partner, so you can map it from the partner definition to each customer number field.

Use this procedure to map the customer number from the document record.

Procedure

1. Double-click the **CUSTNUMHDR** field in the POHEADER record to display the Field Properties dialog box.
2. Select the **Standard Rule** tab.
3. From the select standard rule list, choose **Select**.
4. From the Table and Key list, select **Document record**.
5. Select the check box to indicate that a compliance error should be generated if the select does not find a valid entry.
6. From the Map From list on the left, select **Partner Key**.
7. From the Map To list on the right, select **CUSTNUMHDR**.

The Standard Rule tab of the Field Properties dialog box should now look like this:



8. Click OK to add the standard rule to the CUSTNUMHDR field.

PODATEHDR Field

About PODATEHDR field for EDIFACT

On the EDI side of the map, date information for the map header is contained in the DTM segment. This segment repeats multiple times. You must change the properties of the DTM02 element (Date/Time/Period) to reflect the minimum and maximum length, type, and format that your partner will send you.

If the DTM01 (Date/Time/Period Qualifier) is 4, that iteration of the DTM segment contains the order date/time. You must map the order date/time to the PODATE field in each application record (PODATEHDR, PODATEDET, PODATEDESC, PODATEAC, and PODATESUM).

To map the correct information from the DTM segment to the PODATEHDR field in the application header record, you must determine the data type, then extract the order date/time information. However, before mapping the PODATEHDR field, complete these additional tasks:

- Changing the Properties of the DTM02 Element
- Creating the TEMPDATES Temporary Storage Segment
- Creating the PODATE Temporary Storage Element
- Using an Extended Rule to Extract the Date/Time

About PODATEHDR for TRADACOMS

On the EDI side of the map, the purchase order date information is contained in the ORD segment. The ORD segment contains order reference information for the header part of the map, and repeats multiple times. You need to map the date order placed by customer (ORD ORNO ORN2) to the PODATE field in each of the application records (PODATEHDR, PODATEDET, PODATEDESC, and PODATESUM). The ORD ORNO ORN2 element can be directly linked to the PODATEHDR field in the POHEADER record. Also see Mapping the PODATEHDR Field.

Changing the Properties of the DTM02 Element

You have determined that your partner will send you six digits for the DTM02 element (Date/Time/Period). These six digits will be a Dt/Tm data type in the YYMMDD format (two-digit year, two-digit month, and two-digit day). You must change the minimum and maximum length, the type, and the format of the DTM02 element.

About this task

Use this procedure to change the length of the DTM02 element.

Procedure

1. Double-click the **DTM02** element to display the Element Properties dialog box.
2. Select the **Validation** tab.
3. In the Minimum box, type **6** as the minimum length.
4. In the Maximum box, type **6** as the maximum length.
5. From the data-type list, select **Date/Time**.

The system may prompt you with a message that changing the field's data type invalidates its format. Click **Yes** to acknowledge this message and proceed.

6. From the Format list, select **YYMMDD**.
7. Click **OK** to exit the Element Properties dialog.

Creating the TEMPDATES Temporary Storage Segment

To map the order date/time information to your application file format, you must create a temporary storage segment and element on the EDI side of the map that does not repeat. Use an extended rule to extract the order date/time information from the DTM segment and move it to the appropriate temporary storage elements. Then map the order date/time directly from the temporary storage element to the PODATEHDR application field.

About this task

Use this procedure to create the TEMPDATES temporary storage segment.

Procedure

1. Click the **DTM** segment. The temporary storage segment is located after the DTM segment, at the same level.
2. Select **Edit > Insert > Segment**. The EDI Segment Properties dialog is displayed.
3. In the Name box, type **TEMPDATES**.
4. On the **Tag** tab, type **\$\$\$** in the Tag box.
5. Click **OK** to create the TEMPDATES temporary storage segment.

Creating the PODATE Temporary Storage Element

About this task

Use this procedure to create the PODATE element.

Procedure

1. Click the **TEMPDATES** segment and select **Edit > Create Sub > Element**.
2. In the Name box, type **PODATE**.

3. In the Description box, type **Purchase Order Date**.
4. Select the **Validation** tab.
5. In the Minimum box, type **6** as the minimum length.
6. In the Maximum box, type **6** as the maximum length.
7. From the data-type list, select **Date/Time**. If the system prompts you with a message box, click **Yes** to acknowledge the message and proceed.
8. From the Format list, select **YYMMDD**, to indicate how the element is formatted.
9. Click **OK** to create the PODATE element.

Using an Extended Rule to Extract the Date/Time

Use an extended rule to extract the order date/time from the DTM segment and move it to the PODATE temporary storage element. The system executes the extended rule after reading the DTM01 (Date/Time Qualifier).

About this task

Use this procedure to create the extended rule.

Procedure

1. Double-click the **DTM01** element to display the Element Properties dialog box.
2. Select the **Extended Rule** tab.
3. In the Rule list, type the following:


```
IF #2005 = "4" THEN
BEGIN
$ORDERS.#PODATE=#2380;
END
```

The rule is executed if the DTM01 element contains 4 (order date/time).
4. Click **Compile** to validate the extended rule syntax.
5. Click **OK** to add the extended rule to the DTM01 element.

Mapping the PODATEHDR Field

About this task

Use this procedure to map the purchase order date.

Procedure

1. If the Linking function is not currently active, click **Link**.
2. Click one of the following elements:
 - the **BEG05** element in the BEG record (for ANSI X12)
 - the **PODATE** temporary storage element in the TEMPDATES segment (for EDIFACT)
 - the **ORD ORNO ORN2** element (for TRADACOMS)
3. Click the application field **PODATEHDR** in the POHEADER record.

PONUMHDR Field

Mapping the PONUMHDR

About this task

Use this procedure to map the purchase order number.

Procedure

1. If the Linking function is not currently active, click **Link**.
2. Click one of the following elements:
 - the **BEG03** element in the BEG record (for ANSI X12)
 - the **BGM05** element in the BGM record (for EDIFACT)
 - the **ORD ORNO ORN0** element (for TRADACOMS)
3. Click the application field **PONUMHDR** in the POHEADER record.

Changing the Maximum Length of the BGM05 Element - EDIFACT

Because your partner will only send 22 alphanumeric characters for the BGM05 element (Document/Message Number), you must change the maximum length of the BGM05 element.

About this task

Use this procedure to change the maximum length of the BGM05 element.

Procedure

1. Double-click the **BGM05** element to display the Element Properties dialog box.
2. Select the **Validation** tab.
3. In the Maximum box, type **22** as the maximum length.
4. Continue with Setting up the Document Name for the Inbound PO .

Setting up the Document Name for the Inbound PO

About this task

Use this procedure to set up the Document Name for this Purchase Order.

Procedure

1. Double-click one of the following elements:
 - **BEG03 (0324)** (for ANSI X12)
 - **BGM05 (0324)** (for EDIFACT)
 - **ORD ORNO ORN0** (for TRADACOMS)
2. On the Element Properties dialog box, select the **Standard Rule** tab.
3. From the standard rule list, select **Update**.
4. From the Table list, choose **Document record**. This indicates that you are updating the document record in Sterling Gentran:Server.
5. From the columns list, choose **Document Name**. This indicates that you are updating the Name field in the document browsers with the contents of this element.
6. Click **OK** to set up the document name.

Mapping the POTYPECODE Field

About this task

Complete the following steps to map the purchase order type code:

Procedure

1. If the Linking function is not currently active, click **Link**.
2. Click one of the following elements:
 - **BEG02** element in the BEG record (for ANSI X12)
 - **BGM06** element in the BGM record (for EDIFACT)
 - **ORD CD07** element (for TRADACOMS)
3. Click the application field **POTYPECODE** in the POHEADER record.

Chapter 5. Shipping and Billing

Shipping and Billing Information for ANSI X12 and EDIFACT

About Shipping and Billing Information for ANSI X12 and EDIFACT

On the EDI side of the map, the name and address information is contained in the following:

- For ANSI X12, a series of segments in the N1 group
- For EDIFACT, the NAD segment in the NAD group

These groups contain shipping and billing name and address information and repeat multiple times.

To determine whether shipping or billing information is contained in the group, look for either ST or BT in the following:

- For ANSI X12, the Entity Identifier Code in the N1 segment
- For EDIFACT, the Party Qualifier in the NAD segment

Depending on the value of the group, you need to map the shipping information to the Ship To fields or the billing information to the Bill To fields in the application header record. However, to map the correct information from the group to the Bill To and Ship To fields, you must extract the billing and the shipping information after determining the type.

For ANSI X12, in this tutorial, you have determined that your partner will only send you one N3 segment (address information), so you must change the maximum usage allowed for that segment.

For EDIFACT, in this tutorial, you have determined that your partner will only send you up to two characters for the NAD20 element (Country Sub-Entity Identification), you must change the maximum length allowed for that element.

Temporary Storage Segments for Shipping and Billing

To map the shipping and billing information to your application file format, you must create temporary storage segments and elements on the EDI side of the map that do not repeat.

Use an extended rule to extract the shipping and billing information from the N1 or NAD group and move it to the appropriate temporary storage elements. Map the shipping and billing information directly from the temporary storage elements to your application fields.

This table lists the temporary storage segments that you are creating.

Name	Desc	Tag	Min Usage	Max Usage
BillToDet	Bill To Details	\$\$\$	0	1
ShipToDet	Ship To Details	\$\$\$	0	1

Creating the ShipToDet Segment

About this task

Use this procedure to create the ShipToDet temporary storage segment.

Procedure

1. Click the **N1** (for ANSI X12) or **NAD** (for EDIFACT) group. The two temporary storage segments are located after the group, at the same level.
2. Select **Edit > Insert > Segment**. The EDI Segment Properties dialog box is displayed.
3. In the Name box, type **ShipToDet**.
4. In the Description box, type **Ship To Details**.
5. On the **Tag** tab, type **\$\$\$** in the Tag box.
6. Click **OK** to create the ShipToDet temporary storage segment.

Creating the BillToDet Segment

About this task

Use this procedure to create the BillToDet temporary storage segment.

Procedure

1. Click the **ShipToDet** segment.
2. Select **Edit > Insert > Segment**.
3. In the Name box, type **BillToDet**.
4. In the Description box, type **Bill To Details**.
5. On the **Tag** tab, type **\$\$\$** in the Tag box.
6. Click **OK** to create the BillToDet temporary storage segment.

Temporary Storage Elements for Shipping and Billing

Now, you can add elements to the temporary storage segments you just created.

This table lists the temporary storage elements.

EDI Segment	EDI Element	Max Length
ShipToDet	SHIPTONAME	35
	SHIPTOADDR1	35
	SHIPTOADDR2	35
	SHIPTOCITY (ANSI)	30
	SHIPTOCITY (EDIFACT)	35
	SHIPTOSTATE	2
	SHIPTOPCODE	9

EDI Segment	EDI Element	Max Length
BillToDet	BILLTONAME	35
	BILLTOADDR1	35
	BILLTOADDR2	35
	BILLTOCITY (ANSI)	30
	BILLTOCITY (EDIFACT)	35
	BILLTOSTATE	2
	BILLTOPCODE	9

Creating the ShipToDet Elements

About this task

Use this procedure to create the ShipToDet elements.

Procedure

1. Click the **ShipToDet** segment and select **Edit > Create Sub > Element**.
2. In the Name box, type **SHIPTONAME**.
3. On the **Validation** tab type **35** in the Maximum box.
4. Click **OK** to create the SHIPTONAME element.
5. Now you can create the remaining Ship To elements. With the **SHIPTONAME** element highlighted, select **Edit > Insert > Element**.
6. Repeat Steps 2 - 5 to complete the remaining ShipToDet elements listed in the previous topic.

Creating the BillToDet Elements

About this task

Use this procedure to create the BillToDet elements.

Procedure

1. Click the **BillToDet** segment and select **Edit > Create Sub > Element**.
2. In the Name box, type **BILLTONAME**.
3. On the **Validation** tab type **35** in the Maximum box.
4. Click **OK** to create the BILLTONAME element.
5. Now you can create the remaining Bill To elements. With the **BILLTONAME** element highlighted, select **Edit > Insert > Element**.
6. Repeat Steps 2 - 5 to complete the remaining Bill To elements listed in the table above.

Changing the Maximum Use of the N3 Segment - ANSI X12

You have determined that your partner only sends you one N3 segment (address information), so you must change the maximum usage allowed for that segment.

About this task

Use this procedure to make N3 a nonrepeating segment.

Procedure

1. Right-click the N3 segment (in the N1 group) and select **Properties**.
2. Select the **Looping** tab.
3. In the Max Usage box, type **1** to make N3 a non repeating segment. This box is currently set to 2.
4. Click **OK** to accept the change to the N3 segment.

Changing the Max Length of NAD20 - EDIFACT

Because your partner will only send you two alphanumeric characters for the NAD20 element (Country Sub-Entity Identification), which represent the state, province, or country of the party identified in the NAD01 element (Party Qualifier), you must change the maximum length of the NAD20 element.

About this task

Use this procedure to change the length of the NAD20 element.

Procedure

1. Double-click the **NAD20** element to display the Element Properties dialog box.
2. Select the **Validation** tab.
3. In the Maximum box, type **2** as the maximum length.
4. Click **OK** to exit the Element Properties dialog box.

Creating an Extended Rule to Extract Shipping And Billing Information

Use an extended rule to extract the shipping and billing information from the group and move it to the appropriate temporary storage element. After the system reads an occurrence of the group and executes the extended rule.

About this task

Use this procedure to create the extended rule.

Procedure

1. Right-click one of the following groups and select **Extended Rules**:
 - **N1** (for ANSI X12)
 - **NAD** (for EDIFACT)
2. In the Group Properties dialog box, click the **On End** option to specify that the rule will be executed when the loop terminates. The system loads an occurrence of the group (containing the billing or shipping information) and then executes this rule.

Note: If a segment/record or element/field occurs more than once in a map, it is identified by its name <ID>. Subsequent occurrences are identified by <ID>:n, where 'n' is the number of the occurrence in the map.

3. **For ANSI X12:** In the Editor list, type the following:

```
IF #0098 = "BT" THEN
BEGIN
  $850.#BILLTNAME = #0093;
  $850.#BILLTOADDR1 = #0166;
  $850.#BILLTOADDR2 = #0166:2;
  $850.#BILLTOCITY = #0019;
```

```

$850.#BILLTOSTATE = #0156;
$850.#BILLTOPCODE = #0116;
END
IF #0098 = "ST" THEN
BEGIN
$850.#SHIPTONAME = #0093;
$850.#SHIPTOADDR1 = #0166;
$850.#SHIPTOADDR2 = #0166:2;
$850.#SHIPTOCITY = #0019;
$850.#SHIPTOSTATE = #0156;
$850.#SHIPTOPCODE = #0116;
END

```

4. For **EDIFACT**: In the Editor box, type the following:

```

IF #3035 = "BT" THEN
BEGIN
$ORDERS.#BILLTONAME = #3036;
$ORDERS.#BILLTOADDR1 = #3042;
$ORDERS.#BILLTOADDR2 = #3042:2;
$ORDERS.#BILLTOCITY = #3164;
$ORDERS.#BILLTOSTATE = #3229;
$ORDERS.#BILLTOPCODE = #3251;
END
IF #3035 = "ST" THEN
BEGIN
$ORDERS.#SHIPTONAME = #3036;
$ORDERS.#SHIPTOADDR1 = #3042;
$ORDERS.#SHIPTOADDR2 = #3042:2;
$ORDERS.#SHIPTOCITY = #3164;
$ORDERS.#SHIPTOSTATE = #3229;
$ORDERS.#SHIPTOPCODE = #3251;
END

```

5. Click **Compile** to validate the syntax of the extended rule.
6. Click **OK** to add the extended rule to the NAD group.

Mapping Shipping Information for ANSI X12 and EDIFACT

After creating the temporary storage segments and elements and adding the extended rule, you can map the shipping and billing information directly from the temporary storage elements to your application fields.

About this task

Use this procedure to map the shipping information.

Procedure

1. If the Linking function is not currently active, click **Link**.

Notes:

- All of the elements you are linking *from* are in either the **ShipToDet** or the **BillToDet** temporary storage segment.
 - All of the application fields you are linking *to* are in the **POHEADER** record.
2. Click the **SHIPTONAME** element in the **ShipToDet** temporary storage segment and then the **SHIPTONAME** field.
 3. Click the **SHIPTOADDR1** element (in **ShipToDet**) to then the **SHIPTOADD1** field.
 4. Click the **SHIPTOADDR2** element (in **ShipToDet**) to then the **SHIPTOADD2** field.

5. Click the **SHIPTOCITY** element (in **ShipToDet**) to then the **SHIPTOCITY** field.
6. Click the **SHIPTOSTATE** element (in **ShipToDet**) to then the **SHIPTOSTATE** field.
7. Click the **SHIPTOPCODE** element (in **ShipToDet**) to then the **SHIPTOPOSTALCODE** field.
8. Click the **BILLTONAME** element (in **BillToDet**) to then the **BILLTONAME** field.
9. Click the **BILLTOADDR1** element (in **BillToDet**) to then the **BILLTOADD1** field.
10. Click the **BILLTOADDR2** element (in **BillToDet**) to then the **BILLTOADD2** field.
11. Click the **BILLTOCITY** element (in **BillToDet**) to then the **BILLTOCITY** field.
12. Click the **BILLTOSTATE** element (in **BillToDet**) to then the **BILLTOSTATE** field.
13. Click the **BILLTOPCODE** element (in **BillToDet**) to then the **BILLTOPOSTALCODE** field.

Date Segments and Elements for ANSI X12 and EDIFACT

Temporary Storage for Date Segments and Elements

The DTM segment contains all the date information for the purchase order and repeats multiple times. If the DTM segment contains delivery date information, you must map the delivery information to the DELDATE field in the application header record. If the DTM segment contains cancellation information, you must map the cancel by date information to the CANCELDATE field in the application header record.

An iteration of the DTM segment contains Delivery Date information if one of the following is true:

- For ANSI X12: the Date/Time Qualifier (in the DTM segment) is 001
- For EDIFACT: the Date/Time/Period/Qualifier (in the DTM01 segment) is 117

An iteration of the DTM segment contains cancel by date information if one of the following is true:

- For ANSI X12: the Date/Time Qualifier is 002 (Cancel By Date)
- For EDIFACT: the DTM01 (Date/Time/Period/Qualifier) is 153 (Latest Cancellation Date/Time)

To map the delivery and cancel by date information to your application file, you must create a temporary storage segment and elements on the EDI side of the map. You can use an extended rule to extract the delivery and cancel by date information from the DTM segment and move it to the appropriate temporary storage elements. Then, you can map the delivery and cancel by date information directly from the temporary storage elements to your application fields.

For EDIFACT, you have already changed the properties of the DTM02 element (Date/Time/Period) to reflect the minimum and maximum length, the type, and the format that your partner will send you.

Creating a Temporary Storage Date Segment

About this task

Use this procedure to create the TEMPDATES temporary storage segment.

Procedure

1. Click the **DTM** segment and select **Edit > Insert > Segment**.
2. In the Name box, type **TEMPDATES**.
3. In the Description box, type **Temporary Storage Dates**.
4. On the **Tag** tab, type **\$\$\$** in the Tag box.
5. Click **OK** to create the TEMPDATES temporary storage segment.

Creating a Temporary Storage Date Element

About this task

Use this procedure to create the TEMPDATES elements in the temporary storage segment you just created.

Procedure

1. Click the **TEMPDATES** segment and select **Edit > Create Sub > Element**.
 2. In the Name box, type **DELDATE**.
 3. In the Description box, type **Delivery Date**.
 4. On the **Validation** tab, do the following:
 - In the Minimum box, type **6** as the minimum length.
 - In the Maximum box, type **6** as the maximum length.
 - From the Type list, select **Date/Time**. The system may display a message that changing the field's date type will invalidate its format. Click **Yes** to acknowledge this message and proceed.
- Note:** To turn off the message prompt, select **Options > Preferences**. Select the **Confirmations** tab and clear the **Confirm everything** check box. From the confirm list, clear the **Change format or datatype of a field** check box. Click **OK** to save.
- From the Format list, select **YYMMDD**.
 5. Click **OK** to create the DELDATE element.
 6. To create the CANCELDATE element, click the **DELDATE** element and select **Edit > Insert > Element**.
 7. In the Name box, type **CANCELDATE**.
 8. In the Description box, type **Cancel By Date**.
 9. On the **Validation** tab, do the following:
 - In the Minimum box, type **6** as the minimum length.
 - In the Maximum box, type **6** as the maximum length.
 - From the Type list, select **Date/Time**.
 - From the Format list, select **YYMMDD**.
 10. Click **OK** to create the CANCELDATE element.

Using an Extended Rule to Extract Delivery and Cancel by Dates

Use an extended rule to extract the delivery and cancel by date information from the DTM segment and move it to the appropriate temporary storage elements. The extended rule should be executed after the system reads the DTM01 (Date/Time Qualifier).

About this task

Use this procedure to create the extended rule.

Procedure

1. Double-click the **DTM01** element to display the Element Properties dialog box.
2. Select the **Extended Rule** tab.
3. **For ANSI X12:** In the Rule list, type the following:

```
IF #0374:6="002" THEN
BEGIN
    $850.#DELDATE=#0373:6;
END
IF #0374:6="001" THEN
BEGIN
    $850.#CANCELDATE=#0373:6;
END
```

The system loads the DTM02 element (containing the delivery and cancel by date information), and then executes this rule.

4. **For EDIFACT:** In the Rule list, type the following:

```
IF #2005 = "117" THEN
BEGIN
    $ORDERS.#DELDATE = #2380;
END
IF #2005 = "153" THEN
BEGIN
    $ORDERS.#CANCELDATE = #2380;
END
```

The Rule list already contains the extended rule that extracts the order date/time from the DTM segment (if the Date/Time/Period/Qualifier is 4) and moves it to the PODATE temporary storage element.

5. Click **Compile** to validate the syntax of the extended rule.
6. Click **OK** to add the extended rule to the DTM01 element.

Mapping the Delivery Date Information

After creating the temporary storage segment and elements and adding the extended rule, you can map the delivery and cancel by date information directly from the temporary storage elements to your application fields.

About this task

Use this procedure to map the delivery date information.

Procedure

1. To map the delivery date, link the **DELDATE** element in the **TEMPDATES** temporary storage segment to the application field **DELDATE** in the **POHEADER** record.

2. To map the cancel by date, link the **CANCELDATE** element in the **TEMPDATES** temporary storage segment to the application field **CANCELDATE** in the **POHEADER** record.

NUMLINEITEMS

On the EDI side of the map, the **CNT** segment contains control total information. The segment contains all the control values and repeats multiple times. If the **CNT01** (Control Qualifier) is 2, that iteration of the **CNT** segment contains the number of line items in the message. You must map the total number of line items to the **NUMLINEITEMS** field in the **POHEADER** application record and the **NUMLINEITEMSUM** field in the **POSUMMARY** application record.

To map the correct information from the **CNT** segment to the **NUMLINEITEMS** field in the application header record, you must determine the control value type and extract the number of line items.

To map the number of line items to your application file format, you need to create a temporary storage segment and element on the EDI side of the map that do not repeat. You can use an extended rule to extract the number of line items from the **CNT** segment and move it to the appropriate temporary storage element. Map the number of line items directly from the temporary storage element to the **NUMLINEITEMS** application field.

Creating the **TEMPCONTROL** Temporary Storage Segment About this task

Use this procedure to create the **TEMPDATES** temporary storage segment.

Procedure

1. Click the **CNT** segment and select **Edit > Insert > Segment**.
2. In the Name box, type **TEMPCONTROL**.
3. On the **Tag** tab, type **\$\$\$** in the Tag box.
4. Click **OK** to create the **TEMPCONTROL** temporary storage segment.

Creating the **NUMLINEITEMS** Temporary Storage Element About this task

Add an element to the temporary storage segment you just created. Use this procedure to create the **NUMLINEITEMS** element.

Procedure

1. Click the **TEMPCONTROL** segment and select **Edit > Create Sub > Element**.
2. In the Name box, type **NUMLINEITEMS**.
3. In the Description box, type **Number of Line Items**.
4. On the **Validation** tab, do the following:
 - In the Maximum box, type **6** as the maximum length.
 - From the data-type list, select **Number**.
 - From the Format list, select **R9**. This indicates that the element will be formatted with an explicit decimal point and contain up to nine places.
5. Click **OK** to create the **NUMLINEITEMS** element.

Creating an Extended Rule to Extract the Number of Line Items

Use an extended rule to extract the number of line items from the CNT segment and move it to the NUMLINEITEMS temporary storage element. After the system reads an occurrence of the CNT segment, it executes the extended rule.

About this task

Use this procedure to create the extended rule.

Procedure

1. Right-click the **CNT** segment and select **Extended Rules** to display the EDI Segment Properties dialog box.
2. Select the **On End** option. This specifies that the rule will be executed when the loop terminates. The system loads an occurrence of the CNT segment (containing the control value information), and executes this rule.
3. In the Editor box, type the following:

```
IF #6069 = "2" THEN
BEGIN
  $ORDERS.#NUMLINEITEMS = #6066;
END
```
4. Click **Compile** to validate the extended rule syntax.
5. Click **OK** to add the extended rule to the CNT segment.

Mapping the NUMLINEITEMS Field for ANSI X12 and EDIFACT

About this task

Use this procedure to map the NUMLINEITEMS field.

Procedure

- For ANSI X12: Link the **CTT01** element in the CTT segment to the application field **NUMLINEITEMS** in the POHEADER record.
- For EDIFACT: Link the **NUMLINEITEMS** temporary storage element in the TEMPCONTROL temporary storage segment to the application field **NUMLINEITEMS** in the POHEADER record.

Shipping and Billing Information for TRADACOMS

About Shipping and Billing Information for TRADACOMS

Mapping the Shipping Information for TRADACOMS

On the EDI side of the map, the shipping name and address information is contained in the CLO (Customer's Location) segment. You need to map the shipping information to the Ship To fields in the application header record.

About this task

Use this procedure to map the shipping information.

Procedure

1. If the Linking function is not currently active, click **Link**.

Note: The application fields you are linking to are all in the POHEADER record.

2. Click the **CLO CNAM** element (Customer's Name) and then the **SHIPTONAME** field.
3. Click the **CLO CADD CAD0** element (Customer's Address Line 1) and then the **SHIPTOADD1** field.
4. Click the **CLO CADD CAD1** element (Customer's Address Line 2) and then the **SHIPTOADD2** field.
5. Click the **CLO CADD CAD2** element (Customer's Address Line 3) and then the **SHIPTOCITY** field.
6. Click the **CLO CADD CAD3** element (Customer's Address Line 4) and then the **SHIPTOSTATE** field.
7. Click the **CLO CADD CAD4** element (Customer's Post Code) and then the **SHIPTOPOSTALCODE** field.

Mapping Customer Information for TRADACOMS

On the EDI side of the map, the customer name and address information is contained in the CDT (Customer Details) segment. You need to map the customer name and address information to the Customer fields in the application header record.

About this task

Use this procedure to map the customer information.

Procedure

1. If the Linking function is not currently active, click **Link**.

Note: The application fields you are linking to are all in the POHEADER record.

2. Click the **CDT CNAM** element (Customer's Name) and then the **CUSTNAME** field.
3. Click the **CDT CADD CAD0** element (Customer's Address Line 1) and then the **CUSTADD1** field.
4. Click the **CDT CADD CAD1** element (Customer's Address Line 2) and then the **CUSTADD2** field.
5. Click the **CDT CADD CAD2** element (Customer's Address Line 3) and then the **CUSTCITY** field.
6. Click the **CDT CADD CAD3** element (Customer's Address Line 4) and then the **CUSTSTATE** field.
7. Click the **CDT CADD CAD4** element (Customer's Post Code) and then the **CUSTPOSTALCODE** field.

Mapping the DELDATE Field for TRADACOMS

On the EDI side of the map, the delivery date information is contained in the DIN (Delivery Instructions) segment. You can map the delivery date directly from the DIN LDAT element (Latest Delivery Date).

About this task

Use this procedure to map the delivery date.

Procedure

1. If the Linking function is not currently active, click **Link**.
2. Click the **DIN LDAT** element (Latest Delivery Date).
3. Click the application field **DELDATE** in the POHEADER record.

Mapping the NUMLINEITEMS Field for TRADACOMS

You can map the NUMLINEITEMS field directly from the OTR LORD element (Lines Ordered), by using the Link function. Later in this tutorial, you will also link the OTR LORD element to the NUMLINEITEMSUM field in the POSUMMARY record.

About this task

Use this procedure to map the number of line items.

Procedure

1. If the Linking function is not currently active, click **Link**.
2. Click the **OTR LORD** element (Lines Ordered).
3. Click the application field **NUMLINEITEMS** in the POHEADER record.

Chapter 6. PODetailGroup Records

PODETAIL Record

About the PODETAIL Record

The PODetailGroup (detail information) contains all the line item information for the order, including the item, unit of measure, price, quantity ordered, description, and the allowance and charge information.

Each PODetailGroup can occur up to 100,000 times (for ANSI X12), 200,000 (for EDIFACT), or 999,999 times (for TRADACOMS) per order translated. The information within the PODetailGroup is further segregated by record:

- PODETAIL
- PODESC
- POALLCHG. (for ANSI X12 and EDIFACT)

The following topics describe the actions necessary for you to map to each field in the PODETAIL (detail) record. The PODETAIL (detail) record contains line item code number, unit of measure, price, and quantity ordered. The PODETAIL record occurs once per PODetailGroup.

Mapping the CUSTNUMDET Field

The Application Integration subsystem allows you to use selected information from the Sterling Gentran:Server Partner Editor in your maps using a standard rule. You can map information from your trading partner's profile in the Partner Editor to a selected element or field in the map. The information that you can use in the map includes any field in a table (location, lookup, or cross-reference) or data from the partner key.

About this task

For this map, you need to pull information from the partner definition to populate the Customer Number fields on the application side of the map. Since the Customer Number is a part of the record key for each record, it appears five times in the application, once in each record (CUSTNUMHDR, CUSTNUMDET, CUSTNUMDESC, CUSTNUMAC, and CUSTNUMSUM).

Your partner's customer number is part of the record key for each record in the application. It identifies which trading partner sent the purchase order. This customer number is already part of the partner definition for this partner, so you can map it from the partner definition to each customer number field.

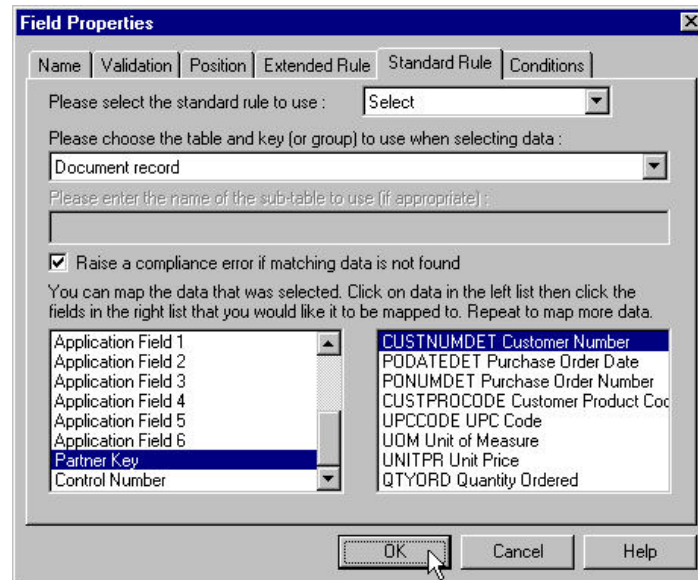
Use this procedure to map the customer number from the partner definition.

Procedure

1. Double-click the **CUSTNUMDET** field (in the PODETAIL record) to display the Field Properties dialog box.
2. Select the **Standard Rule** tab.
3. From the Standard Rule list, choose **Select**.
4. From the Table and Key list, select **Document record**.

5. Select the check box to indicate that a compliance error should be generated if the select does not find a valid entry.
6. From the Map From list on the left, select **Partner Key**.
7. From the Map To list on the right, select **CUSTNUMDET**.

The Standard Rule tab on the Field Properties dialog box should look similar to the following. (This is an ANSI X12 example; some details may be different for EDIFACT and TRADACOMS.)



8. Click **OK** to add the standard rule to the CUSTNUMDET field.

Mapping the PODATEDET Field

For ANSI X12, the BEG05 (Purchase Order Date) element contains the purchase order date. You need to map the information from the BEG05 element to the PODATEDET field in the PODetailGroup, which repeats 100,000 times.

About this task

For EDIFACT, the purchase order date is contained in the DTM segment. The DTM segment contains date information for the header part of the map and repeats multiple times. If the DTM01 (Date/Time/Period Qualifier) is 4, that iteration of the DTM segment contains the order date/time. You must map the order date/time to the PODATE field in each application record (PODATEHDR, PODATEDET, PODATEDESC, PODATEAC, and PODATESUM).

You created a temporary storage segment and element and used an extended rule to extract the order date/time information from the DTM segment and move it to the PODATE temporary storage element. You must map the information from the PODATE temporary storage element to the PODATEDET field in the PODetailGroup, which repeats 200,000 times.

For TRADACOMS, the purchase order date information is contained in the ORD segment. The ORD segment contains order reference information for the header part of the map and repeats multiple times. You need to map the date order placed by customer (ORD ORNO ORN2) to the PODATE field in each of the application records (PODATEHDR, PODATEDET, PODATEDESC, and PODATESUM).

To map the purchase order date to this application field, you need to use an extended rule to extract the purchase order date from the element and move it to the `PODATEDET` field.

Use this procedure to create the extended rule.

Procedure

1. Double-click the `PODATEDET` field to display the Field Properties dialog box.
2. Select the **Extended Rule** tab.
3. In the Rule list, type one of the following:
 - For ANSI X12:
`#PODATEDET=$850.#0323;`
The `$850.#0323` is the addressing for the `BEG05` element.
 - For EDIFACT:
`#PODATEDET = $ORDERS.#PODATE;△`
The `$ORDERS.#PODATE` is the addressing for the `PODATE` temporary storage element.
 - For TRADACOMS:
`#PODATEDET = $ORDERS.#ORN2;△`
The `$ORDERS.#ORN2` is the addressing for the `ORD ORNO ORD2` element.The system executes this rule when it processes the `PODATEDET` field.
4. Click **Compile** to validate the syntax of the extended rule.
5. Click **OK** to add the extended rule to the `PODATEDET` field.

Mapping the `PONUMDET` Field

For ANSI X12, the `BEG03` (Purchase Order Number) element contains the purchase order number. The `BEG` segment does not repeat. You need to map the information from the `BEG03` element to the `PONUMDET` field. However, the `PONUMDET` field is included in the `PODetailGroup`, which repeats 100,000 times.

About this task

For EDIFACT, the `BGM05` (Document/Message Number) element contains the purchase order number. The `BGM` segment does not repeat. You must map the information from the `BGM05` element to the `PONUMDET` field. The `PONUMDET` field is included in the `PODetailGroup`, which repeats 200,000 times.

For TRADACOMS, the `ORD ORNO ORN0` element contains the customer order number. The `ORD` segment does not repeat. You need to map the information from the `ORD ORNO ORN0` element to the `PONUMDET` field. However, the `PONUMDET` field is included in the `PODetailGroup`, which repeats 999,999 times.

To map the purchase order date to this application field, you need to use an extended rule to extract the purchase order date from the element and move it to the `PONUMDET` field.

Use this procedure to create the extended rule.

Procedure

1. Double-click the `PONUMDET` field to display the Field Properties dialog box.
2. Select the **Extended Rule** tab.

3. In the Rule list, type one of the following:
 - For ANSI X12:
#PONUMDET=\$850.#0324;△
The \$850.#0324 is the addressing for the BEG03 element.
 - For EDIFACT:
#PONUMDET = \$ORDERS.#1004;△
The \$ORDERS.#1004 is the addressing for the BGM5 element.
 - For TRADACOMS:
#PONUMDET = \$ORDERS.#ORN0;△
The \$ORDERS.#ORN0 is the addressing for the ORD ORNO ORD0 element.
The system executes this rule when it processes the PONUMDET field.
4. Click **Compile** to validate the syntax of the extended rule.
5. Click **OK** to add the extended rule to the PONUMDET field.

Mapping the CUSTPROCEDURE/UPCCODE Fields

You need to map information from a cross-reference table for the Petzone partner profile to translate your partner's code for a purchased item to your code for the same item.

About this task

Your partner uses a customer product code (CUSTPROCEDURE), and your company uses a UPC code (UPCCODE). After you and your partner determine what the equivalent customer product code is for each UPC code, you can create a cross-reference table in the Partner Editor and use that information in your map. For this tutorial, use the cross-reference table included in the tutorial data. See Cross-reference Tables for more information.

You need to use a standard rule to map the translated value from the CUSTPROCEDURE field to the UPCCODE field.

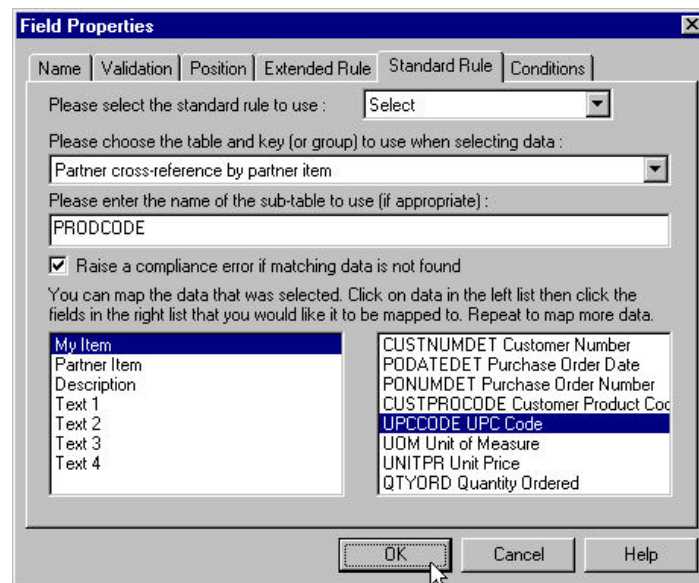
Use this procedure to map the customer's item code and then translate your partner's customer product code to your UPC code for each item your partner ordered.

Procedure

1. If the Linking function is not currently active, click **Link**.
2. Do one of the following:
 - For ANSI X12: Link the **PO107** element in the PO1 segment (in the PO1 group) to the application field **CUSTPROCEDURE** in the PODETAIL record.
 - For EDIFACT: Link the **LIN03** element in the LIN segment (in the LIN group) to the application field **CUSTPROCEDURE** in the PODETAIL record.
 - For TRADACOMS: Link the **OLD:2 CPRO CPR1** element (Customer's Item Code) to the application field **CUSTPROCEDURE** in the PODETAIL record.
3. Double-click the **CUSTPROCEDURE** field (in the PODETAIL record).
The Field Properties dialog box is displayed.
4. Select the **Standard Rule** tab.
5. From the Standard Rule list, choose **Select**.

6. From the Table and Key box, select **Partner cross-reference by partner item**. This indicates that you are using a cross-reference table and translating by your item.
7. In the Sub Table box, type **PRODCODE**. This is the name of the cross-reference table.
8. Select the check box to signal the system that if this translation fails, you want the system to generate an error.
9. From the Map From list on the left, select **My Item**. This is the field from which the contents are mapped.
10. From the Map To list on the right, select **UPCCODE**.

The Standard Rule tab of the Field Properties dialog box should look similar to the following. (This is an ANSI X12 example; some details may be different for EDIFACT and TRADACOMS.)



11. Click **OK** to map from the cross-reference table.

CONSUMERUNITS Field—TRADACOMS

On the EDI side of the map, the consumer units information is contained in the OLD:2 UNOR UNO0 element (Consumer Units in Traded Units), which is defined as a string-type element in the standard. However, you have determined that your partner will send you a number-type element in N0 format (formatted with an implicit decimal point and no decimal place).

You need to change the properties of the OLD:2 UNOR UNO0 element to reflect the type and the format that you have determined that your partner will send you. Then, you can map the consumer units information directly from the OLD:2 UNOR UNO0 element to the CONSUMERUNITS application field.

Complete the following tasks:

- Changing the Properties of the OLD:2 UNOR UNO0 Element
- Mapping the Consumer Units

Changing the Properties of the OLD:2 UNOR UNO0 Element

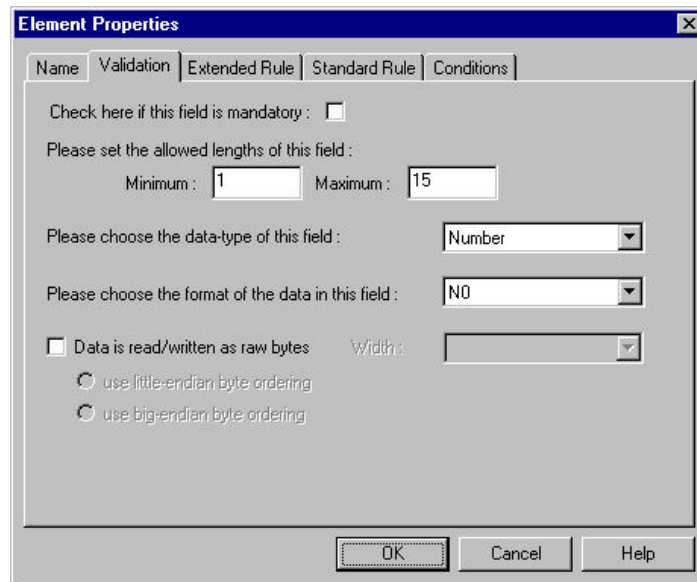
About this task

Use this procedure to change the type and format of the OLD:2 UNOR UNO0 element.

Procedure

1. Double-click the **OLD:2 UNOR UNO0** element (Consumer Units in Traded Units) to display the Element Properties dialog box.
2. Select the **Validation** tab.
3. From the data-type list, select **Number** to indicate that this is a number element. The system will prompt you with a message that changing the field's data type invalidates its format. Click **Yes** to acknowledge this message and proceed.
4. From the Format list, select **N0** to indicate that this element will be formatted with an implicit decimal point and no decimal place.

The Field Properties dialog box should now look like this:



The screenshot shows the 'Element Properties' dialog box with the 'Validation' tab selected. The dialog has a title bar with a close button. Below the title bar are five tabs: 'Name', 'Validation', 'Extended Rule', 'Standard Rule', and 'Conditions'. The 'Validation' tab is active. The main area contains the following options and fields:

- 'Check here if this field is mandatory':
- 'Please set the allowed lengths of this field':
 - Minimum:
 - Maximum:
- 'Please choose the data-type of this field':
- 'Please choose the format of the data in this field':
- 'Data is read/written as raw bytes':
- 'Width':
- Radio buttons for byte ordering:
 - use little-endian byte ordering
 - use big-endian byte ordering

At the bottom of the dialog are three buttons: 'OK', 'Cancel', and 'Help'.

5. Click **OK** to exit the Element Properties dialog box.

Mapping the Consumer Units

About this task

Use this procedure to map the consumer units.

Procedure

1. If the Linking function is not currently active, click **Link**.
2. Link the **OLD:2 UNOR UNO0** element (Consumer Units in Traded Units) to the application field **CONSUMERUNITS** in the **PODETAIL** record.

UOM Field

Mapping the UOM Field

About this task

Use this procedure to map the UOM field.

Procedure

- For ANSI X12: Link the **PO103** element in the PO1 segment (in the PO1 group) to the application field **UOM** in the PODETAIL record.
- For TRADACOMS: Link the **OLD:2 UNOR CD04** element in the PO1 segment (in the PO1 group) to the application field **UOM** in the PODETAIL record.
- For EDIFACT: Complete the following tasks:
 1. Creating the **TEMPLINEITEM** Temporary Storage Segment
 2. Creating the **UNITMEAS** Temporary Storage Element
 3. Creating an Extended Rule to Extract the Unit of Measurement
 4. Mapping the Unit of Measurement

Creating the **TEMPLINEITEM** Temporary Storage Segment

On the EDI side of the map, the line item information is contained in the LIN group (in the LIN, IMD:2, QTY:3, and MOA:5 segments). The QTY:3 segment repeats multiple times. You need to map the unit of measurement from the QTY:3 segment to the PODETAIL application record, which does not repeat. To map the unit of measurement from the QTY:303 element to the PODETAIL record, you must create a temporary storage segment and element on the EDI side of the map that do not repeat.

About this task

Use an extended rule to extract the unit of measurement from the QTY:3 segment and move it to the appropriate temporary storage element. Map the unit of measurement directly from the temporary storage element to the UOM application field.

Use this procedure to create the **TEMPLINEITEM** temporary storage segment.

Procedure

1. Click the **LIN** segment and select **Edit > Insert > Segment**.
2. In the Name box, type **TEMPLINEITEM**.
3. On the **Tag** tab, type **\$\$\$** in the Tag box.
4. Click **OK** to create the **TEMPLINEITEM** temporary storage segment.

Creating the **UNITMEAS** Temporary Storage Element

About this task

Use this procedure to create the **UNITMEAS** temporary storage element.

Procedure

1. Click the **TEMPLINEITEM** segment and select **Edit > Create Sub > Element**.
2. In the Name box, type **UNITMEAS**.
3. In the Description box, type **Unit of Measurement**.
4. On the **Validation** tab, type **3** in the Maximum box.

5. Click **OK** to create the UNITMEAS element.

Creating an Extended Rule to Extract the Unit of Measurement

Use an extended rule to extract the unit of measurement from the QTY:3 segment and move it to the UNITMEAS temporary storage element. After the system reads the QTY:303 element (Measure unit Qualifier), it executes the extended rule.

About this task

Use this procedure to create an extended rule.

Procedure

1. Double-click the **QTY:303** element to display the Element Properties dialog box.
2. Select the **Extended Rule** tab.
3. In the Rule box, type the following:

```
IF #6063 = "21" THEN
BEGIN
    $2500_LIN.#UNITMEAS = #6411:2;
END
```

The system verifies that the QTY:303 element does not contain 21 (quantity ordered) and executes this rule.

4. Click **Compile** to validate the syntax of the extended rule.
5. Click **OK** to add the extended rule to the QTY:303 element.

Mapping the Unit of Measurement

About this task

Use this procedure to map the unit of measurement.

Procedure

Link the UNITMEAS temporary storage element in the TEMPLINEITEM temporary storage segment (in the LIN group) to the application field UOM in the PODETAIL record.

UNITPR Field

Mapping the UNITPR Field

About this task

Use this procedure to map the UNITPR field.

Procedure

- For ANSI X12: Link the **PO104** element in the PO1 segment (in the PO1 group) to the application field **UNITPR** in the PODETAIL record.
- For TRADACOMS: Link the **OLD:2 OUCT OUC0** element in the PO1 segment (in the PO1 group) to the application field **UNITPR** in the PODETAIL record.
- For EDIFACT: Complete the following tasks:
 1. Creating the UNITPRICE Temporary Storage Element
 2. Creating an Extended Rule to Extract the Unit Price
 3. Mapping the Unit Price

Creating the UNITPRICE Temporary Storage Element

The MOA:5 segment repeats multiple times. You must map the unit price from the MOA:5 segment to the PODETAIL application record, which does not repeat. The MOA:5 segment contains monetary amount information, and repeats multiple times. If the MOA:501 element (Monetary Amount Type) is **146**, that iteration of the MOA:5 segment contains the unit price for that line item. You must map the unit price to the UNITPR field in the PODETAIL application record.

About this task

To map the correct information from the MOA:5 segment to the UNITPR field, you must extract the unit price after determining the type of monetary amount. You must create a temporary storage element in the TEMPLINEITEM temporary storage segment that you have already created.

Use an extended rule to extract the unit price from the MOA:5 segment and move it to the appropriate temporary storage element. You can then map the unit price directly from the temporary storage element to the UNITPR application field.

Use this procedure to create the UNITPRICE temporary storage element.

Procedure

1. Click the **UNITMEAS** temporary storage element and select **Edit > Insert > Element**.
2. In the Name box, type **UNITPRICE**.
3. In the Description box, type **Unit Price**.
4. On the **Validation** tab, do the following:
 - In the Maximum box, type **14** as the maximum length.
 - From the data-type list, select **Number**.
 - From the Format list, select **R9** to indicate that the element will be formatted with an explicit decimal point and contain up to nine places.
5. Click **OK** to create the UNITPRICE element.

Creating an Extended Rule to Extract the Unit Price

Use an extended rule to extract the unit price from the MOA:5 segment and move it to the UNITPRICE temporary storage element. After the system reads the MOA:501 element (Monetary Amount Type), it executes the extended rule.

About this task

Use this procedure to create an extended rule to extract the unit price.

Procedure

1. Double-click the **MOA:501** element to display the Element Properties dialog box.
2. Select the **Extended Rule** tab.
3. In the Rule box, type the following:

```
IF #5025 = "145" THEN
BEGIN
    $2500_LIN.#UNITPRICE = #5004;
END
```

The system verifies that the MOA:501 element does not contain 146 (unit price) and executes this rule.

4. Click **Compile** to validate the syntax of the extended rule.
5. Click **OK** to add the extended rule to the MOA:501 element.

Mapping the Unit Price

About this task

Use this procedure to map the unit price.

Procedure

Link the **UNITPRICE** temporary storage element in the **TEMPLINEITEM** temporary storage segment (in the **LIN** group) to the application field **UNITPR** in the **PODETAIL** record.

QTYORD Field

Mapping the QTYORD Field

About this task

Use this procedure to map the QTYORD field.

Procedure

- For ANSI X12: Link the **PO102** element in the **PO1** segment (in the **PO1** group) to the application field **QTYORD** in the **PODETAIL** record.
- For TRADACOMS: Link the **OLD:2 OQTY OQT0** element in the **PO1** segment (in the **PO1** group) to the application field **QTYORD** in the **PODETAIL** record.
- For EDIFACT: Complete the following tasks:
 1. Changing the Properties of the QTYORD Field
 2. Creating the QTYORD Temporary Storage Element
 3. Creating an Extended Rule to Extract the Unit Price
 4. Mapping the Quantity Ordered

Changing the Properties of the QTYORD Field

You must map the quantity ordered from the **QTY:3** segment to the **PODETAIL** application record, which does not repeat. The **QTY:3** segment contains quantity information and repeats multiple times. If the **QTY:301** element (Quantity Qualifier) is **21**, that iteration of the **QTY:3** segment contains the quantity ordered for that line item. You must map the quantity ordered to the **QTYORD** field in the **PODETAIL** application record.

About this task

To map the correct information from the **QTY:3** segment to the **QTYORD** field, you must determine the quality type and extract the quantity ordered. You must create a temporary storage element in the **TEMPLINEITEM** temporary storage segment that you have already created.

The **QTYORD** field is defined as a string-type element in the standard. However, if you determine that your partner will send you a number-type element, you will need to change the properties of the field.

Use this procedure to change the type and format of the **QTYORD** field.

Procedure

1. Double-click the **QTYORD** field element to display the Element Properties dialog box.
2. Select the **Validation** tab.
3. From the data-type list, select **Number** to indicate that this is a number element.

The system will prompt you with a message that changing the field's data type will invalidate its format. Click **Yes** to acknowledge this message and proceed.

4. From the Format list, select **N0** to indicate that this element will be formatted with an implicit decimal point and no decimal place.
5. Click **OK** to exit the Element Properties dialog box.

Creating the QTYORD Temporary Storage Element About this task

Use this procedure to create the QTYORD temporary storage element.

Procedure

1. Click the **UNITPRICE** temporary storage element and select **Edit > Insert > Element**.
2. In the Name box, type **QTYORD**.
3. In the Description box, type **Quantity Ordered**.
4. On the **Validation** tab, do the following:
 - In the Maximum box, type **9** as the maximum length.
 - From the data-type list, select **Number**.
 - From the Format list, select **R9** to indicate that the element will be formatted with an explicit decimal point and contain up to nine places.
5. Click **OK** to create the QTYORD element.

Creating an Extended Rule to Extract the Unit Price

Use an extended rule to extract the unit price from the QTY:3 segment and move it to the QTYORD temporary storage element. After the system reads the QTY:301 element (Quantity Qualifier), it executes the extended rule.

About this task

Use this procedure to create the extended rule.

Procedure

1. Double-click the **QTY:301** element to display the Element Properties dialog box.
2. Select the **Extended Rule** tab.
3. In the Rule box, type the following:

```
IF #6063 = "21" THEN
BEGIN
    $2500_LIN.#QTYORD = #6060;
END
```

The system verifies that the QTY:301 element does contain 21 (quantity ordered) and executes this rule.

4. Click **Compile** to validate the syntax of the extended rule.
5. Click **OK** to add the extended rule to the QTY:301 element.

Mapping the Quantity Ordered

About this task

Use this procedure to map the quantity ordered.

Procedure

Link the **QTYORD** temporary storage element in the **TEMPLINEITEM** temporary storage segment (in the **LIN** group) to the application field **QTYORD** in the **PODETAIL** record.

PODESC Record

About the PODESC Record

The PODESC record contains descriptive information about the line item ordered. The PODESC record can occur up to five times per **PODetailGroup**, so multiple descriptive terms can be used for each item. For example, if the line item is a dog collar, the description fields may be red, large, and leather.

Mapping the CUSTNUMDESC Field

For this map, you must use information from the partner definition to populate the Customer Number fields on the application side of the map. Since the Customer Number is a part of the record key for each record, it appears multiple times in the application file format, once in each record (**CUSTNUMHDR**, **CUSTNUMDET**, **CUSTNUMDESC**, **CUSTNUMAC**, and **CUSTNUMSUM**).

About this task

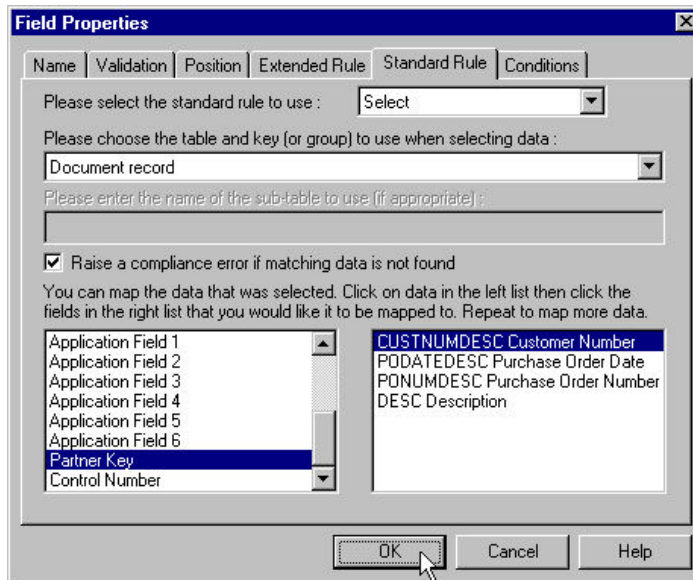
Your partner's customer number is part of the record key for each record in the application file format, identifying which trading partner sent the purchase order. This customer number is part of the definition for this partner, so you can map it from the partner definition to each customer number field.

Use this procedure to map the customer number from the partner definition.

Procedure

1. Double-click the **CUSTNUMDESC** field (in the **PODESC** record) to display the Field Properties dialog box.
2. Select the **Standard Rule** tab.
3. From the standard rule list, choose **Select**.
4. From the Table and Key list, select **Document record**.
5. Select the check box to indicate that a compliance error should be generated if the select does not find a valid entry.
6. From the Map From list on the left, select **Partner Key**.
7. From the Map To list on the right, select **CUSTNUMDESC**.

The Standard Rule tab of the Field Properties dialog box should now look like this:



8. Click OK to add the standard rule to the CUSTNUMDESC field.

Mapping the PODATEDESC Field

For ANSI X12, the BEG05 (Purchase Order Date) element contains the date of the purchase order. The BEG segment does not repeat. You need to map the information from the BEG05 element to the PODATEDESC field. However, the PODATEDESC field is included in the PODetailGroup, which repeats 100,000 times.

About this task

For EDIFACT, the date information is contained in the DTM segment. The DTM segment contains date information for the header part of the map, and repeats multiple times. If the DTM01 (Date/Time/Period Qualifier) is 4, that iteration of the DTM segment contains the order date/time. You must map the order date/time to the PODATE field in each the of the application records (PODATEHDR, PODATEDET, PODATEDESC, PODATEAC, and PODATESUM). You must map the information from the PODATE temporary storage element to the PODATEDESC field. The PODATEDESC field is included in the PODESC segment in the PODetailGroup, which repeats 200,000 times.

For TRADACOMS, the purchase order date information is contained in the ORD segment. The ORD segment contains order reference information for the header part of the map, and repeats multiple times. You need to map the date order placed by customer (ORD ORNO ORN2) to the PODATE field in each of the application records (PODATEHDR, PODATEDET, PODATEDESC, and PODATESUM).

To map the purchase order date to this application field, you need to use an extended rule to extract the purchase order date from the element and move it to the PODATEDESC field.

Use this procedure to create the extended rule.

Procedure

1. Double-click the **PODATEDESC** field to display the Field Properties dialog box.
2. Select the **Extended Rule** tab.
3. In the Rule list, type one of the following:
 - For ANSI X12:
`#PODATEDESC=$850.#0323;`
 - For EDIFACT:
`#PODATEDESC=$ORDERS.#PODATE;`
 - For TRADACOMS:
`#PODATEDESC = $ORDERS.#ORN2;△`
The `$ORDERS.#ORN2` is the addressing for the ORD ORNO ORD2 element.
The system executes this rule when it processes the **PODATEDESC** field.
4. Click **Compile** to validate the syntax of the extended rule.
5. Click **OK** to add the extended rule to the **PODATEDESC** field.

Mapping the PONUMDESC Field

For ANSI X12, the BEG03 (Purchase Order Number) element contains the purchase order number. The BEG segment does not repeat. You need to map the information from the BEG03 element to the **PONUMDESC** field. However, the **PONUMDESC** field is included in the **PODetailGroup**, which repeats 100,000 times.

About this task

For EDIFACT, the BGM05 (Document/Message Number) element contains the purchase order number. The BGM segment does not repeat. You must map the information from the BGM05 element to the **PONUMDESC** field. The **PONUMDESC** field is included in the **PODetailGroup**, which repeats 200,000 times.

For TRADACOMS, the ORD ORNO ORN0 element contains the customer's order number. The ORD segment does not repeat. You need to map the information from the ORD ORNO ORN0 element to the **PONUMDESC** field. However, the **PONUMDESC** field is included in the **PODetailGroup**, which repeats 999,999 times.

To map the purchase order number to this application field, you need to use an extended rule to extract the purchase order number from the element and move it to the **PONUMDESC** field.

Use this procedure to create the extended rule.

Procedure

1. Double-click the **PONUMDESC** field to display the Field Properties dialog box.
2. Select the **Extended Rule** tab.
3. In the Rule list, type one of the following:
 - For ANSI X12:
`#PONUMDESC=$850.#0324;`
 - For EDIFACT:
`#PONUMDESC=$ORDERS.#1004;`
 - For TRADACOMS:
`#PONUMDESC = $ORDERS.#ORN0;△`

The \$ORDERS.#ORN0 is the addressing for the ORD ORNO ORN0 element.
The system executes this rule when it processes the PONUMDESC field.

4. Click **Compile** to validate the syntax of the extended rule.
5. Click **OK** to add the extended rule to the PONUMDESC field.

DESC Field

Mapping the DESC Field for ANSI X12 and EDIFACT

The DESC field is available so your partner can describe the item in a free-form format.

About this task

Use this procedure to map the DESC field and to change the maximum usage of the PID group.

Procedure

1. Do one of the following:
 - For ANSI X12, link the **PID:205** element in the PID:2 segment (in the PID group) to the application field **DESC** in the PODESC record.
 - For EDIFACT, link the **IMD:206** element in the IMD:2 segment (in the LIN group) to the application field **DESC** in the PODESC record.

Note: You need to change the maximum usage of the segment, because you know that your partner will only send you five iterations of the segment.

2. Right-click on one of the following and select **Properties**:
 - For ANSI X12: **PID** group (in the P01 group)
 - For EDIFACT: **IMD:2** segment (LIN group)
3. On the **Looping** tab, type 5 in the Max Usage box. This box is currently set to 1000
4. Click **OK** to exit the EDI Segment Properties dialog box.

Mapping the DESC Field for TRADACOMS

The DESC field is available so your partner can describe the item in a free-form format. You can map the DESC field (description) by using an extended rule to concatenate the OLD:2 TDES TDE0 (Traded Unit Descr. Line 1) and OLD:2 TDES TDE1 (Traded Unit Descr. Line 2) elements, and map the concatenated string to the DESC field in the PODESC record.

About this task

Use this procedure to create the extended rule.

Procedure

1. Double-click the **DESC** field to display the Field Properties dialog box.
2. Select the **Extended Rule** tab.
3. In the Rule list, type the following:

```
#DESC = $OLD.#TDE0+" "+$OLD.#TDE1;
```

The \$OLD.#TDE0 is the addressing for the OLD:2 TDES TDE0 element. The + concatenates the character strings. The quotation marks indicate a blank space to distinguish the two concatenated lines of text. The \$OLDE.#TDE1 is the

addressing for the OLD:2 TDES TDE1 element. The system executes this rule when it processes the PONUMDET field.

4. Click **Compile** to validate the syntax of the extended rule.
5. Click **OK** to add the extended rule to the PONUMDESC field.

POALLCHG Record for ANSI X12 and EDIFACT

About the POALLCHG Record

The POALLCHG record contains the allowance or charge indicator, method of handling code, and total allowance or charge amount. The POALLCHG record can occur up to five times per PODetailGroup, so multiple allowances and/or charges can be used for each item.

Mapping the CUSTNUMAC Field

For this map, you need to pull information from the partner definition to populate the Customer Number fields on the application side of the map. Since the Customer Number is a part of the record key for each record, it appears five times in the application file format, once in each record (CUSTNUMHDR, CUSTNUMDET, CUSTNUMDESC, CUSTNUMAC, and CUSTNUMSUM).

About this task

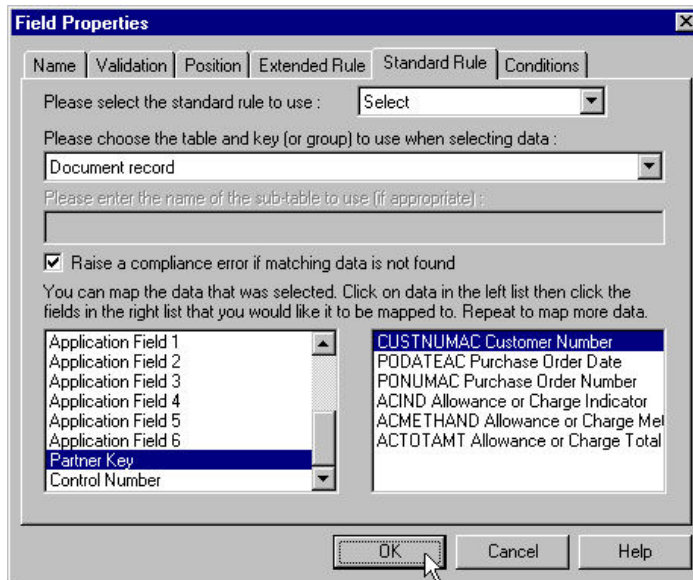
Your partner's customer number is part of the record key for each record in the application file format. It identifies which trading partner sent the purchase order. This customer number is already part of the partner definition for this partner, so you can map it from the partner definition to each customer number field.

Use this procedure to map the customer number from the partner definition.

Procedure

1. Double-click the **CUSTNUMAC** field (in the POALLCHG record) to display the Field Properties dialog box.
2. Select the **Standard Rule** tab
3. From the Standard Rule list, choose **Select**.
4. From the Table and Key list, select **Document record**.
5. Select the check box to indicate that a compliance error should be generated if the select does not find a valid entry.
6. From the Map From list on the left, select **Partner Key**.
7. From the Map To list, select **CUSTNUMAC**.

The Standard Rule tab of the Field Properties dialog should now look like this:



8. Click **OK** to add the standard rule to the CUSTNUMAC field.

Mapping the PODATEAC Field

To map the purchase order date to the application field, you need to use an extended rule to extract the purchase order date from the element and move it to the PODATEAC field.

About this task

For ANSI X12, the BEG05 (Purchase Order Date) element contains the date of the purchase order. The BEG segment does not repeat. You need to map the information from the BEG05 element to the PODATEAC field. However, the PODATEAC field is included in the PODetailGroup, which repeats up to 200,000 times.

For EDIFACT, the date information is contained in the DTM segment. The DTM segment contains date information for the header part of the map, and repeats multiple times. If the DTM01 (Date/Time/Period Qualifier) is 4, that iteration of the DTM segment contains the order date/time. You need to map the order date/time to the PODATE field in each the of the application records (PODATEHDR, PODATEDET, PODATEDESC, PODATEAC, and PODATESUM). You must map the information from the PODATE temporary storage element to the PODATEAC field. The PODATEAC field is included in the POALLCHG segment in the PODetailGroup, which repeats up to 200,000 times.

Use this procedure to create the extended rule:

Procedure

1. Double-click the **PODATEAC** field to display the Field Properties dialog box.
2. Select the **Extended Rule** tab.
3. In the Rule list, type one of the following:
 - For ANSI X12:
#PODATEAC=\$850.#0323;
 - For EDIFACT:

```
#PODATEAC = $ORDERS.#PODATE;
```

The system executes this rule when it processes the PODATEAC field.

4. Click **Compile** to validate the syntax of the extended rule.
5. Click **OK** to add the extended rule to the PODATEAC field.

Mapping the PONUMAC Field

To map the purchase order number to this application field, you need to use an extended rule to extract the purchase order number from the element and move it to the PONUMAC field.

About this task

For ANSI X12, the BEG03 (Purchase Order Number) element contains the purchase order number. The BEG segment does not repeat. You need to map the information from the BEG03 element to the PONUMAC field. However, the PONUMAC field is included in the PODetailGroup, which repeats 200,000 times.

For EDIFACT, the BGM05 (Document/Message Number) element contains the purchase order number. The BGM segment does not repeat. You need to map the information from the BGM05 element to the PONUMAC field. However, the PONUMAC field is included in the PODetailGroup, which repeats 200,000 times.

Use this procedure to create the extended rule.

Procedure

1. Double-click the **PONUMAC** field to display the Field Properties dialog box.
2. Select the **Extended Rule** tab.
3. In the Rule list, type one of the following:
 - For ANSI X12:

```
#PONUMAC=$850.#0324;
```
 - For EDIFACT:

```
#PONUMAC = $ORDERS.#1004;
```

The system executes this rule when it processes the PONUMAC field.

4. Click **Compile** to validate the syntax of the extended rule.
5. Click **OK** to add the extended rule to the PONUMAC field.

ACIND Field

Loading a Code Table for EDIFACT

On the EDI side of the map, the line item information is contained in the ALC group (in the ALC:2 and MOA:8 segments). The ALC:2 segment is contained in the ALC group, which is subordinate to the LIN group. You can use a code list table to define which codes will be accepted for the allowance or charge indicator in the ALC:201 element. The allowable codes are A (Allowance) and C (Charge). Then map the allowance or charge indicator from the ALC:2 segment directly to the POALLCHG application record, which is subordinate to the PODetailGroup.

About this task

Use this procedure to load a code list table from the standard.

Procedure

1. Double-click the **ALC:201** element to display the Element Properties dialog box.
2. On the **Standard Rule** tab, select **Use Code** from the Standard Rule list.
3. Click **Edit**.

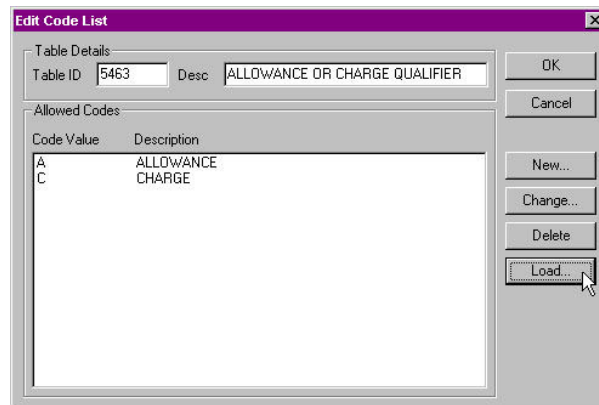
The Table ID contains the element for which this code list table is used. The Description box contains the element description for which this code list table is used.

4. In the Edit Code List dialog box, do the following:
 - Click **Load** to display the Select Data Source Name dialog box.
 - Select the **Machine Data Source** tab.
 - Select the Data Source Name that the system uses to access the EDI message standards (probably **Gentran Standards**).
 - Click **OK**.

The system displays the Load Code List dialog box.

5. In the Codes in Standard list, select both **A Allowance** and **C Charge**. Click **Add** to move both codes to the Codes Selected list.
6. Click **OK** to load the selected codes and exit the Load Code List dialog box.

The Edit Code List dialog box should now look like this.



7. Click **OK** to exit the Edit Code List dialog box.

On the Standard Rule tab, verify that the code list contains **5463 ALLOWANCE OR CHARGE QUALIFIER**.
8. Select the raise a compliance error check box to specify that, for compliance reasons, the element must contain one of two codes that you loaded from the code table list.
9. Click **OK** to exit the Element Properties dialog box.

Mapping the ACIND Field

Link the **ALC:201** element in the ALC:2 segment (in the ALC group in the LIN group) to the application field **ACIND** in the POALLCHG record.

About this task

If the system prompts you with a message box stating that the maximum usage of the output map component is smaller than the input map component, click **Yes** to acknowledge the message. You know that you will not lose data because your partner will not send more than five iterations of the ALC group. However, if your partner decides to send more than five ALC groups, you will need to change the maximum usage of the POALLCHG segment on the output side of the map to

match that number, or you would be at risk for losing data.

Mapping the ACMETHAND Field

About this task

Use this procedure to map the ACMETHAND field.

Procedure

- For ANSI X12, ACMETHAND is the allowance or charge method of handling code.

Link the **ITA:204** element in the ITA:2 segment to the application field **ACMETHAND** in the POALLCHG record.

- For EDIFACT, the line item information is contained in the ALC group (in the ALC:2 and MOA:8 segments). The ALC:2 segment is contained in the ALC group, which is subordinate to the LIN group. You can map the allowance or charge method of handling code directly from the ALC:204 element in the ALC:2 segment to the POALLCHG application record, which is subordinate to the PODetailGroup.

Link the **ALC:204** element in the ALC:2 segment (ALC group, LIN group) to the application field **ACMETHAND** in the POALLCHG record.

Note: If the system prompts you with a message box stating that the maximum usage of the output map component is smaller than the input map component, click **Yes** to acknowledge the message. You know that you will not lose data because your partner will not send more than five iterations of the ALC group. However, if your partner decides to send more than five ALC groups, you will need to change the maximum usage of the POALLCHG segment on the output side of the map to match that number, or you would be at risk for losing data.

ACTOTAMT Field

Changing the MOA Properties for EDIFACT

On the EDI side of the map, the line item information is contained in the ALC group (in the ALC:2 and MOA:8 segments). The MOA:8 segment is contained in the MOA group, which is subordinate to the ALC group, which is subordinate to the LIN group. You must map the allowance or charge total amount from the MOA:8 segment to the POALLCHG application record, which is subordinate to the PODetailGroup. The MOA group repeats up to two times. Because your partner will only send you one iteration of the MOA group, you must change the maximum usage for the group and promote the records in the group to the parent level.

About this task

You also know that the only Monetary Amount Type Qualifier (MOA:801 element) that your partner will send you is 128, which specifies that iteration of the MOA group contains the total amount of the allowance or charge. Since 128 is the only qualifier that your partner will send, you do not have to perform any mapping operations with this element.

Use this procedure to change the properties of the MOA group.

Procedure

1. Right-click the **MOA** group (ALC group, LIN group) and select **Properties**.
2. Select the **Looping** tab.
3. In the Maximum Usage box, type **1** to make the MOA group a non-repeating group.
4. Select the **Promote records to parent** check box to specify that when this group is compiled, the subordinate segments will be extracted and placed in the parent group. This locates the MOA:8 segment at the same level as the POALLCHG record on the application side of the map.
5. Click **OK** to exit the Group Properties dialog box.

Mapping the ACTOTAMT Field

About this task

Use this procedure to map the ACTOTAMT field.

Procedure

- For ANSI X12, ACTOTAMT is the allowance or charge total amount. Link the **ITA:207** element in the ITA:2 segment to the application field **ACTOTAMT** in the POALLCHG record.
- For Edifact, link the **MOA:802** element in the MOA:8 segment (ALC group, LIN group) to the application field **ACTOTAMT** in the POALLCHG record.

Note: If the system prompts you with a message box stating that the maximum usage of the output map component is smaller than the input map component, click **Yes** to acknowledge the message. You know that you will not lose data because your partner will not send more than five iterations of the ALC group. However, if your partner decides to send more than five ALC groups, you will need to change the maximum usage of the POALLCHG segment on the output side of the map to match that number, or you would be at risk for losing data.

Chapter 7. POSUMMARY Record

About the POSUMMARY Record

The following topics describe the actions necessary for you to map to each field in the POSUMMARY (summary) record. The POSUMMARY record contains the total number of line items in the purchase order. The POSUMMARY record occurs once for each purchase order translated.

Mapping the CUSTNUMSUM Field

You need to pull information from the partner definition to populate the Customer Number fields on the application side of the map. Since the Customer Number is a part of the record key for each record, it appears five times in the application file format, once in each record (CUSTNUMHDR, CUSTNUMDET, CUSTNUMDESC, CUSTNUMAC, and CUSTNUMSUM).

About this task

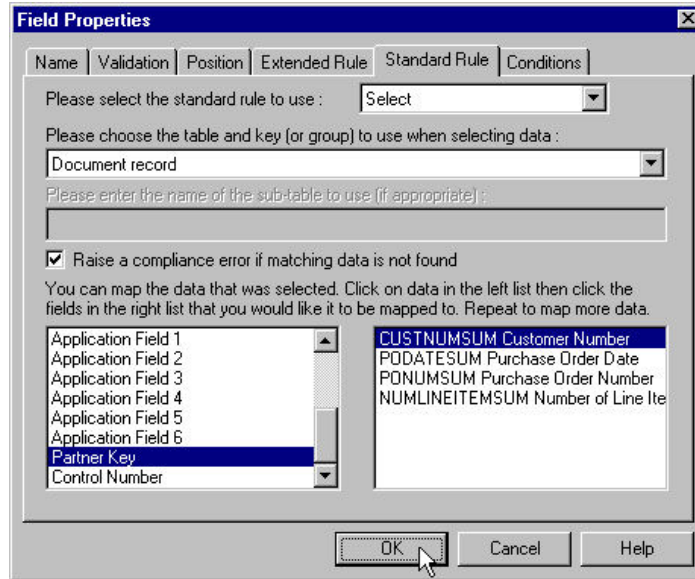
Your partner's customer number is part of the record key for each record in the application file format. It identifies which trading partner sent the purchase order. This customer number is already part of the partner definition for this partner, so you can map it from the partner definition to each customer number field.

Use this procedure to map the customer number from the partner definition.

Procedure

1. Double-click the **CUSTNUMSUM** field (in the POSUMMARY record) to display the Field Properties dialog box.
2. Select the **Standard Rule** tab.
3. From the standard rule list, choose **Select**.
4. From the Table and Key list, select **Document record**.
5. Select the check box to indicate that a compliance error should be generated if the select does not find a valid entry.
6. From the Map From list on the left, select **Partner Key**.
7. From the Map To list on the right, select **CUSTNUMSUM**.

The Standard Rule tab of the Field Properties dialog box should now look like this:



8. Click OK to add the standard rule to the CUSTNUMSUM field.

Mapping the PODATESUM Field

About this task

Use this procedure to map the PODATESUM field.

Procedure

- For ANSI X12, link the **BEG05** element in the BEG record to the application field **PODATESUM** in the POSUMMARY record.
- For EDIFACT, the date information is contained in the DTM segment. The DTM segment contains date information for the header part of the map, and repeats multiple times. If the DTM01 (Date/Time/Period Qualifier) is 4, that iteration of the DTM segment contains the order date/time. You need to map the order date/time to the PODATE field in each the of the application records (PODATEHDR, PODATEDET, PODATEDESC, PODATEAC, and PODATESUM). To map the correct information from the DTM segment to the PODATESUM field in the application header record, you must determine the data type and extract the order date/time.

Link the **PODATE** temporary storage element in the TEMPDATES segment to the application field **PODATESUM** in the POSUMMARY record.

Note: PODATE is already linked to the PODATEHDR in the PODATE record. It is acceptable to have multiple links to an element.

- For TRADACOMS, the date information is contained in the ORD segment. The ORD segment contains order reference information for the header part of the map, and repeats multiple times. You need to map the date order placed by customer (ORD ORNO ORN2) to the PODATE field in each of the application records (PODATEHDR, PODATEDET, PODATEDESC, and PODATESUM). The ORD ORNO ORN2 element is already directly linked to the PODATEHDR field in the POHEADER record. You will also map the ORD ORNO ORN2 element directly to the PODATESUM application field.

Link the **ORD ORNO ORN2** element in the BEG record to the application field **PODATESUM** in the POSUMMARY record.

Mapping the PONUMSUM Field

About this task

Use this procedure to map the PONUMSUM field.

Procedure

- For ANSI X12, link the **BEG03** element in the BEG record to the application field **PONUMSUM** in the POSUMMARY record.
- For EDIFACT, link the **BGM05** element in the BGM record to the application field **PONUMSUM** in the POSUMMARY record. Your partner will only send you up to 22 characters for the BGM05 element (Document/Message Number), so you have already changed the maximum length allowed for that element. Also, the BGM05 element is already mapped (linked) to the PONUMHDR field in the POHEADER application record, so you will be adding an additional link from this element to the PONUMSUM field in the POSUMMARY record.
- For TRADACOMS, link the **ORD ORNO ORN0** element in the BEG record to the application field **PONUMSUM** in the POSUMMARY record. The ORD ORNO ORN0 element is already mapped (linked) to the PONUMHDR field in the POHEADER application record, so you will add an additional link from this element to the PONUMSUM field in the POSUMMARY record.

Mapping the NUMLINEITEMSUM Field

About this task

Use this procedure to map the NUMLINEITEMSUM field.

Procedure

- For ANSI X12, link the **CTT01** element in the CTT record to the application field **NUMLINEITEMSUM** in the POSUMMARY record.

The CTT01 element is already directly linked to the NUMLINEITEMS field in the POHEADER record, but Sterling Gentran:Server allows you to link an input element to multiple output fields.

- For EDIFACT, link the **NUMLINEITEMS** temporary storage element in the TEMPCONTROL temporary storage segment to the application field **NUMLINEITEMSUM** in the POSUMMARY record.

On the EDI side of the map, the control total information is contained in the CNT segment. The CNT segment contains all control values and repeats multiple times. If the CNT01 (Control Qualifier) is 2, that iteration of the CNT segment contains the number of line items in the message. You must map the total number of line items to the NUMLINEITEMS field in the POHEADER application record and the NUMLINEITEMSUM field in the POSUMMARY application record. To map the correct information from the CNT segment to the NUMLINEITEMSUM field in the application summary record, you must extract the number of line items after determining the type of control value. You have already created a temporary storage segment (TEMPCONTROL) and element (NUMLINEITEMS) on the EDI side of the map that do not repeat. You used an extended rule to extract the number of line items from the CNT segment and move it to the appropriate temporary storage element. The NUMLINEITEMS temporary storage element is already mapped (linked) to the NUMLINEITEMS field in the POHEADER application record so you will be adding an additional link from this element to the NUMLINEITEMSUM field in the POSUMMARY record.

- For TRADACOMS, link the **OTR LORD** element in the CTT record to the application field **NUMLINEITEMSUM** in the POSUMMARY record.
The OTR LORD element is already mapped (linked) to the NUMLINEITEMS field in the POHEADER application record, so you will be adding an additional link from this element to the NUMLINEITEMSUM field in the POSUMMARY record.

Chapter 8. Finalize the Map

Compiling the Map for the Inbound PO

The Compile function compiles the map and generates a translation object. The map that you created using Sterling Gentran:Server is a source map. When that source map is compiled, the result is a compiled translation object.

About this task

This translation object must be registered with the Sterling Gentran:Server system before you can use it.

Use this procedure to compile a map and generate a translation object.

Procedure

1. Select **File > Save** to save the source map before compiling.
2. Select **File > Compile**.
3. The File name box automatically contains one of the following:
 - **pet_850.tpl** (for ANSI X12)
 - **pet_eord.tpl** (for EDIFACT)
 - **pet_tord.tpl** (for TRADACOMS)

This is the name of the translation object, using the default .TPL file extension. We recommend that you name the translation object (.TPL file) the same file name (1-8 characters long) as you named the map (.MAP file). Preserving the same file name (with different file extensions) means that the relationship between the source map and the compiled translation object remains evident.

Important: Do not overlay the source map with the compiled translation object. Use the .TPL file extension to distinguish the translation object.

4. Navigate to where the compiled translation object is stored, if necessary.

Important: Do not store the compiled translation object in the GENSRVNT\RegTransObj subfolder. This subfolder is reserved for storing a copy of each translation object you register with Sterling Gentran:Server.

5. Click **Save**.

The system compiles the map and generates a translation object. The Compile Error dialog box is displayed. Verify that no errors occurred (scroll down to the bottom of the list).

6. Click **OK**.

The date on which the translation object was compiled is automatically loaded into the Compiled on box on the Translation Object Details dialog box. See the results below.

7. Select **File > Save** to save the source map with the Compiled on date.

Note: You must register this translation object with the Sterling Gentran:Server system before you can use it.

Printing the Inbound PO Report

The Sterling Gentran:Server report enables you to validate and review the map and make modifications as needed. If you review the report and determine that the map is incorrect, you should refine the mapping process as many times as necessary (make the modifications, save the map, recompile the translation object, and print the report again).

About this task

Use this procedure to print the report for the inbound purchase order.

Procedure

1. Select **File > Print**.
2. Accept the defaults on this dialog box so the report includes all sections and only the activated map components.
3. Click **OK**.
The Print dialog box is displayed.
4. Click **OK** on the Print dialog box and the mapping report is printed.

Testing the Inbound PO Translation Object

After you compile the map, print and verify the report, and register the translation object with Sterling Gentran:Server, you should test the translation object to verify that the data is translated correctly.

About this task

To test the compiled translation object, you should obtain test data from your partners and process the data. You should also verify acknowledgement processing (if applicable) and verify communications with your trading partner.

Use this procedure to test the inbound (Export) translation object you just created. The default path is C:\GENSRVNT\TRANSOBJ\.

Procedure

1. Register the translation object with Sterling Gentran:Server.
2. Import the partner relationship into Sterling Gentran:Server.
3. Verify (in Partner Editor) that the export translation object is selected for the Inbound Relationship.
4. Use the Process File option in Sterling Gentran:Server to process the data file through the translation object. The data file is located in the TUTORIAL subfolder under the folder where Sterling Gentran:Server is installed.
After the document is translated, it is located in the In Documents in Sterling Gentran:Server.
5. Export the document to ensure that it was translated correctly.

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing

IBM Corporation

North Castle Drive

Armonk, NY 10504-1785

U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing

Legal and Intellectual Property Law

IBM Japan Ltd.

19-21, Nihonbashi-Hakozakicho, Chuo-ku

Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be

incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

J46A/G4

555 Bailey Avenue

San Jose, CA 95141-1003

U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© IBM 2012. Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. 2012.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.ibm.com)[®] are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java™ and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Linear Tape-Open, LTO, the LTO Logo, Ultrium and the Ultrium Logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Connect Control Center®, Connect:Direct®, Connect:Enterprise®, Gentran®, Gentran®:Basic®, Gentran:Control®, Gentran:Director®, Gentran:Plus®, Gentran:Realtime®, Gentran:Server®, Gentran:Viewpoint®, Sterling Commerce™, Sterling Information Broker®, and Sterling Integrator® are trademarks or registered trademarks of Sterling Commerce®, Inc., an IBM Company.

Other company, product, and service names may be trademarks or service marks of others.



Product Number: 5725-D09

Printed in USA