Gentran:Server® for Windows®

Application Integration
ANSI Tutorial Guide

Version 5.1

Sterling Commerce
An IBM Company

Copyright Notice

Table of Contents

Glossary

Getting Started

Contents

Overview

Introduction

**Application
integration
subsystem**

Welcome to Gentran:Server® for Windows® Application Integration subsystem, Sterling Commerce's electronic commerce (EC) mapping software for the Windows operating system. Gentran:Server for Windows provides you with the tools you need 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).

In this Guide

Overview

This guide contains a tutorial for the Gentran:Server for Windows Application Integration subsystem, using the ANSI X12 standard. This tutorial includes an inbound mapping example (purchase order) and an outbound mapping example (invoice). The purpose of this tutorial is to break the mapping process down into concise and understandable steps, using examples that enable you to understand and master the tasks and processes required to create your own maps.

To access reference information, refer to the Gentran:Server for Windows Application Integration online Help or *Application Integration User's Guide*. For additional information on the various tasks described in this guide, refer to the Gentran:Server for Windows *User's Guide*.

Note

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 Gentran:Server for Windows product.

Contents

The *Tutorial Guide* is organized into chapters. This table provides a brief description of each chapter's contents.

Chapter	Description
1 Getting Started	Explains the content, organization, and conventions in this guide, as well as the system requirements. This chapter also describes a brief outline of how to build maps using Gentran:Server for Windows Application Integration, how to customize global display options, and gives you an introduction to mapping and Gentran:Server for Windows.
2 Inbound Purchase Order	Uses a step-by-step approach to explain the tasks and procedures necessary to create an export map for an ANSI inbound (850) purchase order. *(Continued on next page)*

Chapter	Description
3 Outbound Invoice	Uses a step-by-step approach to explain the tasks and procedures necessary to create a system import map and an import map for an ANSI outbound (810) invoice.
Glossary	Contains definitions of mapping and Gentran:Server for Windows Application Integration subsystem terms and concepts.

Introducing Mapping

Mapping Overview

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

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

Outbound mapping   For outbound mapping, the system translates your application 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 Gentran:Server for Windows 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. Please see *Creating a System Import Map* on page 225 for more information on creating a system import map.

Translation objects   The map you create is the *source* from which you compile a translation object. A translation object is a set of rules that instruct the translator how 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 Gentran:Server for Windows EDI translation. Translation objects can belong to the following two categories:

◗   system translation objects or

◗   user translation objects.

Translation objects (contd)

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 the required system translation objects are automatically installed with the Gentran:Server for Windows system.

Gentran:Server for Windows Application Integration enables you to generate import, export, and document turnaround translation objects. The Gentran:Server for Windows Forms Integration subsystem enables you to generate data entry and print translation objects. Once you generate a translation object using the Gentran:Server for Windows Application Integration subsystem (or the Gentran:Server for Windows Forms Integration subsystem), you must register the translation object with Gentran:Server for Windows. In addition, you must establish correct trading 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.

Reference
See the *Gentran:Server for Windows User's Guide* for more information about translation objects, trading partner relationships, and naming translation objects.

Diagram: Inbound and outbound translation process

This diagram illustrates the inbound and outbound translation process.
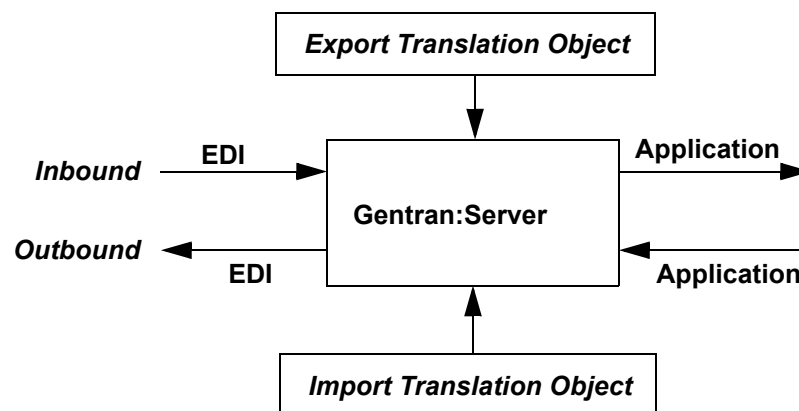
When you create an import map, you must define to Gentran:Server for Windows your application file format and the EDI standard format in which your partners expect to receive the documents. When you create an export map, you need to define to Gentran:Server for Windows the EDI standard format in which your partners sends documents and your application file format (how the data needs to be formatted for your application to process it).

File Formats

| Overview | The Application File Format and EDI File Format are described in this section. |

Application file format

If you are creating an import or export map, you must define your application to the Gentran:Server for Windows Application Integration subsystem. In Gentran:Server for Windows 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 either need to extract from your partner's document (if the map is inbound) or need to send to your partner (if the map is outbound).

Map components

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 as illustrated to the left.
Record		Contains a group of related fields. A record can occur once or can repeat multiple times. A record is visually represented in the Application Integration subsystem as illustrated to the left.
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. A field is visually represented in the Application Integration subsystems as illustrated to the left.

Map components (contd)

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 rule a black asterisk appears to the right of the element 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 Gentran:Server for Windows Application Integration subsystem generates an EDI file for you, based on the standard (agency), version, transaction set, and release (for the TRADACOMS standard *only*) 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. Please see *Creating a New Map* on page 18 for more information on generating an EDI file when you create a new map. 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 Gentran:Server for Windows 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.

Specialized EDI standard versions

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

Map components

This table describes the map components Gentran:Server for Windows 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 as illustrated to the left.
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. A segment is visually represented in the Application Integration subsystem as illustrated to the left.
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). A composite is visually represented in the Application Integration subsystem as illustrated to the left.
Element		This is the smallest piece of information defined by the EDI standards. An individual element can have different meanings depending on the context. Elements are typically not considered to have useful meaning until they are combined into segments. 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. An element is visually represented in the Application Integration subsystem as illustrated to the left.

Map components (contd)

Notes

◗   When an element has a mapping operation performed against it, a red checkmark is displayed over the element icon.

◗   When an element contains a standard rule a black asterisk appears to the right of the element icon.

Reference

Please see the *Gentran:Server for Windows Application Integration User's Guide* for more information on customizing the properties of the map components, and using specialized Gentran:Server for Windows functions to manipulate the EDI file structure.

The Application Integration Subsystem

Overview

Introduction   The Gentran:Server for Windows 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).

Navigation   You can access the Gentran:Server for Windows Application Integration Window in one of four ways:

◗   Click the menu option from the Main Menu Bar.

◗   Click the appropriate button on the Main Toolbar.

◗   Click the appropriate part of the map.

◗   Click a map component with the right mouse button to access a shortcut menu, which contains all the functions that are currently available for that map component. The content of the shortcut menus varies, depending on the type and level of the selected map component.

Note
When you start Gentran:Server for Windows Application Integration, the Main Menu Bar contains a subset of menu items. A full set of menu items is displayed after you create a new map or open (load) an existing map.

Application Integration Window

Application integration window

This diagram illustrates the Gentran:Server for Windows - Application Integration Window.
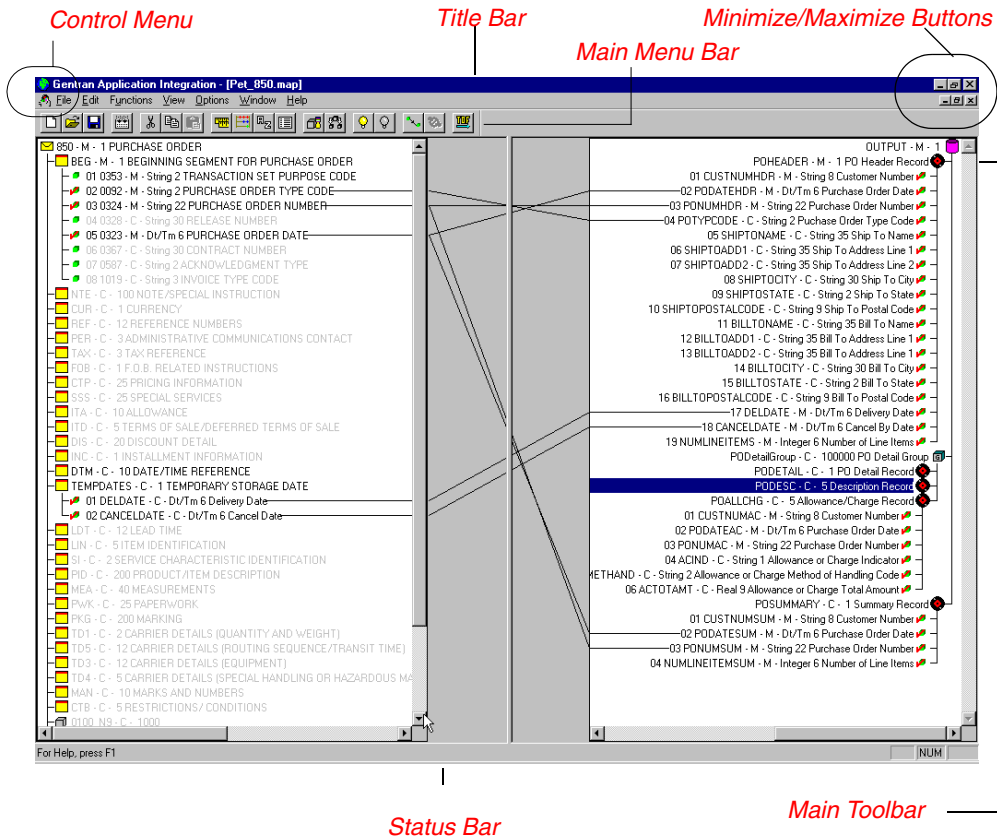
Application integration window components

This table describes the Application Integration Window components.

Part	Function
Control Menu	Contains the following commands: ◗ Restore ◗ Move ◗ Size ◗ Minimize ◗ Maximize ◗ Close *(Continued on next page)*

Part	Function
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 the drop-down menus. Unavailable items are dimmed. **Reference** A detailed description of each command can be found in the online Help.
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 modes for typing (e.g., CAP for the "Caps Lock" key).

Customizing Global Display Options

Overview

Introduction

The Gentran:Server for Windows 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. This section explains how to perform the following:
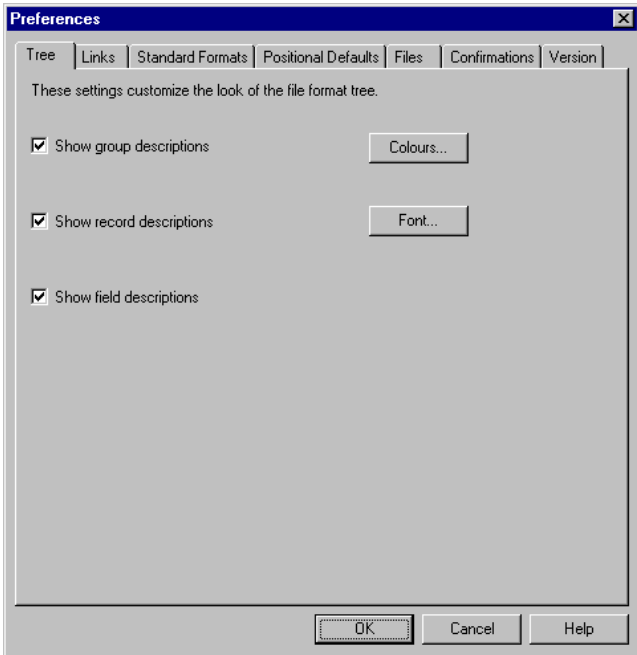
◗   Customizing global display options

◗   Displaying links (global option)

◗   Setting global confirmation options

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

Customizing global display options

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

Step	Action
1	From the Options menu, select Preferences. The Preferences dialog box enables you to set global defaults for Gentran:Server for Windows. **System Response** The Preferences dialog box is displayed. The Tree tab is displayed by default. (Continued on next page)
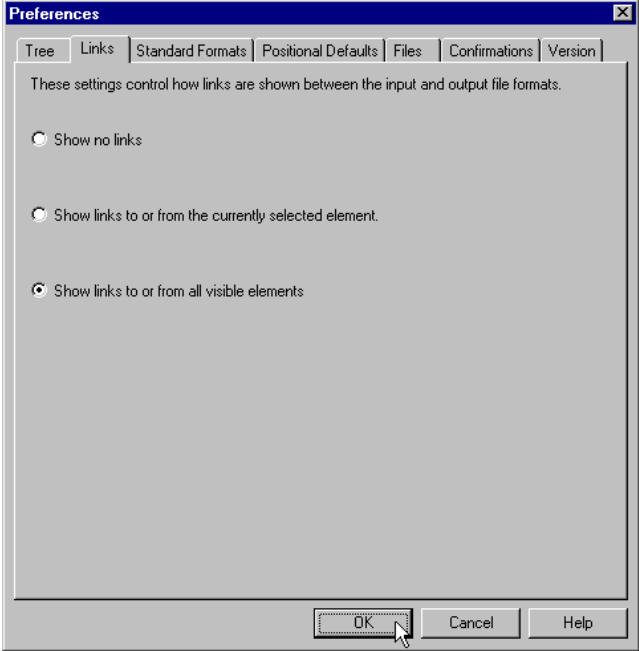
Step	Action
2	To turn on the default display of group, record (segment), and field (element) descriptions, select the appropriate options. **Tip** You typically want all descriptions displayed for reference. But depending on the size of your monitor, it may be easier to see the entire map if the descriptions are not displayed. **Note** You may want to experiment with shrinking the size of the font for the map (see *Customizing Global Fonts* in the Gentran:Server for Windows *Application Integration User's Guide*) 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.

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

Step	Action
1	From the Options menu, select Preferences. **System Response** The Preferences dialog box is displayed. The Tree tab is displayed by default. *(Continued on next page)*

Step	Action
2	Click the Links tab to display the linking options. The Links tab enables you to set the default display of mapping links.
3	Select a setting. ◗ 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 (this option enables you to concentrate on the selected field and removes the confusion of viewing many links at once. ◗ Show links to or from all visible elements: display all the mapping links The option you select is the default for all maps.
4	Click **OK** to save changes and exit the Preferences dialog box.

How to customize global confirmation options

The Confirmations tab on the Preferences dialog allows you specify when you want confirmation messages displayed.

Warning
Depending on how you set these options, you may receive either more or fewer confirmation messages than described in this tutorial.

Use this procedure to set the confirmation options.

Step	Action
1	From the Options menu, select **Preferences**. **System response** The system displays the Preferences dialog box (Tree tab displayed by default).
2	Select the **Confirmation** tab. **System response** The system displays the confirmation options. (Continued on next page)

(Contd) Step	Action
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. **System response** The system sets the confirmation options you selected.

Building a Map with Application Integration

Overview

Map-building process

This 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 Gentran:Server for Windows Application Integration subsystem, you should establish the default date format that the system uses. Please see the *Gentran:Server for Windows User's Guide* for more information on setting the global date default.
3	**Create, Save, and Name a New Map** Please see *Creating the Export Map* on page 2 - 17 for more information on creating, saving, and naming a map.
4	**Activate the Appropriate EDI Groups, Segments, and Elements** Please see *Activating EDI Map Components* on page 2 - 22 for more information on activation.
5	**Define your Application File Format** Please see *Defining Your Application File Format* on page 2 - 27 for more information on defining your application file format.
6	**Map the Appropriate Data for Each Application Field** Please see *Mapping Your Data* on page 2 - 54 for more information on mapping your data.
7	**Compile the Translation Object** Please see *How to Compile the Map* on page 2 - 108 for more information on compiling the translation object and translation object naming conventions.
8	**Print the Mapping Report** Validate and review the map, and make modifications as needed. Please see *How to Print the Report* on page 2 - 110 for more information on printing the mapping report. *(Continued on next page)*

(Contd) Stage	Description
9	**Register the Translation Object with Gentran:Server for Windows** Please see the *Gentran:Server for Windows User's Guide* for more information on registering translation objects.
10	**Create the Appropriate Trading Relationship** Establish the appropriate trading relationship in Gentran:Server for Windows for your trading partners. Please see the *Gentran:Server for Windows User's Guide* for more information on creating trading relationships. The trading relationship that you will use in this tutorial (PETZONE) is provided on the tutorial diskettes. You need to import the partner profile (PETZONE.PAR) into Gentran:Server for Windows, before you begin this tutorial. Please see the *Gentran:Server for Windows User's Guide* for more information on importing trading relationships.
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.

Using the Tutorial

Tutorial basics 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\PETTEST.INT
Outbound (system import and import)	PET_EINV.TXT	C:\GENSRVNT\TUTORIAL\PET_810.TXT

Inbound Purchase Order (850)

Overview

Introduction

In this chapter   This tutorial defines the tasks necessary to create a map used to translate ANSI X12 purchase orders (850) that you receive from your trading partner.

This chapter explains all the tasks and processes associated with creating a map for an inbound purchase order, including the following:

◗ Mapping analysis
◗ Creating the new map
◗ Activating groups, segments, and elements
◗ Defining your application file format
◗ Mapping your data
◗ Finalizing the map

Getting Started

Overview   This tutorial exercise 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 (although you may want to verify your completed map with the model).

The purpose of completing each task in this tutorial in sequential order as if it were your own scenario, is to teach you not only how to perform the individual mapping tasks, but also to teach you a logical approach and methods that should be utilized when you create your own map.

Inbound mapping assumptions and scenario   Make these assumptions in this tutorial exercise:

◗ 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.

Mapping analysis   The first step in creating a map is the analysis of the mapping requirements. This is the most important step in creating a successful map. If the analysis you perform 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, it is likely that creating the map is a much longer and arduous task, and the map may be invalid due to oversights and omissions.

This table describes the steps used in mapping analysis. These steps apply to 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	Begin creating the map.

Analyzing Your Application File Format

Overview   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 Gentran:Server for Windows. 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.

Application file layout   The application file layout for this tutorial is as follows.

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (POHEADER) 1 per document					
CUSTNUMHDR	X	Customer Number	String	1	8
PODATEHDR	X	Purchase Order Date	Dt/Tm (MMDDYY)	6	6
PONUMHDR	X	Purchase Order Number	String	1	22
POTYPCODE		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	30
SHIPTOSTATE		Ship To State	String	0	2
SHIPTOPOSTAL CODE		Ship To Postal Code	String	0	9
					(Continued on next page)

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
BILLTONAME		Bill To Name	String	0	35
BILLTOADD1		Bill To Address Line 1	String	0	35
BILLTOADD2		Bill To Address Line 2	String	0	35
BILLTOCITY		Bill To City	String	0	30
BILLTOSTATE		Bill To State	String	0	2
BILLTOPOSTAL CODE		Bill To Postal Code	String	0	9
DELDATE	X	Delivery Date	Dt/Tm (MMDDYY)	6	6
CANCELDATE	X	Cancel By Date	Dt/Tm (MMDDYY)	6	6
NUMLINEITEMS	X	Number of Line Items	Number (N0)	1	6
PO Detail Group (PODetailGroup) up to 100,000 per document					
Detail Record (PODETAIL) 1 per document					
CUSTNUMDET	X	Customer Number	String	1	8
PODATEDET	X	Purchase Order Date	Dt/Tm (MMDDYY)	6	6
PONUMDET	X	Purchase Order Number	String	1	22
CUSTPROCODE	X	Customer Product Code	String	1	30
UPCCODE	X	UPC Code	String	1	30
UOM		Unit of Measure	String	0	7
UNITPR		Unit Price	Number (N2)	0	14
QTYORD	X	Quantity Ordered	Number (N0)	1	9

(Continued on next page)

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Description Record (PODESC) up to 5 per document					
CUSTNUMDESC	X	Customer Number	String	1	8
PODATEDESC	X	Purchase Order Date	Dt/Tm (MMDDYY)	6	6
PONUMDESC	X	Purchase Order Number	String	1	22
DESC		Description	String	0	80
Allowance/Charge Record (POALLCHG) up to 10 per document					
CUSTNUMAC	X	Customer Number	String	1	8
PODATEAC	X	Purchase Order Date	Dt/Tm (MMDDYY)	6	6
PONUMAC	X	Purchase Order Number	String	1	22
ACIND		Allowance or Charge Indicator	String	0	1
ACMETHAND		Allowance or Change Method of Handling Code	String	0	2
ACTOTAMT		Allowance or Charge Total Amount	Number (N2)	0	9
Summary Record (POSUMMARY) 1 per document					
CUSTNUMSUM	X	Customer Number	String	1	8
PODATESUM	X	Purchase Order Date	Dt/Tm (MMDDYY)	6	6
PONUMSUM	X	Purchase Order Number	String	1	22
NUMLINEITEM SUM	X	Number of Line Items	Number (N0)	1	6

Application file definition

The above table contains the minimum information necessary to define your application file format to Gentran:Server for Windows (name, mandatory status, description, data type, and field length). 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

Record/group definitions

This table provides the Record/Group Definitions.

Record/Group	Definition
POHEADER (header)	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. This header record is mandatory and occurs once for each purchase order translated.
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 per order translated. This equals a maximum order of 100,000 items. The information within the PODetailGroup is further segregated by record.
PODETAIL (detail)	Contains line item code number, unit of measure, price, and quantity ordered. The PODETAIL record occurs once per PODetailGroup.
PODESC (description)	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. **Example** If the line item is a "dog collar," the description fields may be "red," "large," and "leather." *(Continued on next page)*

(Contd) Record/Group	Definition
POALLCHG (allowance and charge)	Contains the allowance or charge indicator, method of handling code, and total allowance or charge amount. The POALLCHG record can occur up to ten times per PODetailGroup, so multiple allowances and/or charges can be used for each item.
POSUMMARY (summary)	Contains the total number of line items in the purchase order. The POSUMMARY record occurs once for each purchase order translated.

Next step   Now that you have thoroughly analyzed your application file format, the next step is to determine the requirements of the purchase order that you receive from your partners.

Analyzing the EDI File

Overview

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 you are using. In this tutorial, we are using the ANSI X12 standard, version 003030, transaction set 850 (purchase order). 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 very 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.

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).

Map components

This table lists the map components that are required for this tutorial, but are not currently active.

Groups		Segment	Element
		DTM	DTM02
N1		N1	N102
		N3	N302
			N401
		N4	N402
			N403
PO1 (already activated)		PO1	PO102
			PO103
			PO104
			PO107
	PID	PID:2	PID:205
		ITA:2	ITA:207

**Map components
(contd)**

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

Recommendation
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 Gentran:Server for Windows. This allows you to differentiate between documents in the document browsers in Gentran:Server for Windows.

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. 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.

After you complete the analysis of the EDI file and your application, the final step in mapping analysis is correlating the two components.

Correlating Your Application File and the EDI File

Overview

To reconcile your application file format with the EDI file, you must identify each application file 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 Gentran:Server for Windows 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.

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

Mapping operations

This table describes the mapping operations (by application field) required for this tutorial.

Field Name	Description	Mapping Operations
Header Record (POHEADER) 1 per document		
CUSTNUMHDR	Customer Number	Extract Application ID from Partner record
PODATEHDR	Purchase Order Date	Link to BEG05
PONUMHDR	Purchase Order Number	Link to BEG03
RECTYPHDR	Record Type	HDR
POTYPCODE	Purchase Order Type Code	Link to BEG02
SHIPTONAME	Ship To Name	Link to N101 using "ST" (ShipTo) Code
SHIPTOADD1	Ship To Address Line 1	Link to N301 using "ST" (ShipTo) Code

(Contd) Field Name	Description	Mapping Operations
SHIPTOADD2	Ship To Address Line 2	Link to N302 using "ST" (ShipTo) Code
SHIPTOCITY	Ship To City	Link to N401 using "ST" (ShipTo) Code
SHIPTOSTATE	Ship To State	Link to N402 using "ST" (ShipTo) Code
SHIPTOPOSTAL CODE	Ship To Postal Code	Link to N403 using "ST" (ShipTo) Code
BILLTONAME	Bill To Name	Link to N101 using "BT"(BillTo) Code
BILLTOADD1	Bill To Address Line 1	Link to N301 using "BT"(BillTo) Code
BILLTOADD2	Bill To Address Line 2	Link to N302 using "BT"(BillTo) Code
BILLTOCITY	Bill To City	Link to N401 using "BT"(BillTo) Code
BILLTOSTATE	Bill To State	Link to N402 using "BT"(BillTo) Code
BILLTOPOSTAL CODE	Bill To Postal Code	Link to N403 using "BT"(BillTo) Code
DELDATE	Delivery Date	Link to DTM02 using "001" Code
CANCELDATE	Cancel By Date	Link to DTM02 using "002" Code
NUMLINEITEMS	Number of Line Items	Link to CTT01
Detail Record (PODETAIL) up to 200,000 per document		
CUSTNUMDET	Customer Number	Extract Application ID from Partner record
PODATEDET	Purchase Order Date	Extended rule to map from BEG05
PONUMDET	Purchase Order Number	Extended rule to map from BEG03
RECTYPDET	Record Type	DET

(Contd) Field Name	Description	Mapping Operations
CUSTPROCODE	Customer Product Code	Link to P0107
UPCCODE	UPC Code	Extract Partner xref by partner item from subtable "PRODCODE"
UOM	Unit of Measure	Link to P0103
UNITPR	Unit Price	Link to P0104
QTYORD	Quantity Ordered	Link to P0102
Description Record (PODESC) up to 5 per document		
CUSTNUMDESC	Customer Number	Extract Application ID from Partner record
PODATEDESC	Purchase Order Date	Extended rule to map from BEG05
PONUMDESC	Purchase Order Number	Extended rule to map from BEG03
RECTYPDESC	Record Type	DES
DESC	Description	Link to PID:205
Allowance/Charge Record (POALLCHG) up to 5 per document		
CUSTNUMAC	Customer Number	Extract Application ID from Partner record
PODATEAC	Purchase Order Date	Extended rule to map from BEG05
PONUMAC	Purchase Order Number	Extended rule to map from BEG03
RECTYPAC	Record Type	AC
ACIND	Allowance or Charge Indicator	Link to ITA:201
ACMETHAND	Allowance or Change Method of Handling Code	Link to ITA:204
ACTOTAMT	Allowance or Charge Total Amount	Link to ITA:207

(Contd) Field Name	Description	Mapping Operations
Summary Record (POSUMMARY) 1 per document		
CUSTNUMSUM	Customer Number	Extract Application ID from Partner record
PODATESUM	Purchase Order Date	Link to BEG05
PONUMSUM	Purchase Order Number	Link to BEG03
RECTYPSUM	Record Type	SUM
NUMLINEITEM SUM	Number of Line Items	Link to CTT01

Cross-Reference tables

You must determine whether or not you need to use code tables to convert data for the map. If you need to use tables, you must decide whether to create a code table in Gentran:Server for Windows that can only be used with this map, or add the table to either the partner relationship or the internal system partner in Partner Editor. In this tutorial, you need to map information from a cross-reference table to translate your partner's code for a purchased item and 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).

Reference
Please see the Gentran:Server for Windows *User's Guide* for more information on cross-reference tables.

This table describes the information in the cross-reference table included in this tutorial.

My Value	Partner Value	Description	Reference Data
12345678901234	555666411	Large Dog Collar	Red
23456789012345	555666817	Spring Breeze Cat Litter	10# 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	

Note
The cross-reference table is included in the tutorial data with the PETZONE partner (PETZONE.PAR)

After you finish correlating your application file format and EDI file on paper, you can begin creating the map.

The Export Map
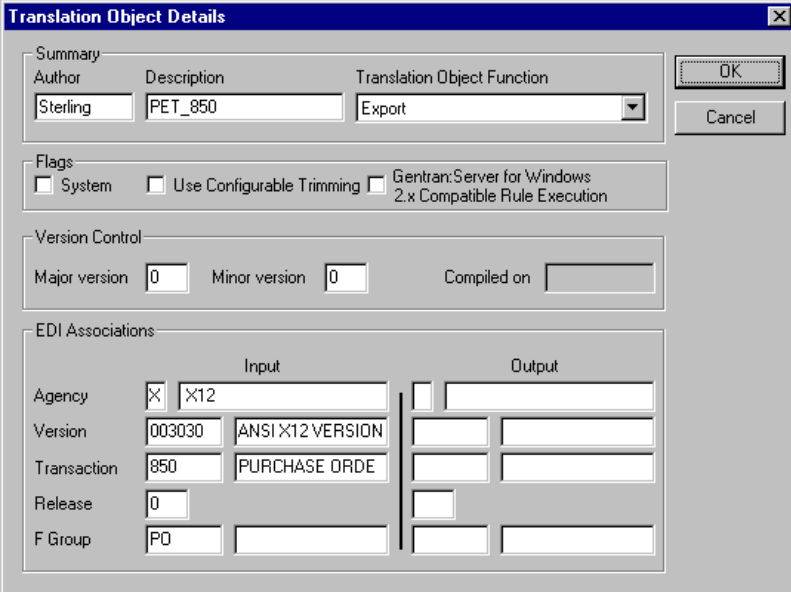
Creating the Export Map

Overview   The first task in creating a new map is to define the format that is used for the input and output sides of the map. In this example, you are creating an export map so Gentran:Server for Windows can translate a purchase order that your partner sends to your application file. The input side of the map must reflect the ANSI X12 850 purchase order that you receive. Instead of defining the 850 document to the system yourself, you want Gentran:Server for Windows 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.
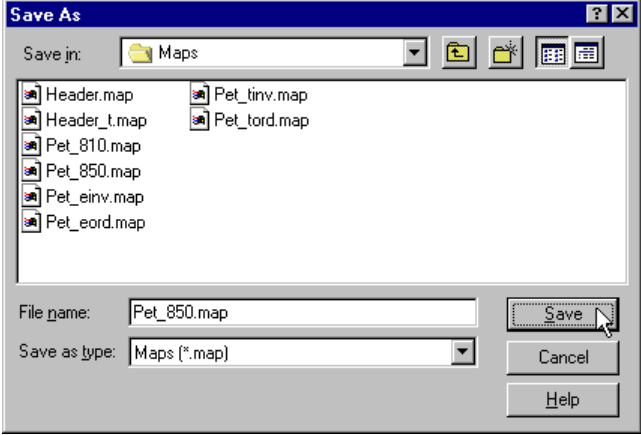
Procedure   Use this procedure to create the export map for the Inbound Purchase Order.

Step	Action
1	From the File menu, select **New**. **System response** The system displays the New Map Wizard.
2	Answer the following questions and then click **Next**. ◗ *What kind of map are you creating?* Select **Export**. ◗ *What is the name of the map?* Type **PET_850** as the unique name of the map. The system adds the .MAP extension. ◗ *What is your name?* Type your name if it differs from the user name prompted by the system. **System response** The system displays the New Map Wizard - Input Format dialog box. **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 Gentran:Server system.
3	From the Create a new data format using this syntax list, select **Delimited EDI** and click **Customize**. **System response** The system displays the New Delimited EDI Wizard. *(Continued on next page)*

(Contd) Step	Action
4	Click **Next**.
5	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**.
6	Select the following: a. Select the **[X] X12** standards agency, b. **[003030] ANSI X12 VERSION 3 RELEASE 3** version, and c. **[850] PURCHASE ORDER** transaction set. d. Click **Next**.
5	Click **Finish**. **System response** The system loads the selected transaction set and returns to the New Map Wizard - Input Format dialog box.
6	Click **Next**. **System response** The system displays the New Map Wizard - Output Format dialog box.
7	From the Create a new data format using this syntax list, select **Positional** and click **Next**. **System response** The system displays the New Map Wizard.
8	Click **Finish** to create the new map (this may take a few seconds). **System response** 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.

(Contd) Step	Action
9	From the Edit menu, select **Details**. **System response** The system displays the Translation Object Details dialog box.
10	In the Description box, type **PET X 3030 850 Export**.
11	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 (e.g., 2.0, 3.0). If you make a minor change to the map, increase the version by a percentage point (e.g., 2.1, 2.2).
12	Click **OK** to save the translation object details. (Continued on next page)

(Contd) Step	Action
13	From the File menu, select Save As. **System response** The system displays the Save As dialog box.
14	Verify that the File name box contains pet_850.map. **Note** You allocate a file name to a map when you save the map for the first time or use the Save As function to save a copy of the map under a different name. The name of the map must be one to eight unique characters that define the map in a meaningful way. The default file extension is ".MAP". Using this extension distinguishes map files from other types of files, such as translation objects. The naming conventions in Gentran:Server for Windows include: ◗ The source map (map created in Gentran:Server for Windows) should be named using the file extension **.MAP** (e.g., HEADER.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** (e.g., HEADER.TPL). **Recommendation** 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. **Caution** Be very careful not to overlay the source map with the compiled translation object. Use the .TPL file extension to distinguish the translation object. (Continued on next page)

(Contd) Step	Action
15	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 by selecting Save from the File menu. |

Next step   Now that you've finished creating and saving the new Export map, you need to define the Input and Output sides of the map.

Activating EDI Map Components

Overview   This table describes the map components for an EDI File. This table describes the map components for an EDI File.

Icon	Description
✉	EDI File
▢	Segment
🔲	Loop (Group)
🎨	Composite Element
🔹	Element

This table describes the map components for a Positional File.

Icon	Description
▢	Positional File
⬤	Record
🔲	Loop (Group)
🔹	Field

The specific EDI map components that you use depend on the type of map you are creating, the standard, version, and transaction set (document) selected, and which segments and elements your company requires. We recommend that you determine which segments and elements you are using before starting this step.

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

Overview (contd)   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.

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

Step	Action
1	Double-click the groups and segments in this table to open them. These are the groups, subgroups, segments and elements you need to activate. 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.

Groups		Segment	Element
		DTM	DTM02
N1		N1	N102
		N3	N302
		N4	N401 N402 N403
PO1 (already activated)		PO1	PO102
			PO103
			PO104
			PO107
	PID	PID:2	PID:205
		ITA:2	ITA:207

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

Step	Action
2	Click the **Activate** button on the Main Toolbar.

(Contd) Step	Action
3	Click each inactive group that you need to use. This activates the groups. **Note** If you accidentally click a group that you did not mean to activate, click the group with the right mouse button to access the shortcut menu.
4	Click each inactive segment that you need to use. This activates the segments.
5	Click each inactive element that you need to use. This activates the elements.
6	Once you have activated all the necessary groups, segments, and elements, click the **Activate** button on the Main Toolbar to turn activation mode off.
7	From the View menu, select Show Active Only. **Note** This function instructs the system to only display the active map components.

Verifying EDI Delimiters

Overview　If you are using an EDI standard that contains composite elements or subelements, you must verify that Gentran:Server for Windows 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). Since 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.

Recommendation

Although verifying EDI delimiters in Gentran:Server for Windows is mandatory *only* if you are using a standard with composite elements or subelements (which we are not using in this tutorial), we recommend that perform this task regardless of which standard you use.

Procedure　Use this procedure to verify EDI delimiters.

Step	Action
1	Right-click the 850 PURCHASE ORDER EDI file icon to access the shortcut menu, and select **Properties** from the shortcut menu. **System response** The system displays the File Properties dialog box.
2	Select the Delimiters tab. **System response** The system displays the delimiter settings. *(Continued on next page)*

(Contd) Step	Action
3	Verify the required delimiters for the ANSI X12 standard. If the delimiters differ from the default in the table below, type either the character or the hexadecimal value in the correct box. The default delimiters for ANSI X12 are the following (the tag delimiter, release character, and decimal separator are not used for ANSI):

Box	ANSI X12 Default Character	ANSI X12 Default Hex Value
Tag Delimiter		
Element Delimiter	* (asterisk)	0x2A
Segment Delimiter	~ (tilde)	0x7E
Sub Element Delimiter	: (colon)	0x3A
Release Character		
Decimal Separator		

4	Click **OK** to exit the File Properties dialog box.

Application File Format

Defining Your Application File Format

Overview

If either side of your map is positional (fixed-format), you must define your application file format to Gentran:Server for Windows. 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, so that your system can accurately process the data.

Before you define your application file format, obtain a layout of the necessary records, fields, and groups. The records contain related fields and the groups contain related records. 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.

For example, your application file format contains records and groups. The records contain fields and the group contains records. This means that you must create records and groups before you create the subordinate fields and records.

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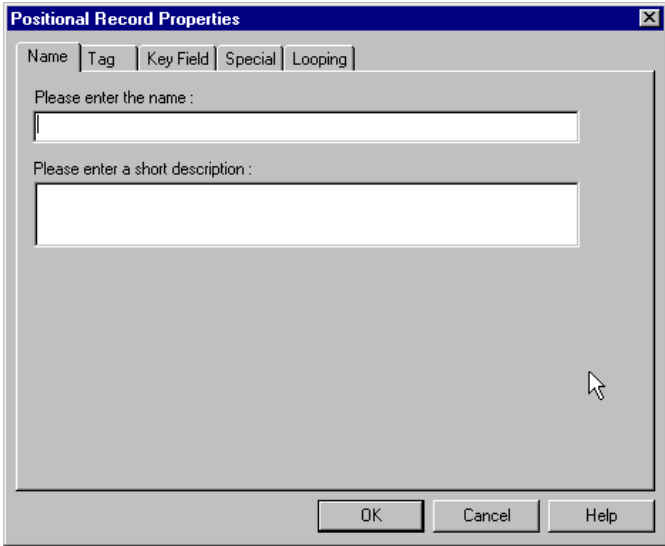
Overview   The first record in your application file is generally the header record. The first record is subordinate to the icon in the application side (output) of the map.

The POHEADER (header) record 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. This header record is mandatory and occurs once for each purchase order translated.
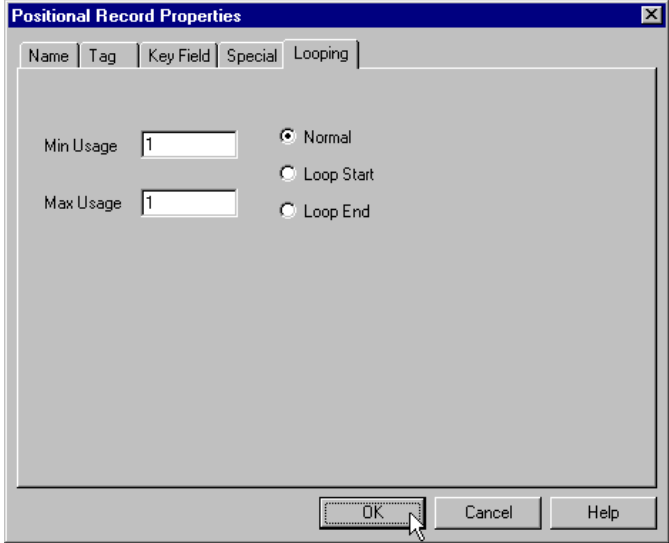
Recommendation
We recommend that you include a header record in each application file. The header record is typically mandatory and repeats only once (not a looping structure).

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

Step	Action
1	Right-click the Positional File icon to highlight it (Output for an inbound map) and select **Create Sub\Record** from the menu. **System Response** The Positional Record Properties dialog box is displayed.
2	In the Name box, type "POHEADER" as the record identification code. **Caution** Do not use spaces or dashes (-) in the Name box. You can use the underscore (_) to separate words. *(Continued on next page)*
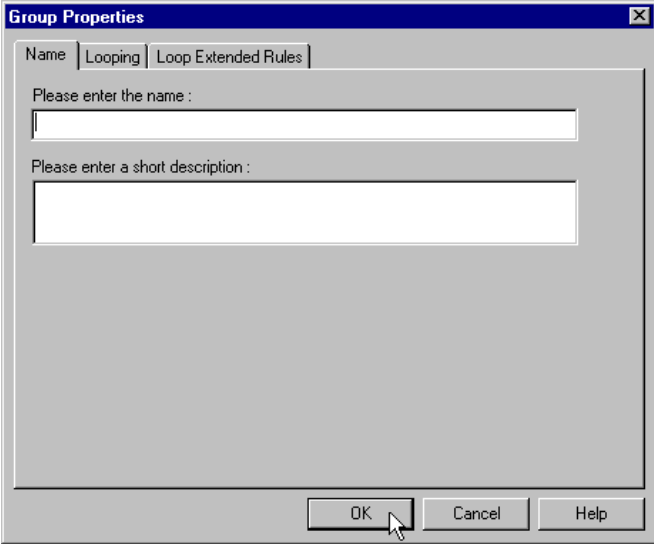
(Contd) Step	Action
3	In the Desc box, type "PO Header Record" as the description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the **Tag** tab.
5	In the Tag box, type "HDR" as the record identification code <TAG>. For example, 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. **Recommendation** 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.
6	In the Position box, accept the default of "1" as the starting column position of the tag in the data record.
7	Select the **Looping** tab. *(Continued on next page)*
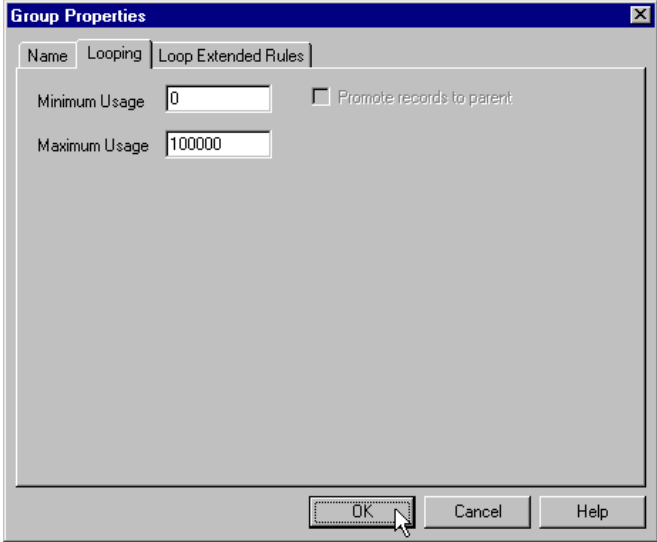
(Contd) Step	Action
8	In the Min Usage box, type "1" to make the header record mandatory. This is the minimum number of times the record must repeat. If you want the record to repeat (loop), type the maximum number of times it can repeat in the Max Usage box. **Note** If the Min Usage box contains a 0 (zero), the record is conditional. If the Min Usage box contains a 1 or greater, the record is mandatory. **Recommendation** We recommend that you always accept the default of "1" so it remains a record that only occurs once.
9	Click **OK** to create the POHEADER record.

Creating the PODetailGroup

Overview   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 per order translated. This equals a maximum order of 100,000 items. 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.

Procedure   Use this procedure to create the PODetailGroup.

Step	Action
1	To create a group that is at the same level as the POHEADER record, highlight the POHEADER and select **Insert** from the Edit menu. Select **Group** from the submenu. **System Response** The Group Properties dialog box is displayed. ![Group Properties dialog box]
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. **Caution** Do not use spaces or dashes (-) in the Name box. You can use the underscore (_) to separate words. *(Continued on next page)*

(Contd) Step	Action
3	In the Description box, type "PO Detail Group" as the loop description. This box is used to provide a brief explanation of the loop. Accept the default of "0" (zero) in the Minimum Usage box. This is the minimum number of times the loop must be repeated. For a conditional loop, the minimum usage should always be "0" (zero). For a mandatory loop, the minimum usage should be "1" or greater.
4	Select the **Looping** tab.
5	In the Maximum Usage box, type "100000". This indicates that this group of records can repeat up to 100,000 times. The Looping tab should now look like this: **Group Properties** Name │ Looping │ Loop Extended Rules Minimum Usage 0 ☐ Promote records to parent Maximum Usage 100000 OK Cancel Help
6	Click **OK** to create the PODetailGroup.

Creating the PODetailGroup Subordinate Records

Overview   In this tutorial, the PODetailGroup contains three subordinate records:
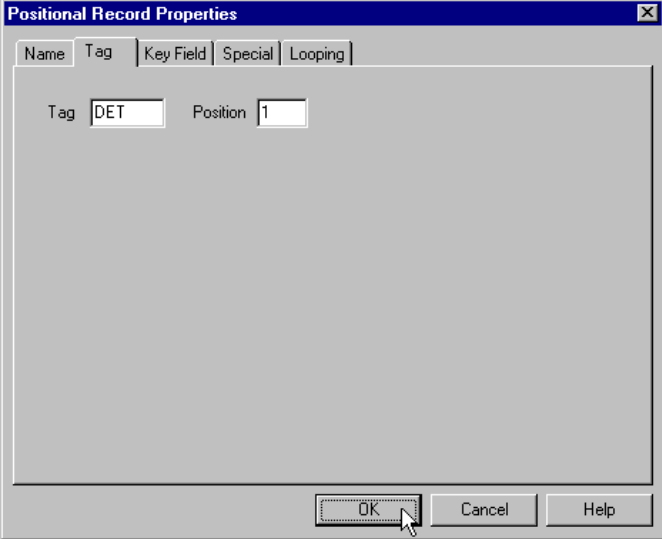
◗ PODETAIL

◗ PODESC

◗ POALLCHG.

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   Use this procedure to create the PODETAIL record.
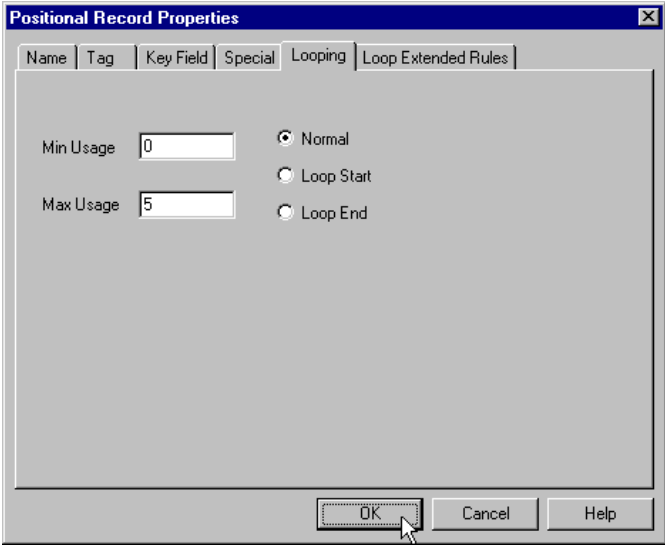
Step	Action
1	Click the PODetail Group to highlight it and select **Create Sub** from the Edit menu. Select **Record** from the submenu. **System Response** The Positional Record Properties dialog box is displayed.
2	In the Name box, type "PODETAIL" as the record identification code.
3	In the Desc box, type "PO Detail Record" as the description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the **Tag** tab.
5	In the Tag box, type "DET" as the record identification code <TAG>. For example, 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. *(Continued on next page)*

(Contd) Step	Action
6	In the Position box, accept the default of "1" as the staring column position of the tag in the data record. The Tag tab should now look like this:
7	Click **OK** to create the PODETAIL record.

**Creating the
PODESC record**

Use this procedure to create the PODESC record.
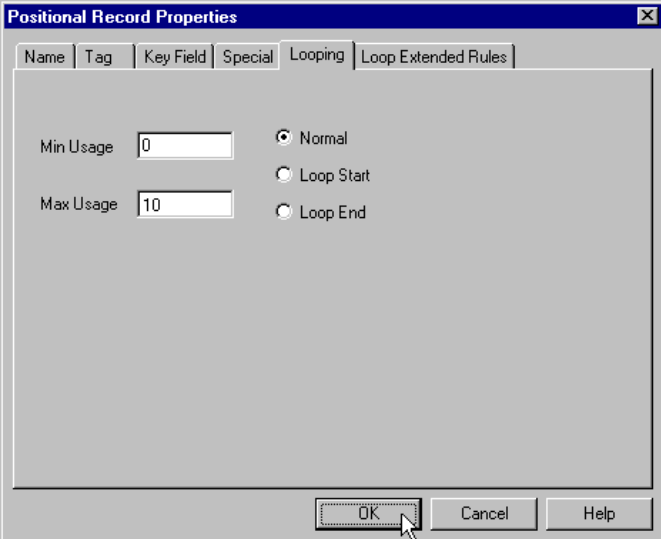
Step	Action
1	Highlight the PODETAIL record. From the Edit menu, select **Insert**. From the submenu, select **Record**. **System response** The system displays the Positional Record Properties dialog box.
2	In the Name box, type "PODESC" the record identification code.
3	In the Desc box, type "Description Record" as a description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the **Tag** tab.

(Contd) Step	Action
5	In the Tag box, type "DES" as the record identification code <TAG>. For example, 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.
6	In the Position box, accept the default of "1" as the starting column position of the tag in the data record.
7	Select the **Looping** tab.
8	The Looping section of the Positional Record Properties dialog box enables you to specify the minimum and maximum number of times this record can repeat, and whether a record is mandatory or conditional. Alter this section *only* if the record is a looping structure. Otherwise, accept the defaults. Accept the default of "0" in the Min Usage box. This is the minimum amount of times the record must repeat. In the Max Usage box, type "5" as the maximum amount of times the record can repeat. The Looping tab should now look like this:
9	Click **OK** to create the PODESC record.

Creating the POALLCHG record

Use this procedure to create the POALLCHG record:

Step	Action
1	Highlight the PODESC record. From the Edit menu, select **Insert**. From the submenu, select **Record**. **System response** 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. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the **Tag** tab.
5	In the Tag box, type "ACG" as the record identification code <TAG>. For example, 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.
6	In the Position box, accept the default of "1" as the starting column position of the tag in the data record.
7	Select the **Looping** tab. (Continued on next page)

(Contd) Step	Action
8	The Looping section of the Positional Record Properties dialog box enables you to specify the minimum and maximum number of times this record can repeat, and whether a record is mandatory or conditional. Alter this section *only* if the record is a looping structure. Otherwise, accept the defaults. Accept the default of "0" in the Min Usage box. This is the minimum amount of times the record must repeat. In the Max Usage box, type "10" as the maximum amount of times the record can repeat. The Positional Record Properties dialog box should now look like this:
9	Click **OK** to create the POALLCHG record.

Creating the POSUMMARY Record
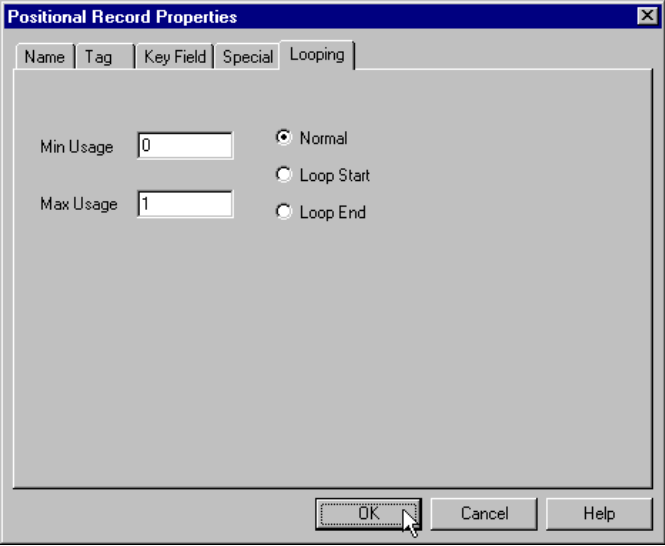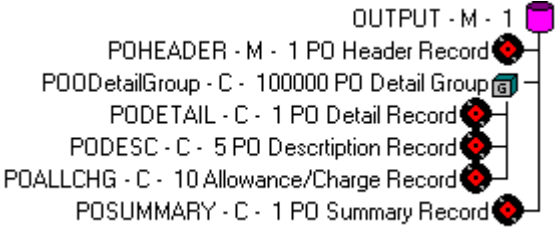
Overview   The final record you need to create (POSUMMARY) is at the same level as the POHEADER record and the PODetailGroup.

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.

Procedure   Use this procedure to create the POSUMMARY record.

Step	Action
1	Highlight the PODetailGroup. From the Edit menu, select **Insert**. From the submenu, select **Record**. **System response** The system displays the Positional Record Properties dialog box. **Note** Use "Insert" from the Edit menu when you want to create equal records/groups and use "Create Sub" to create subordinate groups/records/fields.
2	In the Name box, type "POSUMMARY" as the record identification code.
3	In the Desc box, type "PO Summary Record" as the description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the **Tag** tab.
5	In the Tag box, type "SUM" as the record identification code <TAG>. For example, 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.
6	In the Position box, accept the default of "1" as the starting column position of the tag in the data record.
7	Select the **Looping** tab. *(Continued on next page)*

(Contd) Step	Action
8	Accept the default of "0" in the Min Usage box. This is the minimum number of times the record must repeat. Also, accept the default of "1" in the Max Usage box so the summary record only occurs once (not a looping structure). The Positional Record Properties dialog box should now look like this:
9	Click **OK** to create the POSUMMARY record. The application side of the map (output) should now look like this:

Creating the Application Fields

Overview

Introduction   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.

Application fields   This table describes the application fields that you need to create.
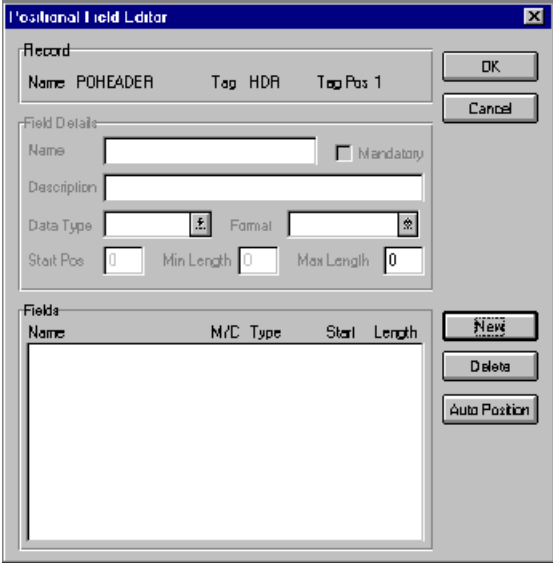
Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (POHEADER) 1 per document					
CUSTNUMHDR	X	Customer Number	String	1	8
PODATEHDR	X	Purchase Order Date	Dt/Tm (MMDDYY)	6	6
PONUMHDR	X	Purchase Order Number	String	1	22
POTYPCODE		Purchase Order Type Code	String	0	2
SHIPTONAME		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	30
SHIPTOSTATE		Ship To State	String	0	2
SHIPTOPOSTAL CODE		Ship To Postal Code	String	0	9
BILLTONAME		Bill To Name	String	0	35
BILLTOADD1		Bill To Address Line 1	String	0	35
BILLTOADD2		Bill To Address Line 2	String	0	35
(Continued on next page)					

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
BILLTOCITY		Bill To City	String	0	30
BILLTOSTATE		Bill To State	String	0	2
BILLTOPOSTAL CODE		Bill To Postal Code	String	0	9
DELDATE	X	Delivery Date	Dt/Tm (MMDDYY)	6	6
CANCELDATE	X	Cancel By Date	Dt/Tm (MMDDYY)	6	6
NUMLINEITEMS	X	Number of Line Items	Number (N0)	1	6
PO Detail Group (PODetailGroup) up to 100,000 per document					
Detail Record (PODETAIL) 1 per document					
CUSTNUMDET	X	Customer Number	String	1	8
PODATEDET	X	Purchase Order Date	Dt/Tm (MMDDYY)	6	6
PONUMDET	X	Purchase Order Number	String	1	22
CUSTPROCODE	X	Customer Product Code	String	1	30
UPCCODE	X	UPC Code	String	1	30
UOM		Unit of Measure	String	0	7
UNITPR		Unit Price	Number (N2)	0	14
QTYORD	X	Quantity Ordered	Number (N0)	1	9
Description Record (PODESC) up to 5 per document					
CUSTNUMDESC	X	Customer Number	String	1	8
PODATEDESC	X	Purchase Order Date	Dt/Tm (MMDDYY)	6	6
PONUMDESC	X	Purchase Order Number	String	1	22
DESC		Description	String	0	80

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Allowance/Charge Record (POALLCHG) up to 10 per document					
CUSTNUMAC	X	Customer Number	String	1	8
PODATEAC	X	Purchase Order Date	Dt/Tm (MMDDYY)	6	6
PONUMAC	X	Purchase Order Number	String	1	22
ACIND		Allowance or Charge Indicator	String	0	1
ACMETHAND		Allowance or Change Method of Handling Code	String	0	2
ACTOTAMT		Allowance or Charge Total Amount	Number (N2)	0	9
Summary Record (POSUMMARY) 1 per document					
CUSTNUMSUM	X	Customer Number	String	1	8
PODATESUM	X	Purchase Order Date	Dt/Tm (MMDDYY)	6	6
PONUMSUM	X	Purchase Order Number	String	1	22
NUMLINEITEMSUM	X	Number of Line Items	Number (N0)	1	6

How to Create the POHEADER Record Application Fields

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

Step	Action
1	Right-click the POHEADER application record to access the shortcut menu.
2	From the Shortcut menu, select **Edit Fields** to access the Positional Field Editor dialog box.
3	If the field you are creating is the first field in the record, click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type "CUSTNUMHDR", which is the name of the first field in the POHEADER record. 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.
5	Select the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system cannot identify who sent the Purchase Order.
6	In the Description box, type "Customer Number" as the description of the box. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.

(Contd) Step	Action
7	From the Data Type list, select "String" as the type of field to specify that this field must contain alphanumeric data. **Tip** If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You want to specify field start positions if, for example, you are only using a few fields but you want them positioned exactly in the record. In this tutorial, you use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see Step 13 below for more information on the Auto Position function.
8	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list.
9	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.
10	In the Max Length box, type "8" as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to *Application fields* on page 2 - 40. **Note** 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. (Continued on next page)

(Contd) Step	Action
13	After adding the last field, click **Auto Position** to automatically position the fields in the record. 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. Click **Yes** to acknowledge the warning message that fields are sequenced in order. **Caution** 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. The Positional Field Editor dialog box should now look like the following (only the last eleven fields in the record are currently visible in the Fields list):
14	When you are finished adding fields to the POHEADER record, click **Close**.
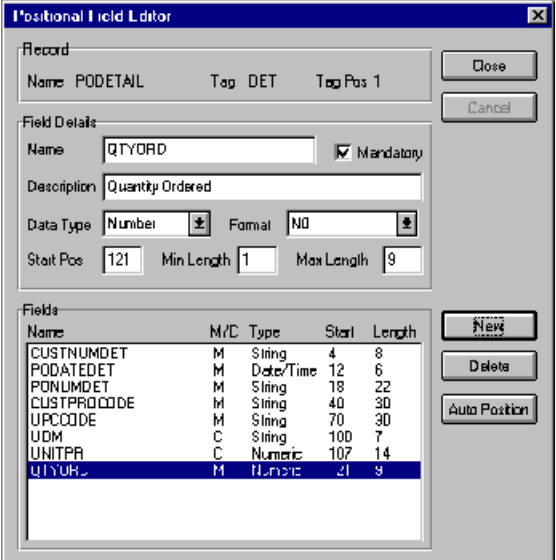
How to Create the PODETAIL Record Application Fields

Procedure   Complete the following steps to create the application fields for the PODETAIL record.

Step	Action
1	Right-click the PODETAIL application record to access the shortcut menu.
2	From the Shortcut menu, select **Edit Fields** to access the Positional Field Editor dialog box.
3	If the field you are creating is the first field in the record, click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type "CUSTNUMDET", which is the name of the first field in the PODETAIL record. 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, "DET" for the detail record.
5	Select the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system cannot identify who sent the Purchase Order.
6	In the Description box, type "Customer Number" as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select "String" as the type of the field, to specify that this field must contain alphanumeric data.
8	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list.
9	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.
10	In the Max Length box, type "8" as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field. *(Continued on next page)*

(Contd) Step	Action
12	Create the rest of the fields according to *Application fields* on page 2 - 40. **Note** 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.
13	After adding the last field, click **Auto Position** to automatically position the fields in the record. 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. Click **Yes** to acknowledge the warning message that fields are sequenced in order. The Positional Field Editor dialog box should now look like the following:
14	When you are completely finished adding fields to the PODETAIL record, click **Close**.

How to Create the PODESC Record Application Fields

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

Step	Action
1	Right-click the PODESC application record to access the shortcut menu.
2	From the Shortcut menu, select **Edit Fields** to access the Positional Field Editor dialog box.
3	If the field you are creating is the first field in the record, click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type "CUSTNUMDESC", which is the name of the first field in the PODESC record. 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, "DESC" for the description record.
5	Select the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system cannot identify who sent the Purchase Order.
6	In the Description box, type "Customer Number" as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select "String" as the type of the field, to specify that this field must contain alphanumeric data.
8	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list.
9	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.
10	In the Max Length box, type "8" as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field. (Continued on next page)

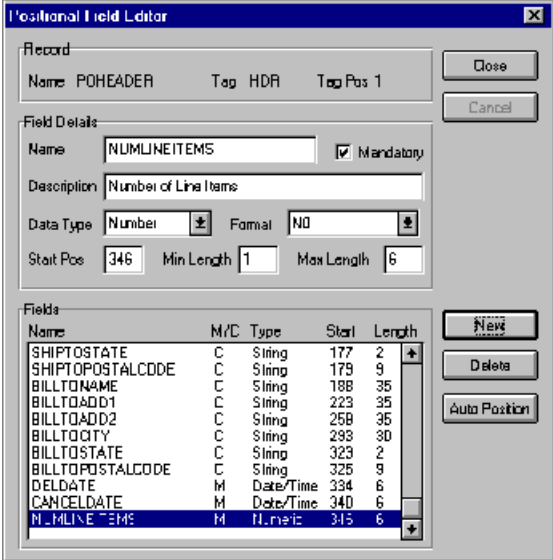| (Contd) Step	Action
12	Create the rest of the fields according to *Application fields* on page 2 - 40. **Note** If you press **ENTER** after typing the information for the last field in the record, a blank field is created. Click **Delete** and then **Yes** to delete the blank field.
13	After adding the last field, click **Auto Position** to automatically position the fields in the record. 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. Click **Yes** to acknowledge the warning message that fields are sequenced in order. The Positional Field Editor dialog box should now look like the following:
14	When you are finished adding fields to the PODESC record, click **Close**.

How to Create the POALLCHG Record Application Fields

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

Step	Action
1	Right-click the POALLCHG application record to access the shortcut menu.
2	From the Shortcut menu, select **Edit Fields** to access the Positional Field Editor dialog box.
3	If the field you are creating is the first field in the record, click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill the field values in the Field Details section.
4	In the Name box, type "CUSTNUMAC", which is the name of the first field in the POALLCHG record. 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, "AC" for the allowance/charge record.
5	Select the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system cannot identify who sent the Purchase Order.
6	In the Description box, type "Customer Number" as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select "String" as the type of field, to specify that this field must contain alphanumeric data.
8	If the Data Type is "Number" or "Dt/Tm", select the appropriate format from the Format list.
9	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.
10	In the Max Length box, type "8" as the maximum length of the field.
11	To add the field and create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field. *(Continued on next page)*

(Contd) Step	Action
12	Create the rest of the fields according to *Application fields* on page 2 - 40. **Note** 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.
13	After adding the last field, click **Auto Position** to automatically position the fields in the record. 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. Click **Yes** to acknowledge the warning message that fields are sequenced in order. The Positional Field Editor dialog box should now look like the following:
14	When you are completely finished adding fields to the POALLCHG record, click **Close**.

How to Create the POSUMMARY Record Application Fields

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

Step	Action
1	Right-click the POSUMMARY application record to access the shortcut menu.
2	From the Shortcut menu, select **Edit fields** to access the Positional Field Editor dialog box.
3	If the field you are creating is the first field in the record, click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type "CUSTNUMSUM", which is the name of the first field in the POSUMMARY record. 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, "SUM" for the summary record.
5	Select the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system cannot identify who sent the Purchase Order.
6	In the Description box, type "Customer Number" as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select "String" as the type of field, to specify that this field must contain alphanumeric data.
8	If the Data Type is "Number" or "Dt/Tm", select the appropriate format from the Format list.
9	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.
10	In the max Length box, type "8" as the maximum length of the field.

(Contd) Step	Action
11	To add the field and to create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field. **Note** 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.
12	Create the rest of the fields according to *Application fields* on page 2 - 40.
13	After adding the last field, click **Auto Position** to automatically position the fields in the record. 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. Click **Yes** to acknowledge the warning message that fields are sequenced in order. The Positional Field Editor dialog box should now look like the following:
14	When you are finished adding fields to the POSUMMARY record, click **Close**.

Next step After defining your application file format to Gentran:Server for Windows, you must map the necessary data from the EDI file to your application fields.

Mapping Your Data

Overview

Introduction   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 three-field record key that ensures that the same identifying information is located at the beginning of each record. Your application file format 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

◗   purchase order date

◗   purchase order number

This section examines each application field sequentially and describes the mapping operations that you need to perform to map the appropriate data into each field.

Mapping the POHEADER Record

Overview

Introduction   This section describes the actions necessary for you to map to each field in the POHEADER (header) record. The POHEADER record contains:

◗   purchase order type code

◗   shipping and billing information

◗   delivery information

◗   cancel by date information

◗   the number of line items in the purchase order

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

CUSTNUMHDR Field

Overview The Application Integration subsystem allows you to use selected information from the Gentran:Server for Windows 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.

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).

Mapping the customer number 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.

Step	Action
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." This indicates that the system updates this field with selected information from the 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." This is the field that contains the customer number for this partner. *(Continued on next page)*

(Contd) Step	Action
7	From the Map To list on the right, select "CUSTNUMHDR." This is the field to which the information from the Partner Editor is mapped. 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

Overview   You can map the PODATEHDR field directly from the BEG05 (Purchase Order Date) field, by using the 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. The link between the two fields is visually represented with a line connecting the two fields.

Procedure   Use this procedure to map the purchase order date.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the BEG05 element in the BEG record. When you click the element, the cursor changes to a "Link Arrow": LINK
3	Click the application field PODATEHDR in the POHEADER record.
4	A line is displayed, visually illustrating the link between the BEG05 element and the PODATEHDR field.

PONUMHDR Field

Overview   You can map the PONUMHDR field directly from the BEG03 (Purchase Order Number) field, by using the 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. The link between the two fields is visually represented with a line connecting the two fields.

Also, 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 Gentran:Server for Windows. This allows you to differentiate between documents in the document browsers in Gentran:Server for Windows.

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. Then, the Name field of the document browsers that contain this document contains the data from the selected element or field.

Recommendation
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." For a Purchase Order, the document name is typically the Purchase Order Number.

Procedure   Use this procedure to map the purchase order number.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the BEG03 element in the BEG record. When you click the element, the cursor changes to a "Link Arrow": LINK
3	Click the application field PONUMHDR in the POHEADER record.
4	A line is displayed, visually illustrating the link between the BEG03 element and the PONUMHDR field.

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

Step	Action
1	Double-click the BEG03 (0324) element to display the Element Properties dialog box.
2	Select the **Standard Rule** tab.
	(Continued on next page)

Step	Action
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 Gentran:Server for Windows.
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. The Standard Rule tab of the Element Properties dialog box should now look like this:
6	Click **OK** to set up the document name.

POTYPCODE Field

Overview   You can map the POTYPCODE field directly from the BEG02 (Purchase Order Type Code) field, by using the 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. The link between the two fields is visually represented with a line connecting the two fields.

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

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the BEG02 element in the BEG record. When you click the element, the cursor changes to a "Link Arrow": LINK
3	Click the application field POTYPCODE in the POHEADER record.
4	A line is displayed, visually illustrating the link between the BEG02 element and the POTYPCODE field.

Mapping Shipping and Billing Information

Overview

Introduction

On the EDI side of the map, the name and address information is contained in a series of segments that are logically grouped in the N1 group. The N1 group contains shipping and billing name and address information, and repeats multiple times. If the Entity Identifier Code (in the N1 segment) is "ST," that iteration of the N1 group contains shipping information. You need to map the shipping information to the Ship To fields in the application header record. If the Entity Identifier Code is "BT," that iteration of the N1 group contains billing information. You need to map the billing information to the Bill To fields in the application header record.

However, to map the correct information (e.g., either bill to or ship to) from the N1 group to the Bill To and Ship To fields in the application header record, you must extract the billing and the shipping information after determining the type. Also, 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.

Temporary storage segments and elements

To map the shipping and billing information to your application file format, you need to create temporary storage segments and elements on the EDI side of the map that do not repeat. You can use an extended rule to extract the shipping and billing information from the N1 group and move it to the appropriate temporary storage elements. Then, you can map the shipping and billing information directly from the temporary storage elements to your application fields.

Temporary Storage Segments

Overview   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

How to create the ShipToDet segment   Use this procedure to create the ShipToDet temporary storage segment.

Step	Action
1	Highlight the N1 group. The two temporary storage segments are located after the N1 group, at the same level.
2	From the Edit menu, select **Insert**. From the submenu, select **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	Select the **Tag** tab.
6	In the Tag box, type "$$$". The system does not read a segment with a value of "$$$" in the Tag box. Therefore, it does not flag this temporary storage segment as an error during compliance checking.
7	Click **OK** to create the ShipToDet temporary storage segment.

How to create the BillToDet segment

Use this procedure to create the BillToDet temporary storage segment.

Step	Action
1	Highlight the ShipToDet segment.
2	From the Edit menu, select **Insert**. From the submenu, select **Segment**.
3	In the Name box, type "BillToDet".
4	In the Description box, type "Bill To Details".
5	Select the **Tag** tab.
6	In the Tag box, type "$$$". The system does not read a segment with a value of "$$$" in the Tag box. Therefore, it does not flag this temporary storage segment as an error during compliance checking.
7	Click **OK** to create the BillToDet temporary storage segment.

Temporary Storage Elements

Overview　　　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 SHIPTOADDR1 SHIPTOADDR2 SHIPTOCITY SHIPTOSTATE SHIPTOPCODE	35 35 35 30 2 9
BillToDet	BILLTONAME BILLTOADDR1 BILLTOADDR2 BILLTOCITY BILLTOSTATE BILLTOPCODE	35 35 35 30 2 9

How to create the ShipToDet elements　　　Use this procedure to create the ShipToDet elements.

Step	Action
1	Highlight the ShipToDet segment. From the Edit menu, select **Create Sub** to create a subordinate element.
2	From the submenu, select **Element**. The Element Properties dialog box is displayed so you can define the characteristics of the new element.
3	In the Name box, type "SHIPTONAME".
4	Select the **Validation** tab.
5	In the Maximum box, type "35" as the maximum length.
6	Click **OK** to create the SHIPTONAME element.
7	Now you can create the remaining Ship To elements. With the SHIPTONAME element highlighted, select **Insert** from the Edit menu. From the submenu, select **Element** to display the Element Properties dialog box.
8	Repeat Steps 3 through 7 to complete the remaining ShipToDet elements listed in the table above.

**How to create the
BillToDet elements**

Use this procedure to create the BillToDet elements.

Step	Action
1	Highlight the BillToDet segment. From the **Edit** menu, select **Create Sub** to create a subordinate element.
2	From the submenu, select **Element**. The Element Properties dialog box is displayed so you can define the characteristics of the new element.
3	In the Name box, type "BILLTONAME".
4	Select the **Validation** tab.
5	In the Maximum box, type "35" as the maximum length.
6	Click **OK** to create the BILLTONAME element.
7	Now you can create the remaining Bill To elements. With the BILLTONAME element highlighted, select **Insert** from the Edit menu. From the submenu, select **Element** to display the Element Properties dialog box.
8	Repeat Steps 3 through 7 to complete the remaining Bill To elements listed in the table above.

Changing the Max Use of the N3 Segment

Overview　　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.

Procedure　　Use this procedure to make N3 a non repeating segment.

Step	Action
1	Right-click the N3 segment (in the N1 group) to access the shortcut menu.
2	From the shortcut menu, select **Properties** to access the EDI Segment Properties dialog box.
3	Select the **Looping** tab.
4	In the Max Usage box, type "1" to make N3 a non repeating segment. This box is currently set to "2."
5	Click **OK** to accept the change to the N3 segment.

Using an extended rule　　Now you can use an extended rule to extract the shipping and billing information from the N1 group and move it to the appropriate temporary storage elements. The extended rule should be executed after the system reads an occurrence of the N1 group.

How to create an extended rule　　Use this procedure to create the extended rule.

Step	Action
1	Right-click the N1 group to access the shortcut menu.
2	From the shortcut menu, select **Extended Rules** to display the Group Properties dialog box.
3	Select the **On End** option. This specifies that the rule is executed when the loop terminates. The system loads an occurrence of the N1 group (containing the billing or shipping information), and then executes this rule. *(Continued on next page)*

(Contd) Step	Action

4	In the Editor list, type the following:
	
	``
	IF #0098 = "BT" THEN
	BEGIN
	   $850.#BILLTONAME = #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
	``
	
	Note
	If a segment/record or element/field occurs more than once in a map, it is identified by its name <ID>. The second and subsequent occurrences are identified by <ID>:n, where 'n' is the number of occurrences in the map.

5	Click **Compile** to validate the syntax of the extended rule. Every 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 the accuracy of the rule after you create it. **Note** This compiles the rule interactively, and allows you to correct any errors that are generated. Any errors or warnings generated in the compilation process are displayed in the Errors list.
6	Click **OK** to add the extended rule to the N1 group.

How to Map the Ship To Name Information

Overview　　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.

Procedure　　Use this procedure to map the Ship To Name.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the SHIPTONAME element in the ShipToDet temporary storage segment. When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field SHIPTONAME in the POHEADER record.
4	A line is displayed, visually illustrating the link between the SHIPTONAME element and the SHIPTONAME application field.

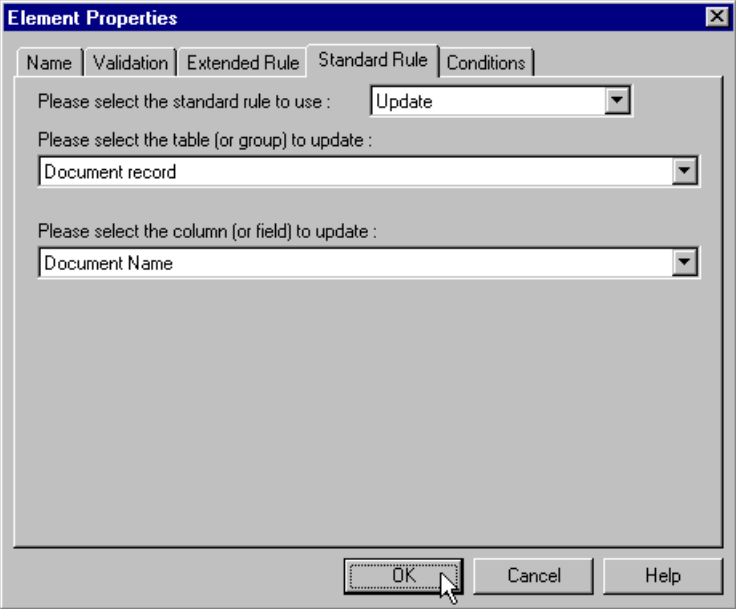**Mapping other application fields and elements**　　This table lists the application fields in the POHEADER record and the elements they are mapped to.

To map POHEADER field	You need to link
SHIPTOADD1	the SHIPTOADDR1 element in the ShipToDet temporary storage segment to the application field SHIPTOADD1 in the POHEADER record.
SHIPTOADD2	the SHIPTOADDR2 element in the ShipToDet temporary storage segment to the application field SHIPTOADD2 in the POHEADER record.
SHIPTOCITY	the SHIPTOCITY element in the ShipToDet temporary storage segment to the application field SHIPTOCITY in the POHEADER record.
SHIPTOSTATE	the SHIPTOSTATE element in the ShipToDet temporary storage segment to the application field SHIPTOSTATE in the POHEADER record.

(Continued on next page)

(Contd) To map POHEADER field	You need to link
SHIPTPCODE	the SHIPTOPCODE element in the ShipToDet temporary storage segment to the application field SHIPTOPOSTALCODE in the POHEADER record.
BILLTONAME	the BILLTONAME element in the BillToDet temporary storage segment to the application field BILLTONAME in the POHEADER record.
BILLTOADD1	the BILLTOADDR1 element in the BillToDet temporary storage segment to the application field BILLTOADD1 in the POHEADER record.
BILLTOADD2	the BILLTOADDR2 element in the BillToDet temporary storage segment to the application field BILLTOADD2 in the POHEADER record.
BILLTOCITY	the BILLTOCITY element in the BillToDet temporary storage segment to the application field BILLTOCITY in the POHEADER record.
BILLTOSTATE	the BILLTOSTATE element in the BillToDet temporary storage segment to the application field BILLTOSTATE in the POHEADER record.
BILLTOPCODE	the BILLTOPCODE element in the BillToDet temporary storage segment to the application field BILLTOPOSTALCODE in the POHEADER record.

Mapping Date Segments and Elements

Overview

Introduction

The DTM segment contains all the date information for the purchase order, and repeats multiple times. If the Date/Time Qualifier (in the DTM segment) is "001," that iteration of the DTM segment contains Delivery Date information. You need to map the delivery information to the DELDATE field in the application header record. If the Date/Time Qualifier is "002, that iteration of the DTM segment contains Cancel By Date information. You need to map the cancel by date information to the CANCELDATE field in the application header record.

Temporary storage segments and elements

However, you do not know the order in which these dates appear in the EDI data. Therefore, to map the delivery and cancel by date information to your application file, you need to 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.

How to Create a Temporary Storage Date Segment

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

Step	Action
1	Highlight the DTM segment. The temporary storage segment is located after the DTM segment, at the same level.
2	From the Edit menu, select **Insert**. From the submenu, select **Segment**. **System response** The EDI Segment Properties dialog box is displayed.
3	In the Name box, type "TEMPDATES".
4	In the Description box, type "Temporary Storage Dates".
5	Select the **Tag** tab.
6	In the Tag box, type "$$$". The system does not read a segment with a value of "$$$" in the Tag box. Therefore, it does not flag this temporary storage segment as an error during compliance checking.
7	Click **OK** to create the TEMPDATES temporary storage segment.

How to Create a Temporary Storage Date Element

Creating the TEMPDATES element

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

Step	Action
1	Highlight the TEMPDATES segment. From the Edit menu, select Create Sub to create a subordinate element.
2	From the submenu, select **Element**. The Element Properties dialog box is displayed and is used to define the characteristics of the new element.
3	In the Name box, type "DELDATE".
4	In the Description box, type "Delivery Date".
5	Select the **Validation** tab.
6	In the Minimum box, type "6" as the minimum length.
7	In the Maximum box, type "6" as the maximum length.
8	From the Type list, select Date/Time. This indicates that this is a date/time element. The system may prompt you with 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 Preferences from the Options menu. Select the Confirmations tab and clear the "Confirm everything" check box. From the "Confirm when I" list, clear the "Change format or datatype of a field" check box. Click OK to save.
9	From the Format list, select "YYMMDD," to indicate that the first two digits contain the year, the second two digits contain the month, and the last two digits contain the day.
10	Click **OK** to create the DELDATE element.
11	Create the CANCELDATE element. Highlight the DELDATE element and select **Insert** from the Edit menu. From the submenu, select **Element** to display the Element Properties dialog box.
12	In the Name box, type "CANCELDATE".
13	In the Description box, type "Cancel By Date".
14	Select the **Validation** tab.
	(Continued on next page)

(Contd) Step	Action
15	In the Minimum box, type "6" as the minimum length.
16	In the Maximum box, type "6" as the maximum length.
17	From the Type list, select Date/Time. This indicates that this is a date/time element.
18	From the Format list, select "YYMMDD," to indicate that the first two digits contain the year, the second two digits contain the month, and the last two digits contain the day.
19	Click **OK** to create the CANCELDATE element.

Using an extended rule

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).

Use this procedure to create the extended rule.

Step	Action
1	Double-click the DTM01 element to access the Element Properties dialog box.
2	Select the **Extended Rule** tab.

3	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	Click **Compile** to validate the syntax of the extended rule. Every 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 the accuracy of the rule after you create it.
5	Click **OK** to add the extended rule to the DTM01 element.

How to Map the Delivery Date Information

Overview

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.

Map DELDATE

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

Map CANCELDATE

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.

Map NUMLINEITEMS

You can map the NUMLINEITEMS field directly from the CTT01 (Number of Line Items) element, by using the Link function to link the CTT01 element in the CTT segment to the application field NUMLINEITEMS in the POHEADER record.

Mapping the PODetailGroup Records: PODETAIL Record

Overview

Introduction   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 per order translated. This equals a maximum order of 100,000 items. The information within the PODetailGroup is further segregated by record:

◗   PODETAIL

◗   PODESC

◗   POALLCHG.

This section describes 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.

CUSTNUMDET Field

Overview   The Application Integration subsystem allows you to use selected information from the Gentran:Server for Windows Partner Editor 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 key.

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.

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

Step	Action
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." This indicates that the system updates this field with the selected information from the 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." This is the field that contains the customer number for this partner. *(Continued on next page)*

(Contd) Step	Action
7	From the Map To list on the right, select "CUSTNUMDET." This is the field to which the information from the Partner Editor is mapped. The Standard Rule tab on the Field Properties dialog box should now look like this:
8	Click **OK** to add the standard rule to the CUSTNUMDET field.

PODATEDET Field

Overview
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 PODATEDET field. However, the PODATEDET field is included in the PODetailGroup, which repeats 100,000 times.

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. For example, you cannot map the purchase order date in the BEG segment directly to your application PODATEDET field.

Using an extended rule
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 BEG05 element and move it to the PODATEDET field.

Use this procedure to create the extended rule.

Step	Action
1	Double-click the PODATEDET field to access the Field Properties dialog box.
2	Select the **Extended Rule** tab.
3	In the Rule list, type the following: `#PODATEDET=$850.#0323;` The "$850.#0323" is the addressing for the BEG05 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.

PONUMDET Field

Overview   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.

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. For example, you cannot map the purchase order number in the BEG segment directly to your application PONUMDET field.

Using an extended rule   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 BEG03 element and move it to the PONUMDET field.

Use this procedure to create the extended rule.

Step	Action
1	Double-click the PONUMDET field to access the Field Properties dialog box.
2	Select the **Extended Rule** tab.
3	In the Rule list, type the following: `#PONUMDET=$850.#0324;` The "$850.#0324" is the addressing for the BEG03 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.

CUSTPROCODE/UPCCODE Fields

Overview

The Application Integration subsystem enables you to use selected information from the Gentran:Server for Windows 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.

For this map, 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 and your code for the same item. Your partner uses a customer product code (CUSTPROCODE) for each item that is meaningful to him. However, your company uses a UPC code (UPCCODE) for each item. 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.

A cross-reference table is used when you and your partner each reference an item by a different name (or number). For example, if your unique name for a "widget" is "wid" and your partner's unique name for the same "widget" is "1234," the system uses the cross-reference table to translate the two names for the "widget" item.

In this map, you have one code (UPC code) for each item ordered. Your partner refers to the same item by a different code (customer product code). You need to map the customer product code from the PO107 element (Product/Service ID) to the customer product code field (CUSTPROCODE) on the application side of the map. Therefore, you must use a cross-reference table to translate the customer product code to your UPC code. You and your partner determined what the equivalent customer product code is for each UPC code. In this tutorial, the cross-reference table is included with the partner (PETZONE.PAR) you import into Partner Editor later in this tutorial.

This table contains the information in the cross-reference table. The "My Value" field contains your UPC code for the item. The "Partner Value" field contains your partner's customer product code for the item. The "Description" field contains a description (and size, if applicable) of the item, and the Reference Data field contains additional quantity or color information.

My Value	Partner Value	Description	Reference Data
12345678901234	555666411	Large Dog Collar	Red
23456789012345	555666817	Spring Breeze Cat Litter	10# 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	

Overview (contd)

Note

You do not need to create this cross-reference table—it is included with the PETZONE partner (PETZONE.PAR) that is provided with the tutorial data when you install Gentran:Server.

Finally, you need to use a standard rule to map the translated value from the CUSTPROCODE field to the UPCCODE field.
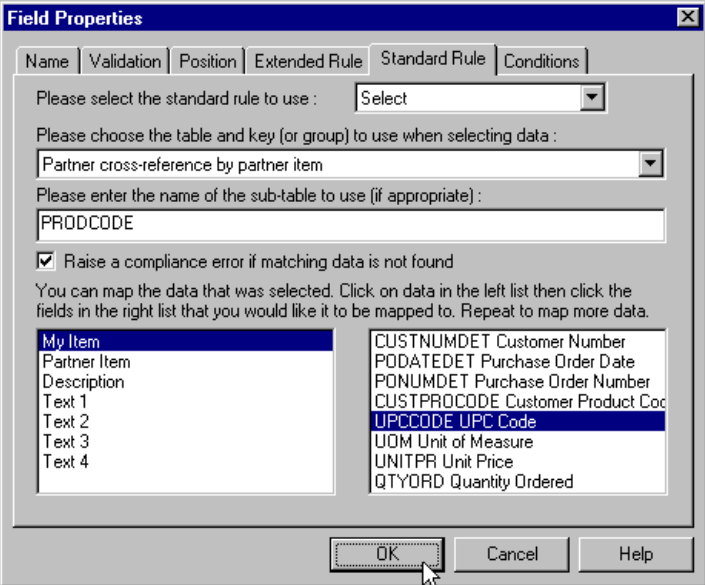
Mapping the customer product code

To map the customer product code, link the PO107 element in the PO1 segment (in the PO1 group) to the application field CUSTPROCODE in the PODETAIL record.

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

Step	Action
1	Double-click the CUSTRPOCODE field (in the PODETAIL record). The Field Properties dialog box is displayed.
2	Select the **Standard Rule** tab.
3	From the standard rule list, choose **Select**.
4	From the Table and Key box, select "Partner xref by partner item." This indicates that you are using a cross-reference table and translating by your item.
5	In the Sub Table box, type "PRODCODE". This is the name of the cross-reference table you created.
6	Select the check box to signal the system that if this translation fails, you want the system to generate an error.
7	From the Map From list on the left, select "My Item." This is the field from which the contents are mapped.

(Contd) Step	Action
8	From the Map To list on the right, select "UPCCODE." This is the field to which the information from the Partner Editor is mapped. The Standard Rule tab of the Field Properties dialog box should now look like this:
9	Click **OK** to map from the cross-reference table.

UOM Field

Overview   You can map the UOM (Unit of Measurement) field directly from the PO103 (Unit or Basis for Measurement) field, by using the Link function.

Procedure   Link the PO103 element in the PO1 segment (in the PO1 group) to the application field UOM in the PODETAIL record.

UNITPR Field

Overview   You can map the UNITPR (unit price) field directly from the PO104 element, by using the Link function.

Procedure   Link the PO104 element in the PO1 segment (in the PO1 group) to the application field UNITPR in the PODETAIL record.

QTYORD Field

Overview   You can map the QTYORD (quantity ordered) field directly from the PO102 element, by using the Link function.

Procedure   Link the PO102 element in the PO1 segment (in the PO1 group) to the application field QTYORD in the PODETAIL record.

Mapping the PODetailGroup Records: PODESC Record

Overview

Introduction   This section describes the actions necessary for you to map to each field in the PODESC (description) record. The PODESC (description) 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."

CUSTNUMDESC Field

Overview   The Application Integration subsystem allows you to use selected information from the Gentran:Server for Windows Partner Editor 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 key.

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 five times in the application file format, once in each record (CUSTNUMHDR, CUSTNUMDET, CUSTNUMDESC, CUSTNUMAC, and CUSTNUMSUM).
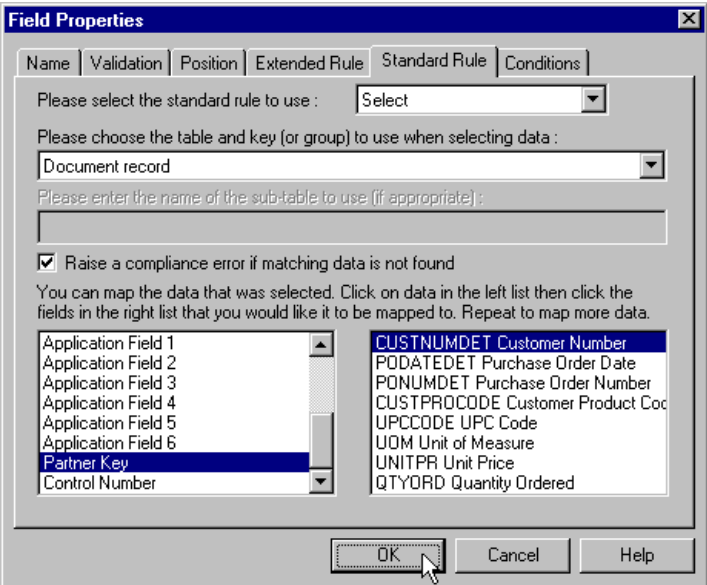
Your partner's customer number is part of the record key for each record in the application file format. It identifies which trading partner sent the purchase order. This customer number is already part of the partner definition for this partner, so you can map it from the partner definition to each customer number field.

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

Step	Action
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." This indicates that the system updates this field with selected information from the 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." This is the field that contains the customer number for this partner. *(Continued on next page)*

(Contd) Step	Action
7	From the Map To list on the right, select "CUSTNUMDESC." This is the field to which the information from the Partner Editor is mapped. 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.

PODATEDESC Field

Overview

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.

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. For example, you cannot map the purchase order date in the BEG segment directly to your application PODATEDESC field.

Using an extended rule

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 BEG05 element and move it to the PODATEDESC field.

Use this procedure to create the extended rule.

Step	Action
1	Double-click the PODATEDESC field to access the Field Properties dialog box.
2	Select the **Extended Rule** tab.
3	In the Rule list, type the following: `#PODATEDESC=$850.#0323;` 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.

PONUMDESC Field

Overview
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.

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. For example, you cannot map the purchase order number in the BEG segment directly to your application PONUMDESC field.

Using an extended rule
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 BEG03 element and move it to the PONUMDESC field.

Use this procedure to create the extended rule.

Step	Action
1	Double-click the PONUMDESC field to access the Field Properties dialog box.
2	Select the **Extended Rule** tab.
3	In the Rule list, type the following: `#PONUMDESC=$850.#0324;` 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

Overview	The DESC field is available so your partner can describe the item in a free-form format. You can map the DESC (description) field directly from the PID:205 element, by using the Link function.

Changing the Max Use of the PID group	You have determined that your partner only sends you only five PID groups (product/item descriptions), so you must change the maximum usage allowed for that group. Use this procedure to change the maximum usage of the PID group.

Step	Action
1	Right-click the PID group (in the P01 group) to access the shortcut menu.
2	From the shortcut menu, select **Properties** to access the Group Properties dialog box.
3	Select the **Looping** tab.
4	In the Max Usage box, type "5". This box is currently set to "1000."
5	Click **OK** to accept the change to the PID group.

Linking the DESC field	Link the PID:205 element in the PID:2 segment (in the PID group) to the application field DESC in the PODESC record.

Mapping the PODetailGroup Records: POALLCHG Record

Overview

Introduction   This section describes the actions necessary for you to map to each field in the POALLCHG (allowance/charge) record. The POALLCHG (allowance and charge) 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.

CUSTNUMAC Field

Overview   The Application Integration subsystem allows you to use selected information from the Gentran:Server for Windows Partner Editor 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 key.

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.

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

Step	Action
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." This indicates that the system updates this field with selected information from the 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." This is the field that contains the customer number for this partner. *(Continued on next page)*

(Contd) Step	Action
7	From the Map To list, select "CUSTNUMAC." This is the field to which the information from the Partner Editor is mapped. 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.

PODATEAC Field

Overview   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 200,000 times.

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. For example, you cannot map the purchase order date in the BEG segment directly to your application PODATEAC field.

Using an extended rule   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 BEG05 element and move it to the PODATEAC field.

Use this procedure to create the extended rule:

Step	Action
1	Double-click the PODATEAC field to access the Field Properties dialog box.
2	Select the **Extended Rule** tab.
3	In the Rule list, type the following: `#PODATEAC=$850.#0323;` 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.

PONUMAC Field

Overview   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.

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. For example, you cannot map the purchase order number in the BEG segment directly to your application PONUMAC field.

Using an extended rule   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 BEG03 element and move it to the PONUMAC field.

Use this procedure to create the extended rule.

Step	Action
1	Double-click the PONUMAC field to access the Field Properties dialog box.
2	Select the **Extended Rule** tab.
3	In the Rule list, type the following: `#PONUMAC=$850.#0324;` 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

Overview   You can map the ACIND (allowance or charge indicator) field directly from the ITA:201 element, which indicates whether the total amount is an allowance or charge.

Procedure   Link the ITA:201 element in the ITA:2 segment to the application field ACIND in the POALLCHG record.

ACMETHAND Field

Overview   You can map the ACMETHAND (allowance or charge method of handling code) directly from the ITA:204 element, which indicates the method of handling the allowance or charge.

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

ACTOTAMT Field

Overview   You can map the ACTOTAMT (allowance or charge total amount) directly from the
                 ITA:207 element.

Procedure   Link the ITA:207 element in the ITA:2 segment to the application field ACTOTAMT in
                 the POALLCHG record.

Mapping the POSUMMARY Record

Overview

Introduction   This section describes 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.

CUSTNUMSUM Field

Overview   The Application Integration subsystem allows you to use selected information from the Gentran:Server for Windows Partner Editor 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 key.

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.

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

Step	Action
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." This indicates that the system updates this field with selected information from the 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." This is the field that contains the customer number for this partner.
	(Continued on next page)

(Contd) Step	Action
7	From the Map To list on the right, select "CUSTNUMSUM." This is the field to which the information from the Partner Editor is mapped. 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.

PODATESUM Field

Overview   You can map the PODATESUM field directly from the BEG05 (Purchase Order Date) field, by using the Link function.

Procedure   Link the BEG05 element in the BEG record to the application field PODATESUM in the POSUMMARY record.

PONUMSUM Field

Overview   You can map the PONUMSUM field directly from the BEG03 (Purchase Order Number) field, by using the Link function.

Procedure   Link the BEG03 element in the BEG record to the application field PONUMSUM in the POSUMMARY record.

NUMLINEITEMSUM Field

Overview   You can map the NUMLINEITEMSUM field directly from the CTT01 (Number of Line Items) field, by using the Link function. The CTT01 element is already directly linked to the NUMLINEITEMS field in the POHEADER record, but Gentran:Server allows you to link an input element to multiple output fields.

Procedure   Link the CTT01 element in the CTT record to the application field NUMLINEITEMSUM in the POSUMMARY record.

After you complete the mapping operation for this final application field, the icons representing each of the application fields should be bisected by a red checkmark. The red checkmark indicates that a mapping operation has been performed against the field.

Finalizing the Map

Overview

Introduction   After you create the map, define your application file format, and perform the appropriate mapping operations, you need to finalize the map. To complete the mapping process, you need to save the map, compile the translation object, print and review the Gentran:Server for Windows report, and test the map.
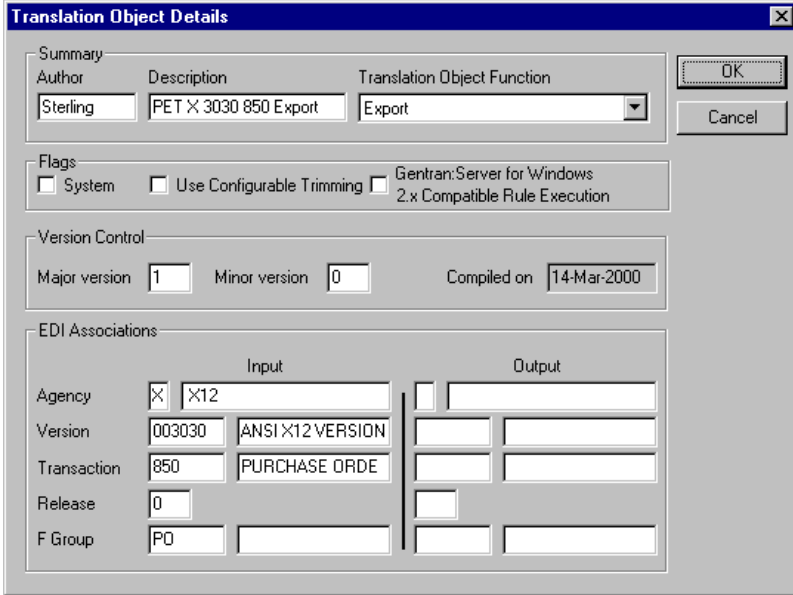
How to Compile the Map

Overview   The Compile function compiles the map and generates a translation object. The map that you created using Gentran:Server for Windows (PET_850.MAP) is a *source map*. When that source map is compiled, the result is a *compiled translation object*.

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

Procedure   Use this procedure to compile a map and generate a translation object.
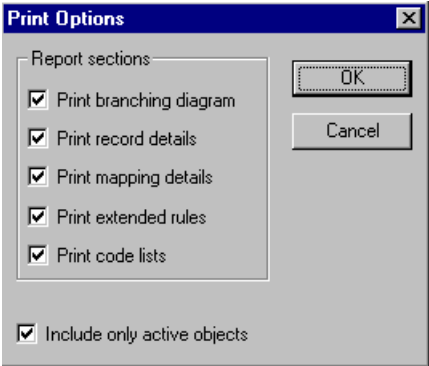
Step	Action
1	From the File menu, select **Save** to save the source map prior to using the Compile function.
2	From the File menu, select **Compile** to display the Run-Time Translation Object Name dialog box.
3	The "File name" box automatically contains "pet_850.tpl." This is the name of the translation object, using the default .TPL file extension and 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. **Caution** Do not overlay the source map with the compiled translation object. Use the .TPL file extension to distinguish the translation object. (Continued on next page)

(Contd) Step	Action
4	Change the Drive and Folder where the compiled translation object is stored, if necessary. **Caution** 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 Gentran:Server for Windows.
5	Click **Save** and 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). Click **OK** to exit the dialog box. The date on which the translation object was compiled is automatically loaded into the Compiled on box on the Translation Object Details dialog box.
6	From the File menu, select **Save** to save the source map with the Compiled on date. **Note** You must register this translation object with the Gentran:Server for Windows system before you can use it. **Reference** See the *Gentran:Server for Windows User Guide* for more information on registering a translation object.

How to Print the Report

Overview   The Gentran:Server for Windows 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).

Procedure   Use this procedure to print the Gentran:Server for Windows report.

Step	Action
1	From the File menu or the Main Toolbar, select Print to access the Print Options dialog box.
2	Accept the defaults on this dialog box (all options checked), so the report includes all sections and only the activated map components. Click **OK** and the Print dialog box is displayed.
3	Set the appropriate options. Click **OK** if you do *not* need to change Setup options, and the mapping report is printed. Click **Setup** if you need to access printer setup information (to select a specific printer, paper orientation, or size and source of paper). The Print Setup dialog box is displayed.
4	Set the appropriate options. Click **OK** if you do *not* need to change Print Options (dithering, intensity control, or print true type as graphics), and you return to the Print dialog box. Click **OK** on the Print dialog box and the mapping report is printed. Click **Options** if you need to access printer setup options information. The Options dialog box is displayed.
5	Set the appropriate options. Click **OK** to return to the Print Setup dialog box. Click **OK** on the Print Setup dialog box to return to the Print dialog. Click **OK** on the Print dialog box and the mapping report is printed.

Testing the Translation Object

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

Procedure   Use this procedure to test the inbound (Export) translation object you just created.

Step	Action
1	Register PET_850.TPL translation object with Gentran:Server for Windows.
2	Import the PETZONE.PAR partner relationship into Gentran:Server for Windows. The default full path for the partner file is: `C:\GENSRVNT\TUTORIAL\PETZONE.PAR`
3	Verify (in Partner Editor) that the PET X 3030 850 Export translation object is selected for the Inbound Relationship.
4	Use the Process File option in Gentran:Server for Windows to process the data file (PETTEST.INT) through the translation object. The data file is located in the TUTORIAL subfolder under the folder where Gentran:Server for Windows is installed. The default full path for the data file is: `C:\GENSRVNT\TUTORIAL\PETTEST.INT`
5	After the document is translated, it is located in the In Documents in Gentran:Server for Windows. Export the document to ensure that it was translated correctly.

Reference
See the *Gentran:Server for Windows User Guide* for more information on registering a translation object, importing a partner, selecting inbound translation objects, and exporting files.

Outbound Invoice (810)

Contents

Overview

Introduction

In this chapter

This tutorial exercise delineates the step-by-step tasks that are necessary to create a map that is used to translate your application file to ANSI X12 invoices (810) that you send to your trading partner.

This chapter explains all the tasks and processes associated with creating a map for an outbound invoice, including the following:

◗ Mapping analysis
◗ Creating the new map
◗ Defining constants
◗ Activating groups, segments, and elements
◗ Defining the application side of the map
◗ Mapping your data
◗ Finalizing the map

Getting Started

Overview

This tutorial exercise 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 (although you may want to verify your completed map with the model).

The purpose of completing each task in this tutorial in sequential order as if it were your own scenario, is to teach you not only how to perform the individual mapping tasks, but also to teach you a logical approach and methods that should be utilized when you create your own map.

For outbound translation, you must set up a system import map and an import map. Generally, you create one system import map for each message type and create one import map for each partner to whom you send messages.

Outbound mapping scenario

In this tutorial exercise, your company is MWT Manufacturing Co., which manufactures a wide variety of supplies for pet stores. Your company sells pet supplies directly to large pet supply retail chains. You need to create a map that enables your company to translate your application file to the invoices that you send to one of your trading partners, Pet Zone (a large pet supply retailer). After you ship an order to Pet Zone, generate the corresponding invoice, and once your application file is translated into an EDI invoice, your company can send the invoice to Pet Zone.

You have an existing application file layout from your accounts payable department, that defines the information your system generates for the invoice. And from discussions with the Pet Zone, you know the requirements for the EDI invoices that you are sending them.

Mapping analysis

The first step in creating a map is the analysis of the mapping requirements. This is the most important step in creating a successful map. If the analysis you perform 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, it is likely that creating the map will be a much longer and arduous task, and the map may be invalid due to oversights and omissions.

This table describes the steps used in mapping analysis. These steps apply to 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 file format and the EDI file.
4	Begin creating the map.

Analyzing Your Application File Format

Overview
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. For an outbound map, your application file contains the data that you need to convert to an EDI file so your system can generate the document correctly. In this tutorial, your application file contains all the information that you need to generate the invoice that you send your partner, so that your company can bill your partner correctly for the goods you shipped to them.

You must define your application file to Gentran:Server for Windows. If your company has an existing application file, as in this tutorial, you should obtain the record layout from the appropriate person. If you did not have an existing application file layout, you would need to create one by determining which fields are necessary to process the data correctly, and then grouping the fields logically under records.

Application file layout: system import map
This table provides the application file layout for the system import map in this tutorial.

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per document					
PARTNER_ID	X	Partner Identifier	String	1	9
STANDARD	X	Standard	String	1	1
VERSION	X	Version	String	6	6
TRANSACTION_SET	X	Transaction Set	String	1	6
TEST_PROD	X	Test/Production Flag	String	1	1

Application file layout: import map
This table provides the application file layout for the import map in this tutorial.

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per document					
PARTNER_ID	X	Partner Identifier	String	1	9
STANDARD	X	Standard	String	1	1
VERSION	X	Version	String	6	6

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
TRANSACTION_ SET	X	Transaction Set	String	1	6
TEST_PROD	X	Test/Production Flag	String	1	1
Beginning Record (INVBEGIN) 1 per document					
BILLTOACCTBEG	X	Bill to Acct Number	String	1	9
DOCTYPBEG	X	Document Type	String	1	6
INVNUMBEG	X	Invoice Number	String	1	10
INVDATE	X	Invoice Date	Dt/Tm (YYMMDD)	6	6
PONUM		Purchase Order Number	String	0	20
PODATE		Purchase Order Date	Dt/Tm (YYMMDD)	6	6
VENDNAME		Vendor Name	String	0	35
VENDADD1		Vendor Address Line 1	String	0	35
VENDADD2		Vendor Address Line2	String	0	35
VENDCITY		Vendor City	String	0	30
VENDSTATE		Vendor State	String	0	2
VENDPOSTAL CODE		Vendor Postal Code	String	0	9
SHIPTONAME		Ship To Name	String	0	35
SHIPTOADD1		Ship To Address Line 1	String	0	35
SHIPTOADD2		Ship To Address Line2	String	0	35
SHIPTOCITY		Ship To City	String	0	30
SHIPTOSTATE		Ship To State	String	0	2
SHIPTOPOSTAL CODE		Ship To Postal Code	String	0	9
SHIPTOLOC		Ship To Store Location	String	0	9
					(Continued on next page)

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
SHIPTODEPTNUM		Ship To Dept Number	String	0	4
REMITTONAME		Remit To Name	String	0	35
REMITTOADD1		Remit To Address Line 1	String	0	35
REMITTOADD2		Remit To Address Line 2	String	0	35
REMITTOCITY		Remit To City	String	0	30
REMITTOSTATE		Remit To State	String	0	2
REMITTOPOSTAL CODE		Remit To Postal Code	String	0	9
BOLNUM		Bill of Lading Number	String	0	12
SHIPDATE		Ship Date	Dt/Tm (YYMMDD)	6	6
TERMS_WHOLE		Terms Percent Whole	Number (N2)	0	3
TERMSNETDUE		Terms Net Due Days	Number (N0)	0	3
TERMSDISDUE		Terms Discount Due Days	Number (N0)	0	3
Comment Group (INVCommentGroup) 1 per document					
Comment Record 1 (INVCOMMENT1) 1 per document					
BILLTOACCTCM1	X	Bill To Acct Number	String	1	9
DOCTYPCM1	X	Document Type	String	1	6
INVNUMCM1	X	Invoice Number	String	1	10
REFNUM	X	Reference Number	String	1	30
Comment Record 2 (INVCOMMENT2) 10 per document					
BILLTOACCTCM2	X	Bill To Acct Number	String	1	9
DOCTYPCM2	X	Document Type	String	1	6
INVNUMCM2	X	Invoice Number	String	1	10
MESSAGETEXT	X	Message Text	String	1	232

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Detail Record (INVDETAIL) up to 200,000 per document					
BILLTOACCTDET	X	Bill To Acct Number	String	1	9
DOCTYPDET	X	Document Type	String	1	6
INVNUMDET	X	Invoice Number	String	1	10
ITEMNUM	X	Item Number (Our Item)	String	1	6
QTYSHIP	X	Quantity Shipped	Number (N0)	1	5
QTYORD	X	Quantity Ordered	Number (N0)	1	5
UNITPRICE	X	Unit Price	Number (N2)	1	7
UOM	X	Unit of Measure	String	1	4
CUSTPROCODE	X	Customer Product Code	String	1	10
UPCCODE	X	UPC Code	String	1	12
ITEMDESC		Item Description	String	0	30
TERMSTYPE		Terms Type Code	String	0	2
TERMSDISP		Terms Discount Percent	Number (N2)	0	6
Summary Record (INVSUMMARY) 1 per document					
BILLTOACCTSUM	X	Bill To Acct Number	String	1	9
DOCTYPSUM	X	Document Type	String	1	6
INVNUMSUM	X	Invoice Number	String	1	10
TOTALAMOUNT	X	Total Monetary Value Summary	Number (N2)	1	10
TRANSTOTAL	X	Transaction Totals	Number (N0)	1	6

Application file definition

The above table contains the minimum information necessary to define your application file to Gentran:Server for Windows (name, mandatory status, description, data type, and field length). Each record contains a group of logically-related application fields. These fields contain the information that your system needs to generate the invoice.

Each record (except the HEADER record, which is used for selecting the appropriate trading relationship) contains a three-field record key that ensures that the same identifying information is located at the beginning of each record. Your application file format contains a record key so that each record can be identified individually. The record key for this invoice consists of:

◗   your partner's bill to account number

◗   the document type

◗   the invoice number.

Record definitions

This table provides the Record Definitions.

Record	Definition
HEADER (header)	Contains the five-field key (partner identifier, standard, version, transaction set, and test/production flag), which is used for selecting the appropriate trading relationship. This header record is mandatory and occurs once for each invoice translated.
INVBEGIN (beginning)	Contains the purchase order number and date; the vendor, ship to, and remit to address information; and the bill of lading number, ship date, and term of the invoice. The beginning record is mandatory and occurs once for each invoice.
INVCommentGroup (comment information)	Groups all the comment information for the invoice, including the reference number and the message text. The INVCommentGroup occurs once per invoice translated. The information within the INVCommentGroup is further segregated by record.
INVCOMMENT1 (first comment)	Contains the reference number. The first comment record is mandatory and occurs once per invoice translated (once for each comment line).
INVCOMMENT2 (second comment)	Contains message text. The second comment record is mandatory and occurs up to ten times per invoice translated (once for each comment line).
	(Continued on next page)

Record	Definition
INVDETAIL (detail)	Contains line item number and description, quantity shipped, quantity ordered, unit of measure, price, product code, and invoice terms. The detail record is mandatory and occurs up to 200,000 times per invoice translated (once for each line item invoiced).
INVSUMMARY (summary)	Contains the total invoice amount and the total transaction amount. The summary record is mandatory and occurs once per invoice translated.

Next step   Now that you have analyzed your application file, the next step is to determine the requirements of the invoice that you send to your partners.

Analyzing the EDI File

Overview   To analyze the EDI file, you must first determine what you need to send your trading partner and what your partner expects to receive. You and your partners must agree on which standard, version, and transaction set you are using. In this tutorial, we are using an ANSI X12 version 003030 810 invoice. It is important to know the information that your partners expects you to send within the invoice, and what data is contained in each element used.

Before you analyze the EDI file, it is helpful to review the EDI standard(s) that you will use. Obtain an EDI standards manual for each standard and version that you are using. Standards manuals are available from EDI standard agencies.

After discussing with your partner what you will send, determining which segments and elements your partner requires, and reviewing the EDI standard, make a list of the map components that you must make available for use (activate).

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.

This table lists and describes the map components required for this tutorial.

Groups		Segment	Element
		BIG (already active)	BIG03 BIG04
		REF	REF02
N1		N1	N102 N302
		N3	N401 N402
		N4	N403 REF02
		REF:2	
		ITD	ITD03 ITD05 ITD07
		DTM	DTM02
			(Continued on next page)

(Contd) Groups		Segment	Element
N9		N9	N902
		MSG	
IT1		IT1	IT101 IT106 IT107 IT108 IT109
		QTY	
	PID	PID:2	PID05
		ITD:2	ITD01 ITD03
		CTT (already active)	CTT02

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

Recommendation
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 Gentran:Server for Windows. This allows you to differentiate between documents in the document browsers in Gentran:Server for Windows.

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. 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 import map in this tutorial is the invoice number.

You can also establish a Document Reference for each map, which also assists you in differentiating between documents in the document browsers in Gentran:Server for Windows. When you set up the Reference in the Application Integration subsystem, the RefData field of the document browsers that contain this document in Gentran:Server for Windows 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.

Defining constants   Since you are creating an import map to create outbound data, you must define constants in the qualifier elements on the EDI (output) side of the map. Part of the mapping analysis for an import map is determining in which elements you must define and use qualifiers, and determining which elements they qualify. You must use constant values that are compliant with the EDI standard.

For this tutorial, you must define the following constants to create the required qualifier elements:

Element Using Constant	Constant ID	Constant Value	Element it Qualifies
REF01	BILL_OF_LADING	BM	REF02
N101 (N1_Vendor)	VENDOR	VN	N102 (N1_Vendor)
N101 (N1_ShipTo)	SHIP_TO	ST	N102 (N1_ShipTo)
REF01 (N1_ShipTo)	DEPT_NUMBER	DP	REF02 (N1_ShipTo)
REF01 (N1_ShipTo)	STORE_NUMBER	SN	REF02 (N1_ShipTo)
N101 (N1_RemitTo)	REMIT_TO	RI	N102 (N1_RemitTo)
DTM01	DATE	DTM	DTM02
N901	VENDOR_NUMBER	1V	N902
IT106	UPC_CODE	UI	IT107
IT108	BUYER_CODE	BP	IT109
QTY01	ORIGINAL_QUANTITY	OR	QTY02
PID01	FREE_FORM	F	PID05

Next step   After you complete the analysis of the EDI file and your application file format, the final step in mapping analysis is correlating the two components.

Correlating Your Application File and the EDI File

Overview　　When you are translating data outbound, you need to build an import map and a system import map. The system import map is used by Gentran:Server for Windows to find the partner relationship for a document, to determine which import map is used to translate the data. The system import map builds the key that the translator uses to find the partner relationship. The sole function of the system import map is to identify the appropriate partner relationship; the system import map does *not* map any data.

There are two ways to build the key in a system import map:

◗　the five-field key (six-field for TRADACOMS standard)

◗　the three-field key

The method that we recommend, and demonstrate in this tutorial, requires six EDI-specific fields in the header record:

◗　partner key

◗　standard

◗　version

◗　transaction set

◗　release (for TRADACOMS only)

◗　test/production status

The combination of these six fields defines a unique key that identifies the appropriate partner relationship. We recommend using this method because it is very flexible. Typically, you use this method when you are defining your application from scratch and can easily add the EDI-specific fields that are not already present in the header record.

The second method is easier to build because it only requires three fields in the header record (partner key, application ID, and application alias value). Please note that an empty string is a valid value in the application ID and application alias value fields. The combination of these three fields defines a unique key that identifies the appropriate partner relationship. However, because only three fields are combining to build the key, this method is not as flexible as the preferred method. Typically, you use this method when you are using a legacy (existing) application definition, and you do not want to add EDI-specific information.

To reconcile your application file with the EDI file, for each application field you must determine which EDI element you need to populate with that information, and how you can map that information to the EDI file. To map information to an element, you can use linking, standard rules, extended rules, or a combination of the three.

Overview (contd)   **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 (hereafter referred to as fields) is visually represented with a line connecting the two fields.

Standard rules give you access to functions that are necessary for mapping operations that are more complex than simple linking, but less involved than extended rules.

Extended rules enable you to use a Gentran:Server for Windows proprietary programming language to perform virtually any mapping operation you require.

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

Mapping operations: system import map

This table lists and describes the mapping operations (by application field) that are required for the system import map in this tutorial.

Field Name	Description	Mapping Operations
Header Record (HEADER) 1 per document		
PARTNER_ID	Partner Identifier	Select standard rule to match the EDI Code in the Partner record (Partner by EDI Code)
STANDARD	Standard	Update standard rule to update the Agency field on the Document record
VERSION	Version	Update standard rule to update the Version field on the Document record
TRANSACTION_SET	Transaction Set	Update standard rule to update the Transaction Set ID field on the Document record
TEST_PROD	Test/Production Flag	Update standard rule to update the Test Mode field on the Document record

Mapping operations: import map

This table lists and describes the mapping operations (by application field) that are required for the import map in this tutorial.

Field Name	Description	Mapping Operations
Header Record (HEADER) 1 per document		
PARTNER_ID	Partner Identifier	No mapping required
STANDARD	Standard	No mapping required
VERSION	Version	No mapping required
TRANSACTION_SET	Transaction Set	No mapping required
TEST_PROD	Test/Production Flag	No mapping required
Beginning Record (INVBEGIN) 1 per document		
BILLTOACCTBEG	Bill to Acct Number	No mapping required
DOCTYPBEG	Document Type	No mapping required
INVNUMBEG	Invoice Number	Link to BIG02
INVDATE	Invoice Date	Link to BIG01
PONUM	Purchase Order Number	Link to BIG04
PODATE	Purchase Order Date	Link to BIG03
VENDNAME	Vendor Name	Link to (N1_Vendor) N102
VENDADD1	Vendor Address Line 1	Link to (N1_Vendor) N301
VENDADD2	Vendor Address Line2	Link to (N1_Vendor) N302
VENDCITY	Vendor City	Link to (N1_Vendor) N401
VENDSTATE	Vendor State	Link to (N1_Vendor) N402
VENDPOSTALCODE	Vendor Postal Code	Link to (N1_Vendor) N403
SHIPTONAME	Ship To Name	Link to (N1_ShipTo) N102
SHIPTOADD1	Ship To Address Line 1	Link to (N1_ShipTo) N301
SHIPTOADD2	Ship To Address Line2	Link to (N1_ShipTo) N302
SHIPTOCITY	Ship To City	Link to (N1_ShipTo) N401
SHIPTOSTATE	Ship To State	Link to (N1_ShipTo) N402

(Contd) Field Name	Description	Mapping Operations
SHIPTOPOSTALCODE	Ship To Postal Code	Link to (N1_ShipTo) N403
SHIPTOLOC	Ship To Store Location	Link to (N1_ShipTo) REF02
SHIPTODEPTNUM	Ship To Dept Number	Link to (N1_ShipTo) REF02
REMITTONAME	Remit To Name	Link to (N1_RemitTo) N102
REMITTOADD1	Remit To Address Line 1	Link to (N1_RemitTo) N301
REMITTOADD2	Remit To Address Line2	Link to (N1_RemitTo) N302
REMITTOCITY	Remit To City	Link to (N1_RemitTo) N401
REMITTOSTATE	Remit To State	Link to (N1_RemitTo) N402
REMITTOPOSTALCODE	Remit To Postal Code	Link to (N1_RemitTo) N403
BOLNUM	Bill of Lading Number	Link to REF02
SHIPDATE	Ship Date	Link to DTM02
TERMS_WHOLE	Terms Percent Whole	Link to ITD03
TERMSNETDUE	Terms Net Due Days	Link to ITD07
TERMSDISDUE	Terms Discount Due Days	Link to ITD05
Comment Group (INVCommentGroup) 1 per document		
Comment Record 1 (INVCOMMENT1) 1 per document		
BILLTOACCTCM1	Bill To Acct Number	No mapping required
DOCTYPCM1	Document Type	No mapping required
INVNUMCM1	Invoice Number	No mapping required
REFNUM	Reference Number	Link to N902
Comment Record 2 (INVCOMMENT2) 10 per document		
BILLTOACCTCM2	Bill To Acct Number	No mapping required
DOCTYPCM2	Document Type	No mapping required
INVNUMCM2	Invoice Number	No mapping required
MESSAGETEST	Message Text	Link to MSG01
		(Continued on next page)

(Contd) Field Name	Description	Mapping Operations
Detail Record (INVDETAIL) up to 200,000 per document		
BILLTOACCT	Bill To Acct Number	No mapping required
DOCTYPDET	Document Type	No mapping required
INVNUMDET	Invoice Number	No mapping required
ITEMNUM	Item Number (Our Item)	Link to IT101
QTYSHIP	Quantity Shipped	Link to IT102
QTYORD	Quantity Ordered	Link to QTY02
UNITPRICE	Unit Price	Link to IT104
UOM	Unit of Measure	Link to IT103
CUSTPROCODE	Customer Product Code	Link to IT107
UPCCODE	UPC Code	Link to IT109
ITEMDESC	Item Description	Link to PID05
TERMSTYPE	Terms Type Code	Link to ITD01
TERMSDISP	Terms Discount Percent	Link to ITD03
Summary Record (INVSUMMARY) 1 per document		
BILLTOACCTSUM	Bill To Acct Number	No mapping required
DOCTYPSUM	Document Type	No mapping required
INVNUMSUM	Invoice Number	No mapping required
TOTALAMOUNT	Total Monetary Value Summary	Link to TDS01
TRANSTOTAL	Transaction Totals	Link to CTT01

Next step   After you finish correlating your application file and EDI file on paper, you can begin creating the system import and import maps.

The System Import Map
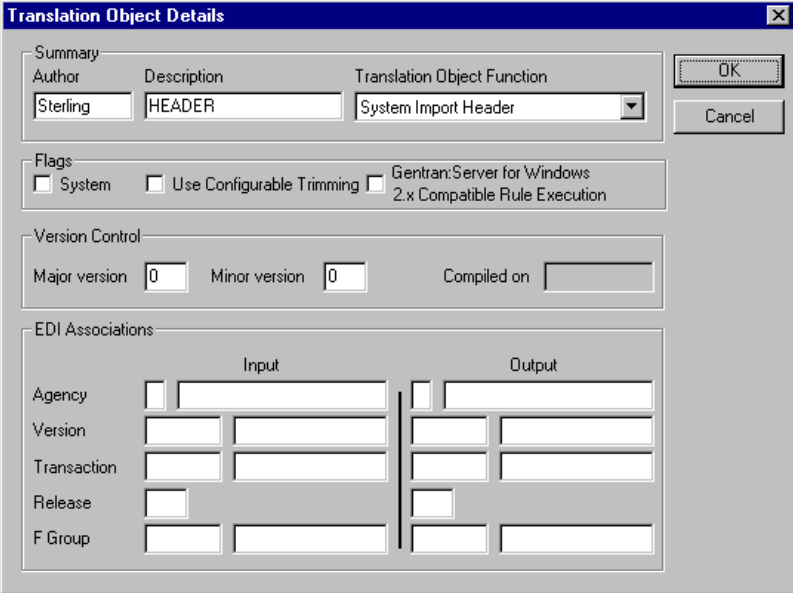
How to Create the System Import Map
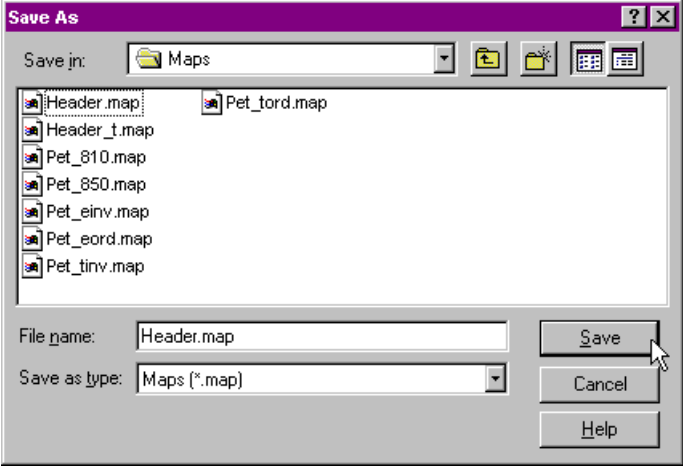
Overview

For outbound translation, you must set up a system import map and an import map. Generally, you create one system import map for each document type, and set up an import map for each partner you are sending documents to.

A system import map is used by the system to find the partner relationship for a document (flat file definition), to determine which import map is used to translate the data. The system import map builds the key that the translator uses to find the partner relationship. The sole function of the system import map is to identify the appropriate partner relationship; the system import map does *not* map any data.

Procedure

Use this procedure to create the system import map.

Step	Action
1	From the File menu, select **New**. **System response** The system displays the New Map Wizard.
2	Answer the following questions and then click **Next**. ◗ *What kind of map are you creating?* Select **System Import Header**. ◗ *What is the name of the map?* Type **HEADER** as the unique name of the map. The system adds the .MAP extension. ◗ *What is your name?* Type your name if it differs from the user name prompted by the system. **System response** The system displays the New Map Wizard - Input Format dialog box. **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 Gentran:Server system.
3	From the Create a new data format using this syntax list, select **Positional** and click **Next**. **System response** The system displays the New Map Wizard - Output Format dialog box. *(Continued on next page)*

(Contd) Step	Action
4	From the Create a new data format using this syntax list, select **Positional** and click **Next**. **System response** The system displays the New Map Wizard.
5	Click **Finish** to create the new map (this may take a few seconds). **System response** The system displays the new map in the Application Integration Window.
6	From the Edit menu, select **Details**. **System response** The system displays the Translation Object Details dialog box.
7	In the Description field, type "System Import HDR". For system import maps, we recommend that you type "System Import*", where the "*" is the identifier for this system import map. The Imports tab of the System Configuration program describes which system import translation object will be used to begin translation during the import process. **Reference** Please see the *Administration Guide* for more information about how your system administrator will need to modify the system configuration program.

(Contd) Step	Action
8	Complete the Major and Minor fields to designate a version number for this translation object. Type "1' as the version number in the Major field and accept the "0" as the release number in the Minor field. **Note** Whenever any major changes are made to your map, increase the version to the next real number (e.g., 2.0, 3.0). If you make a minor change to the map, increase the version by a percentage point (e.g., 2.1, 2.2).
9	Click **OK** to save the translation object details.
10	From the File menu, select **Save**. **System response** The system displays the Save As dialog box. (Continued on next page)

(Contd) Step	Action
11	Verify that the File name box contains Header.map. **Note** You allocate a field name to a map when you save the map for the first time or use the Save As function to save a copy of the map under a different name. The name of the map must be one to eight unique characters that define the map in a meaningful way. The default file extension is ".MAP". Using this extension distinguishes map files from other types of files, such as translation objects. The naming conventions in Gentran:Server for Windows include: ◗ The source map (map created in Gentran:Server for Windows) should be named using the file extension **.MAP** (e.g., HEADER.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** (e.g., HEADER.TPL). **Recommendation** 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. **Caution** Be very careful not to overlay the source map with the compiled translation object. Use the .TPL file extension to distinguish the translation object.
12	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 by selecting Save from the File menu.
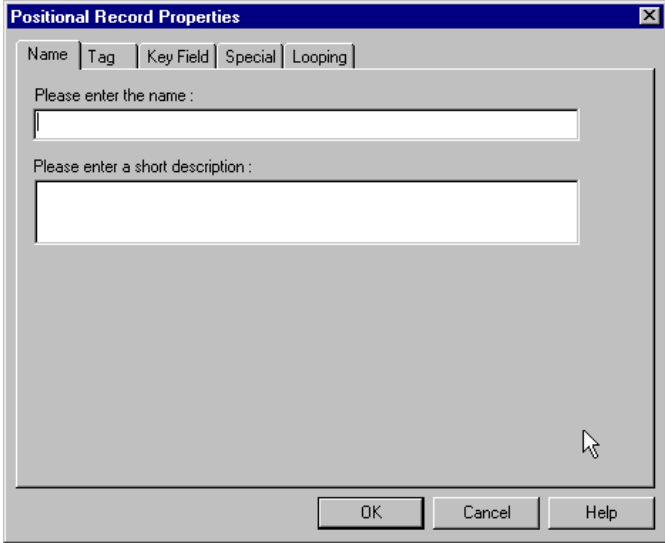
Next step   After creating the system import map, you need to define the header record for the system import map. You use the six-field key (this is the preferred method) in this tutorial.
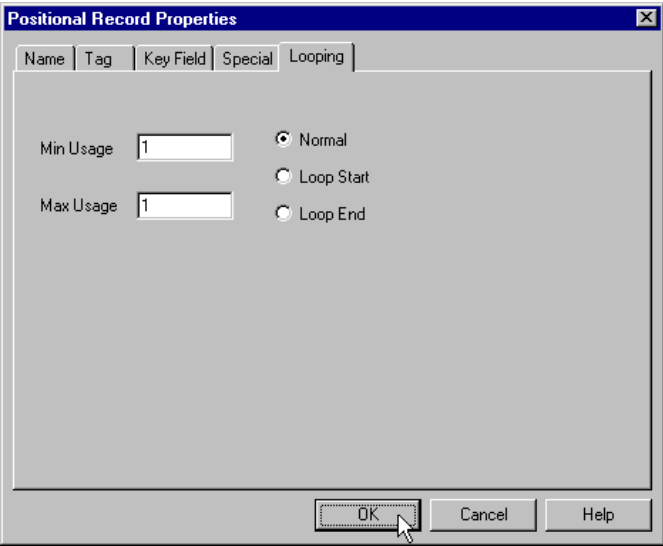
How to Create the HEADER Record

Overview   The only record that you define for the system import map is the header record. The header record is subordinate to the icon on the input side of the map.

The HEADER (header) record contains the five fields that comprise the system import map key. This header record is mandatory and occurs only once.

Procedure   Use this procedure to create the HEADER record.

Step	Action
1	Click the Positional File icon on the input side of the map to highlight it.
2	From the Edit menu, select **Create Sub**. From the submenu, select **Record**. The Positional Record Properties dialog box is displayed.
3	In the Name box, type "HEADER" as the record identification code. **Caution** Do not use spaces or dashes (-) in the Name box. You can use the underscore (_) to separate words.
4	In the Desc box, type "Header Record" as the description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
5	Select the **Tag** tab. (Continued on next page)

(Contd) Step	Action
6	In the Tag box, type "HDR" as the record identification code <TAG>. For example, 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. **Note** You must define the record tag on the Positional Record Properties dialog box for outbound maps, instead of defining fields with the purpose of explicitly containing the record tag.
7	In the Position box, accept the default of "1" for the starting column position of the tag in the data record.
8	Select the **Looping** tab.
9	In the Min Usage box, type "1" to make the header record mandatory. This is the minimum number of times the record must repeat. **Note** if the Min Usage box contains a "0" (zero), the record is conditional. If the Min Usage box contains a "1" or greater, the record is mandatory. The Positional Record Properties dialog box should now look like this:
10	Click **OK** to create the HEADER record.

Next step   Now you have to define the five fields that make up the system import key.

How to Define the HEADER Fields

Overview   To use the preferred method of building the system import key, you need to define at least five fields for the header record (partner key, standard, version, transaction, and test/production status). These fields do not have to be in any order or sequence, but they must be part of the header record.

After you define the header record and fields, you must set up the mapping operations that define the key that the translator uses to find the partner relationship.

Record Layout Table   The easiest way to add fields to a record is to use the Positional Field Editor. This table describes the fields that you need to create.
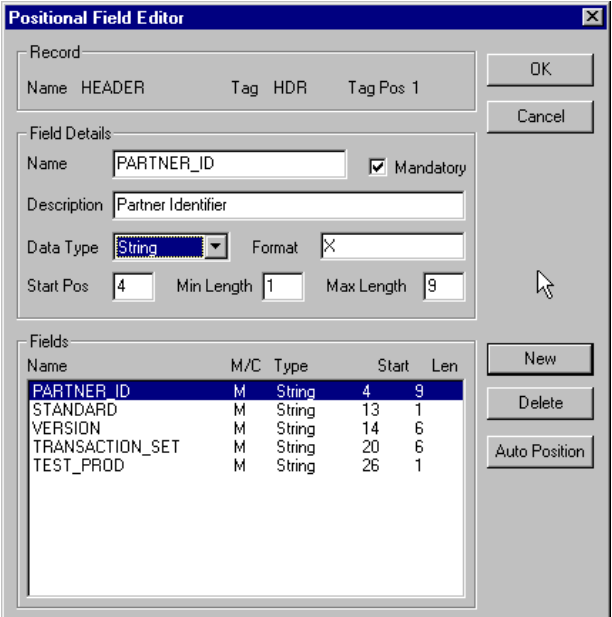
Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per document					
PARTNER_ID	X	Partner Identifier	String	1	9
STANDARD	X	Standard	String	1	1
VERSION	X	Version	String	6	6
TRANSACTION_SET	X	Transaction Set	String	1	6
TEST_PROD	X	Test/Production Flag	String	1	1

Procedure   Use this procedure to create the fields for the HEADER record.

Step	Action
1	Right-click the HEADER record to access the shortcut menu.
2	From the shortcut menu, select **Edit Fields** to access the Positional Field Editor dialog box.
3	The field you are creating is the first field record. Click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type "PARTNER_ID", which is the name of the first field in the HEADER record. Each field must have a unique name.
5	Select the Mandatory check box to identify it as a required field. For the system import map, each field in the key is mandatory.

(Contd) Step	Action
6	In the Description box, type "Partner identifier" as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select "String" as the type of field, to specify that this field must contain alphanumeric data.
8	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list.
9	In the Min Length box, type "1" as the minimum field length.
10	In the max Length box, type "9" as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to the *Record Layout Table* on page 3 - 26. **Note** 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. *(Continued on next page)*

(Contd) Step	Action
13	After adding the last field, click **Auto Position** to automatically position the field in the record. 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 field. Click Yes to acknowledge the warning message that fields are sequenced in order. **Caution** Please note that 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 this tutorial. The Positional Field Editor dialog box should now look like this:
14	When you are completely finished adding fields to the HEADER record, click **Close**.
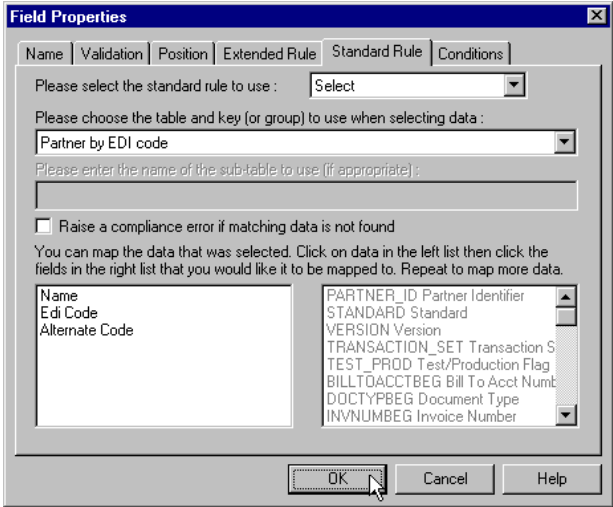
How to Define the Key

Overview   After you define the header record and fields, you can set up the mapping operations that define the key that the translator uses to find the partner relationship. This section provides the procedures to define the:

◗   Partner key

◗   Standard field

◗   Version field

◗   Transaction field

◗   Test/Production field
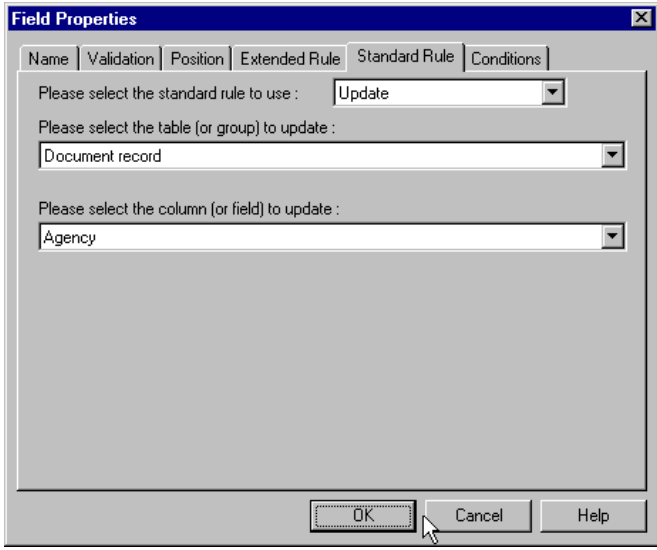
Defining the   Use this procedure to define the partner key.
partner key

Step	Action
1	Double-click the PARTNER_ID field 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 "Partner by EDI Code." This indicates that the system looks up the indicated partner key in the partner database and updates this field. The Standard Rule tab of the Field Properties dialog box should now look like this:
5	Click **OK** to add the standard rule to the partner key field.
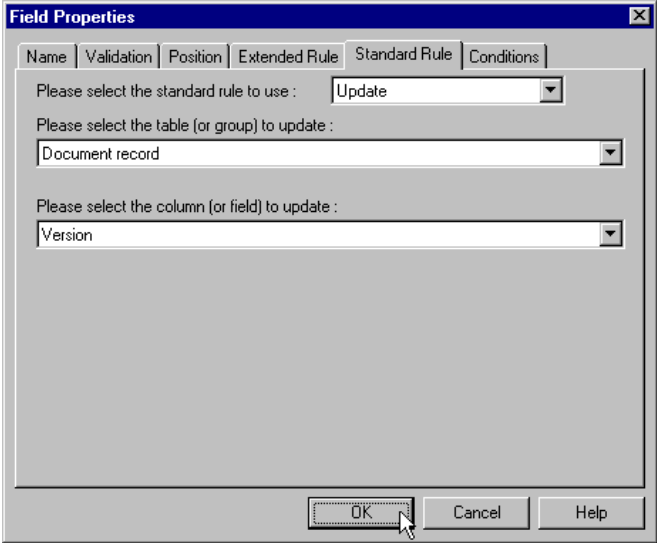
Defining the EDI standard field

Use this procedure to define the EDI standard field.

Step	Action
1	Double-click the STANDARD field to display the Field Properties dialog box.
2	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 wish to store an item of data in the Gentran:Server for Windows tracking database for the document.
5	From the column list, choose "Agency." This indicates that you are updating the agency (standard) field in the system buffer. The Standard Rule tab of the Field Properties dialog box should now look like this:
6	Click **OK** to add the standard rule to the STANDARD field.

Defining the version

Use this procedure to define the version.

Step	Action
1	Double-click the VERSION field to display the Field Properties dialog box.
2	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 wish to store an item of data in the Gentran:Server for Windows tracking database for the document.
5	From the column list, choose "Version." This indicates that you are updating the standard version field in the system buffer. The Standard Rule tab of the Field Properties dialog box should now look like this:
6	Click **OK** to add the standard rule to the VERSION field.
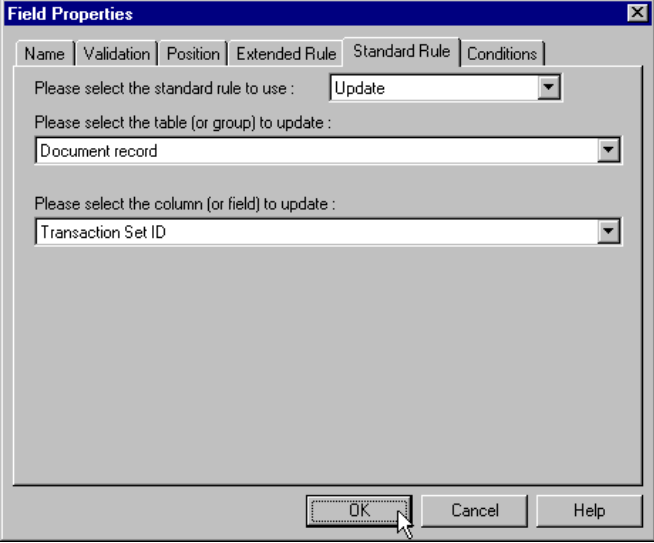
Defining the transaction set

Use this procedure to define the transaction set (document).

Step	Action
1	Double-click the TRANSACTION_SET field to display the Field Properties dialog box.
2	Select the **Standard Rule** tab.
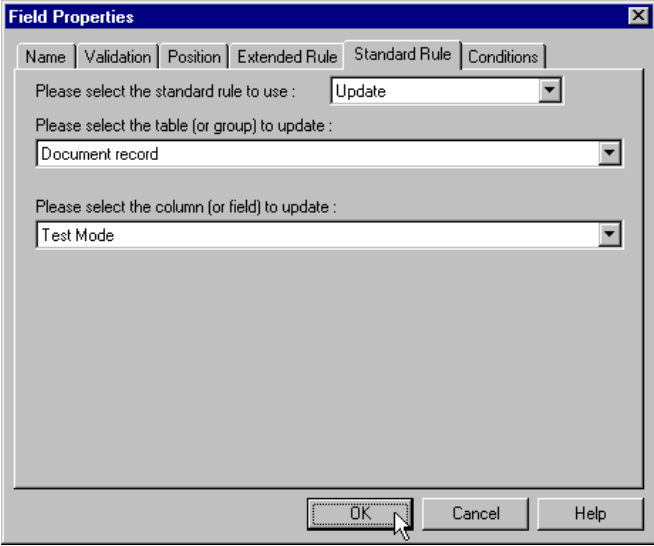
(Contd) Step	Action
3	From the standard rule list, select **Update**.
4	From the table list, choose "Document record." This indicates that you wish to store an item of data in the Gentran:Server for Windows tracking database for the document.
5	From the column list, choose "Transaction Set ID." This indicates that you are updating the transaction set field in the system buffer. The Standard Rule tab of the Field Properties dialog box should now look like this: ![Field Properties dialog box] **Field Properties** Name | Validation | Position | Extended Rule | Standard Rule | Conditions Please select the standard rule to use : Update Please select the table (or group) to update : Document record Please select the column (or field) to update : Transaction Set ID OK Cancel Help
6	Click **OK** to add the standard rule to the TRANSACTION_SET field.

**Defining the test/
production status
field**

Use this procedure to define the test/production status field.

Step	Action
1	Double-click the TEST_PROD field to display the Field Properties dialog box.
2	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 wish to store a item of data in the Gentran:Server for Windows tracking database for the document. *(Continued on next page)*
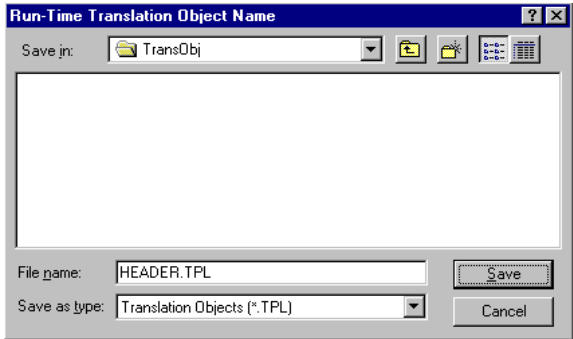
(Contd) Step	Action
5	From the column list, select "Test Mode." This indicates that you are updating the release field in the system buffer. The Standard Rule tab of the Field Properties dialog box should now look like this:
6	Click **OK** to add the standard rule to the TEST_PROD field.
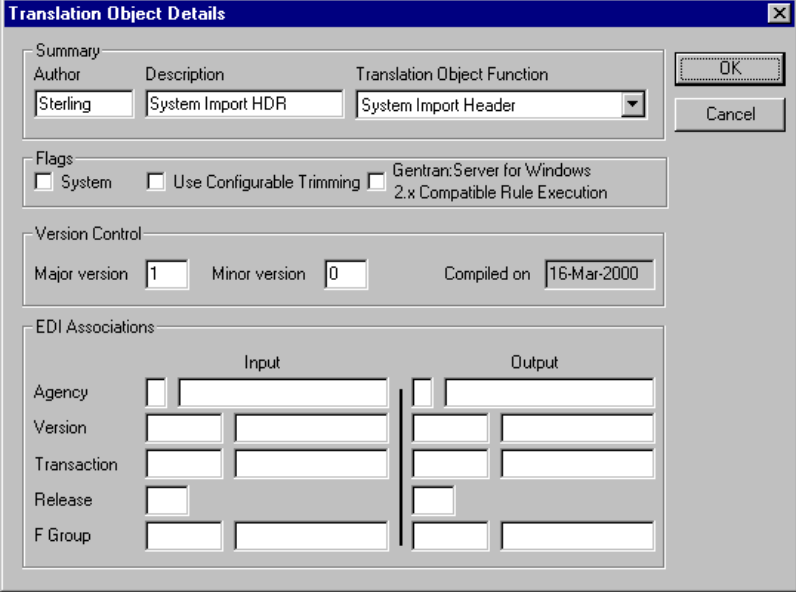
How to Compile the Map

Overview   The Compile function compiles the map and generates a translation object. The map that you created using the Application Integration subsystem (HEADER.MAP) is a *source map*. When that source map is compiled, the result is a *compiled translation object*.

After you save the translation object, you must register it with Gentran:Server for Windows before you can use it.

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

Step	Action
1	From the File menu, select **Save** to save the source map, prior to using the Compile function.
2	From the File menu, select **Compile** to display the Run-Time Translation Object name dialog box.
3	The File name box automatically contains "header.tpl." This is the name of the translation object, using the default .TPL extension and 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 extensions) means that the relationship between the source map and the compiled translation object remains evident. **Caution** Do not overlay the source map with the compiled translation object. Use the .TPL file extension to distinguish the translation object.
4	Change the Drive and Folder where the compiled translation object is stored, if necessary. *(Continued on next page)*
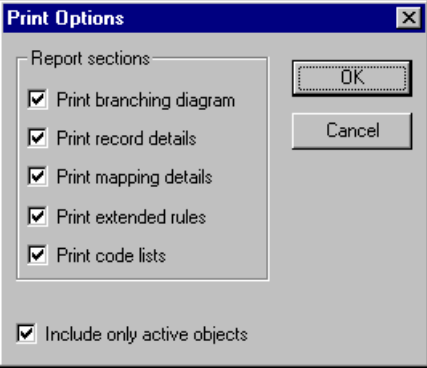
(Contd) Step	Action
5	Click **Save** and the system compiles the map and generates a translation object. The Compile Error dialog box is displayed.

Verify that no errors occurred and click **OK** to exit the dialog box. The date on which the translation object was compiled is automatically loaded into the Compiled on box on the Translation Object Details dialog box.

 |
| 6 | From the File menu, select Save to save the source map with the Compiled on date.

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

Reference
See the *Gentran:Server for Windows User Guide* for information on registering a translation object. |

How to Print the Report

Overview   The Gentran:Server for Windows 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).

Procedure   Use this procedure to print the Gentran:Server for Windows report.

Step	Action
1	From the File menu or Main Toolbar, select Print to access the Print Options dialog box.
2	Accept the defaults on this dialog box (all options checked), so the report includes all sections and only the activated map components. Click **OK** and the Print dialog box is displayed.
3	Set the appropriate options. Click **OK** if you do *not* need to change Setup options, and the mapping report is printed. Click **Setup** if you need to access printer setup information (to select a specific printer, paper orientation, or size and source of paper). The Print Setup dialog box is displayed.
4	Set the appropriate options. Click **OK** if you do *not* need to change Print Options (dithering, intensity control, or print true type as graphics), and you return to the Print dialog box. Click **OK** on the Print dialog box and the mapping report is printed. Click **Options** if you need to access the printer setup options information. The Options dialog box is displayed.
5	Set the appropriate options. Click **OK** to return to the Print Setup dialog box. Click **OK** on the Print Setup dialog box to return to the Print dialog box. Click **OK** on the Print dialog box and the mapping report is printed.

Testing the translation object

After you compile the map, print and verify the report, and register the translation object with Gentran:Server for Windows, you should test the translation object to verify that the data is translated correctly. You test the system import translation object (HEADER.TPL) when you test the import translation object (on 3 - 148).

The Import Map
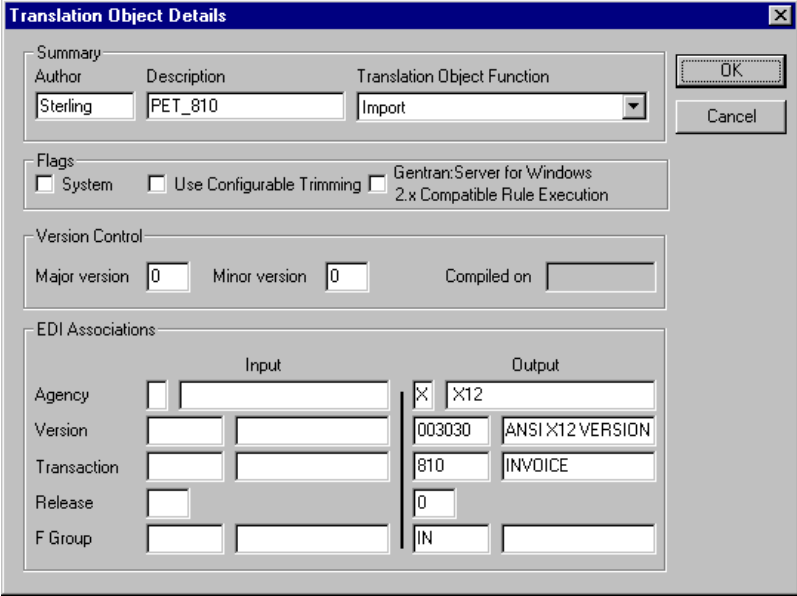
How to Create the Import Map
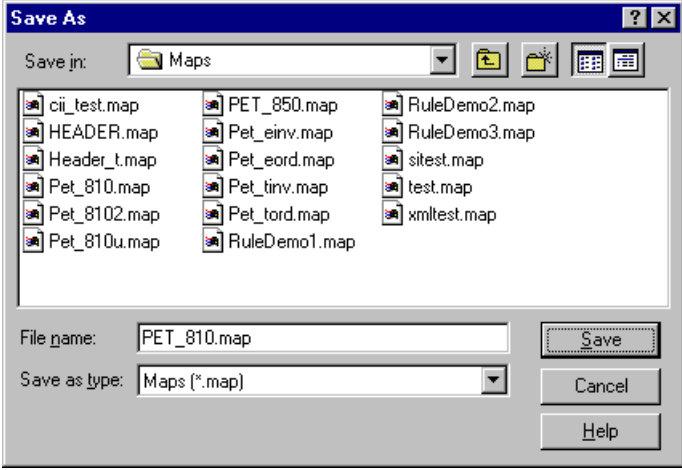
Overview   The first task in creating a new map is to define the format that is used for the input and output sides of the map. In this example, you are creating an import map so Gentran:Server for Windows can translate your application file to an invoice that you are sending to your partner. Therefore, the input side of the map needs to reflect your application file. The output side of the map needs to reflect the ANSI X12 810 invoice that you send to your partner. Instead of defining the 810 document to the system yourself, you want Gentran:Server for Windows to use the standards database on your system to build the invoice according to the standard.

Procedure   Use this procedure to create the import map for the outbound invoice.

Step	Action
1	From the File menu, select **New**. **System response** The system displays the New Map Wizard.
2	Answer the following questions and then click **Next**. ◗ *What kind of map are you creating?* Select **Import**. ◗ *What is the name of the map?* Type **PET_810** as the unique name of the map. The system adds the .MAP extension. ◗ *What is your name?* Type your name if it differs from the user name prompted by the system. **System response** The system displays the New Map Wizard - Input Format dialog box. **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 Gentran:Server system.
3	From the Create a new data format using this syntax list, select **Positional** and click **Next**. **System response** The system displays the New Map Wizard - Output Format dialog box.

(Contd) Step	Action
4	From the Create a new data format using this syntax list, select **Delimited EDI** and click **Customize**. **System response** The system displays the New Delimited EDI Wizard.
5	Click **Next**.
6	From the ODBC data source list, select **Gentran:Server Standards** (if you defined your EDI standards data source with another name, select that name) and click **Next**.
7	Select the following: a. Select the **[X] X12** standards agency, b. **[003030] ANSI X12 VERSION 3 Release 3** version, c. **{810] INVOICE** transaction set, and d. **9** release (for TRADACOMS only). Click **Next**.
5	Click **Finish**. **System response** The system loads the selected transaction set and returns to the New Map Wizard - Output Format dialog box.
6	Click **Next**. **System response** The system displays the New Map Wizard.
7	Click **Finish** to create the new map (this may take a few seconds). **System response** 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. *(Continued on next page)*

(Contd) Step	Action
8	From the Edit menu, select **Details**. **System response** The system displays the Translation Object Details dialog box.
9	In the Description field, type **PET X 3030 810 Import**.
10	Complete the Major and Minor fields to designate a version number for this translation object. Type "1' as the version number in the Major field and accept the "0" as the release number in the Minor field. **Note** Whenever any major changes are made to your map, increase the version to the next real number (e.g., 2.0, 3.0). If you make a minor change to the map, increase the version by a percentage point (e.g., 2.1, 2.2).
11	Click **OK** to save the translation object details. (Continued on next page)

(Contd) Step	Action
12	From the File menu, select **Save**. **System response** The system displays the Save As dialog box.
13	Verify that the File name box contains **pet_810.map**. **Note** You allocate a field name to a map when you save the map for the first time or use the Save As function to save a copy of the map under a different name. The name of the map must be one to eight unique characters that define the map in a meaningful way. The default file extension is ".MAP". Using this extension distinguishes map files from other types of files, such as translation objects. The naming conventions in Gentran:Server for Windows include: ◗ The source map (map created in Gentran:Server for Windows) should be named using the file extension **.MAP** (e.g., PET_810.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** (e.g., PET_810.TPL). **Recommendation** 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. **Caution** Be very careful not to overlay the source map with the compiled translation object. Use the .TPL file extension to distinguish the translation object. *(Continued on next page)*

(Contd) Step	Action
14	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 by selecting Save from the File menu.

Next step   Now that you've finished creating and saving the new Export map, you need to define the Input and Output sides of the map.

How to Activate the EDI Map Components

Overview   This table describes the map components for an EDI File.

Icon	Description
✉	EDI File
▢	Segment
🄖	Loop (Group)
🎨	Composite Element
🌿	Element

This table describes the map components for a Positional File.

Icon	Description
🛢	Positional File
⬤	Record
🄖	Loop (Group)
🌿	Field

The specific EDI map components that you use depend on the type of map you are creating, the standard, version, and transaction set (document) selected, and which segments and elements your company requires. We recommend that you determine which segments and elements you are using before starting this step.

When Gentran:Server for Windows generates the EDI sides of the map, 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. The system activates all the groups, segments, composites, and elements that are defined as "mandatory" (must be present) by the standard. The system does not enable you to deactivate the mandatory groups, segments, composites, and elements. By default, Gentran:Server for Windows 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.

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

Step	Action
1	Double-click the segments and groups in the table below to open them. These are the groups, subgroups, segments and elements that you must activate. 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.

Groups		Segment	Element
		BIG (already active)	BIG03 BIG04
		REF	REF02
N1		N1 N3 N4 REF:2	N102 N302 N401 N402 N403 REF02
		ITD	ITD03 ITD05 ITD07
		DTM	DTM02
N9		N9	N902
		MSG	
IT1		IT1	IT101 IT106 IT107 IT108 IT109
		QTY	
	PID	PID:2	PID05
		ITD:2	ITD01 ITD03
		CTT (already active)	CTT02

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

Step	Action
2	Click the **Activate** button on the Main Toolbar.

(Contd) Step	Action
3	Click each inactive group that you need to use. This activates the groups. **Note** If you click a group, segment, or element that you did not mean to activate, click it with the right mouse button to access the shortcut menu. From the shortcut menu, select Deactivate from the shortcut menu.
4	Click each inactive segment that you need to use. This activates the segments.
5	Click each inactive element that you need to use. This activates the elements.
6	Once you have activated all the necessary groups, segments, and elements, click the **Activate** button on the Main Toolbar to turn activation mode off.
7	From the View menu, select Show Active Only. **Note** This function instructs the system to only display the active map components.

How to Define Literal Constants
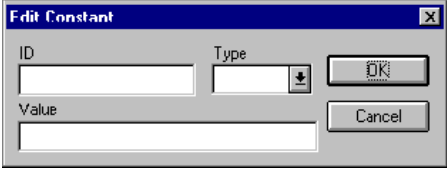
Overview   Literal constants are used by the system as a repository to store information that is used later in the map. For an outbound map, constants are typically used to define qualifying relationships. A qualifying relationship establishes a relationship between an element and its qualifier. A qualifier contains a code that further defines the element or the elements with which it is associated. Qualifying relationships are typically defined in outbound maps. Qualifying relationships are defined using an standard rule, as explained later in this tutorial.

Since you are creating an import map to translate outbound data, you must define constants to define the qualifying relationships on the EDI (output) side of the map. After you determine which constants are used in a map, it saves time later in the map creation process if you define all the constants at one time.

This table lists the constants you must define for this tutorial.

Constant ID	Constant Type	Constant Value
BILL_OF_LADING	String	BM
VENDOR	String	VN
SHIP_TO	String	ST
DEPT_NUMBER	String	DP
STORE_NUMBER	String	SN
REMIT_TO	String	RI
DATE	String	DTM
VENDOR_NUMBER	String	1V
UPC_CODE	String	UI
BUYER_CODE	String	BP
ORIGINAL_QUANTITY	String	OR
FREE_FORM	String	F

Procedure　　Use this procedure to create the literal constants that are used in the import map for this tutorial.

Step	Action
1	From the Edit menu, select **Constants** to display the Map Constants dialog box.
2	Click **New** to access the Edit Constant dialog box.
3	In the ID box, type "BILL_OF_LADING", which is the first literal constant identifier in the table. This is typically a description of the field or element in which the constant is used.
4	From the Type list, select "String" as the category of this literal constant.
5	In the Value box, type "BM" as the actual constant expression. This is the value of the literal constant.
6	Click **OK** to add the constant to the system.
7	Repeat steps 2 – 6 to create the rest of the constants, according to the table on page 3 - 46.
8	Click **Close** to exit the Map Constants dialog box.

How to Verify EDI Delimiters

Overview   If you are using an EDI standard that contains composite elements or subelements, you must verify that Gentran:Server for Windows 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). Since 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.

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

Procedure   Use this procedure to verify EDI delimiters.

Step	Action
1	Right-click the 810 INVOICE file icon to access the shortcut menu. From the shortcut menu, select Properties to display the File Properties dialog box.
2	Select the **Delimiters** tab. **System response** The system displays the delimiter settings. *(Continued on next page)*

(Contd) Step	Action
3	Verify the required delimiters for the ANSI X12 standard. If the delimiters differ from the default in this table, type either the character or the hexadecimal value in the correct box. The default delimiters for ANSI X12 are listed and described in this table (the tag delimiter, release character, and decimal separator are not used for ANSI). <table><tr><th>Box</th><th>ANSI X12 Default Character</th><th>ANSI X12 Default Hex Value</th></tr><tr><td>Tag Delimiter</td><td></td><td></td></tr><tr><td>Element Delimiter</td><td>* (asterisk)</td><td>0x2A</td></tr><tr><td>Segment Delimiter</td><td>~ (tilde)</td><td>0x7E</td></tr><tr><td>Sub Element Delimiter</td><td>: (colon)</td><td>0x3A</td></tr><tr><td>Release Character</td><td></td><td></td></tr><tr><td>Decimal Separator</td><td></td><td></td></tr></table>
4	Click **OK** to exit the File Properties dialog box.

Next step   Now you can define your application file layout to Gentran:Server for Windows.

Application File Format

Defining the Rest of Your Application File Format

Overview   At this point, you have already defined the header record (in the system import map) that you can copy and use in your import map. Now you must define the rest of your application file to Gentran:Server for Windows. In the import map, the input side of the map is positional. Your application file must contain all the information that you need to generate the EDI document that you are sending to your partner.

Before you define your application file format, you should obtain a layout of the necessary records, fields, and groups. The records contain related fields and the groups contain related records. Each level of your application file format must be created sequentially.

For example, your application file format contains records and groups. The records contain fields and the group contains records. This means that you must create records and the group before you create the subordinate fields and records.

How to Copy the HEADER Record

Overview   Since you have already defined the header record for your application file format in the system import map, you can copy that header record to the import map. The copied record contains subordinate fields, which is also be copied. Copied map components retain all the information of the original map component, so you must remove the standard rules you created for the fields in the header record for the system import map.

Procedure   Use this procedure to copy the HEADER record from the system import map to the import map.

Step	Action
1	Open the HEADER.MAP system import map, if it is not already open.
2	Highlight the HEADER record.
3	Click the **Copy** button on the Main Toolbar.
4	Open the PET_810.MAP import map, if it is not already open.
5	Highlight the INPUT positional file icon that you want the copied record pasted *after*.
6	From the Main Toolbar, click **Paste** to past the contents of the Clipboard. The HEADER record is pasted as a subordinate map component to the INPUT positional file icon.

Removing the HEADER standard rules   The HEADER record that you just copied from the system import map contains five fields. The first field (PARTNER_ID) has a Select standard rule applied to it in the system import map. The remaining four fields in the copied HEADER record in the import map (STANDARD, VERSION, TRANSACTION_SET, and TEST_PROD) have Update standard rules applied to them in the system import map. These standard rules are not applicable to the import map, so you must remove them.

Procedure   Use this procedure to remove the standard rules from the HEADER record in the import map.

Step	Action
1	Double-click the PARTNER_ID field in the HEADER record to display the Field Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **<none>**.

(Continued on next page)

Step	Action
4	Depending on how you set the confirmation options on the Preferences dialog box (Confirmations tab), the system may prompt you to confirm removal of the standard rule from the field. If so, click **Yes** to remove the standard rule.
5	Click **OK** to save the change to the field.
6	Repeat steps 1 - 5 for the STANDARD, VERSION, TRANSACTION_SET, and TEST_PROD fields.
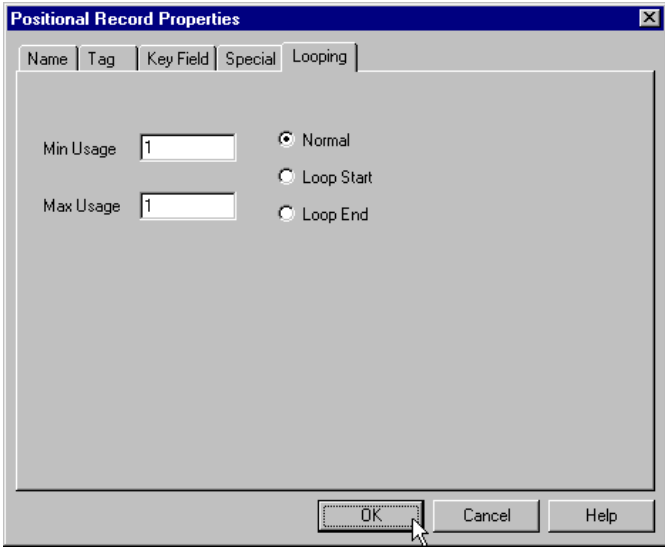
How to Create the INVBEGIN Record

Overview   The second record in your application file format you need to create INVBEGIN is at the same level as the HEADER record.

The INVBEGIN (beginning) record contains the purchase order number and date; the vendor, ship to, and remit to address information; and the bill of lading number, ship date, and terms of the invoice. The beginning record is mandatory and occurs once for each invoice.

Procedure   Use this procedure to create the INVBEGIN record.

Step	Action
1	Highlight the HEADER record. From the Edit menu, select **Insert**. From the submenu, select **Record** to display the Positional Record Properties dialog box. **Note** Use "Insert" from the Edit menu when you want to create equal records/ groups and to use "Create Sub" to create subordinate groups/records/ fields.
2	In the Name box, type "INVBEGIN" as the record identification code. **Caution** Do not use spaces or dashes (-) in the Name box. You can use the underscore (_) to separate words.
3	In the Desc box, type "Beginning Record" as the description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the **Tag** tab.
5	In the Tag box, type "BEG" as the record identification code <TAG>. For example, 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. **Note** You must define the record tag on the Positional Record Properties dialog box for outbound maps, instead of defining fields with the purpose of explicitly containing the record tag. (Continued on next page)

(Contd) Step	Action
6	In the Position box, accept the default of "1" for the staring column position of the tag in the data record.
7	Select the **Looping** tab.
8	In the Min Usage box, type "1" to make the record mandatory. This box contains the minimum number of times the record must repeat.
	Note
	If the Min Usage box contains a "0" (zero), the record is conditional. If the Min Usage box contains a '1' or greater, the record is mandatory.
	Accept the default of "1" in the Max Usage box so the beginning record only occurs once (not a looping structure).
	The Positional Record Properties dialog box should now look like this:
9	Click **OK** to create the INVBEGIN record.

How to Create the INVCommentGroup

Overview　　The INVCommentGroup (comment information) groups all the comment information for the invoice, including the reference number and the message text. The INVCommentGroup occurs once per invoice translated. The information within the INVCommentGroup is further segregated by related records.

Procedure　　Use this procedure to create the INVCommentGroup.

Step	Action
1	To create a group that is at the same level as the INVBEGIN record, highlight the INVBEGIN record. From the Edit menu, select **Insert**. From the submenu, select **Group**. **System response** The Group Properties dialog box is displayed.
2	In the Name box, type "INVCommentGroup" to indicate that this group contains all of the comment records. This is the segment loop identification code.
3	In the Description box, type "Comment Group" as the loop description. This box is used to provide a brief explanation of the loop.
4	Select the **Looping** tab.
5	In the Minimum Usage box, type "1" to make the loop mandatory. This is the minimum number of times the loop must be repeated. For a conditional loop, the minimum usage should always be "0" (zero). For a mandatory loop, the minimum usage should be "1" or greater.
6	Accept the default of "1" in the Maximum Usage box. This indicates that this group of records occurs only once.
7	Click **OK** to create the INVCommentGroup.

How to Create the INVCommentGroup Subordinate Records

Overview

In this tutorial, the INVCommentGroup contains two subordinate records:

◗   INVCOMMENT1
◗   INVCOMMENT2.

After you create the first subordinate record (INVCOMMENT1), you can define the subsequent record at the same level (INVCOMMENT2).
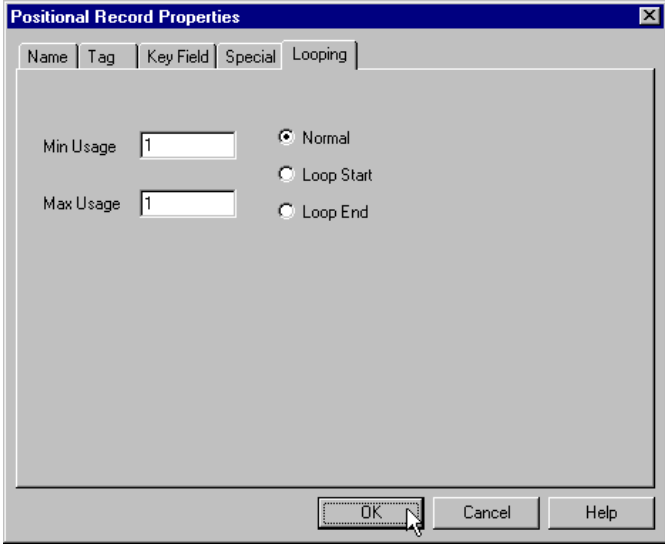
Creating the INVCOMMENT1 record

The INVCOMMENT1 (first comment) record contains the reference number. The first comment record is mandatory and occurs once per invoice translated (once for each comment line).

Procedure

Use this procedure to create the INVCOMMENT1 record.

Step	Action
1	Click the INVCommentGroup to highlight it. From the Edit menu, select **Create Sub**. From the submenu, select **Record**. **System response** The Positional Record Properties dialog box is displayed.
2	In the Name box, Type "INVCOMMENT1" as the record identification code.
3	In the Desc box, type "Header Comment Record 1" as the description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the **Tag** tab.
5	In the Tag box, type "CM1" as the record identification code <TAG>. For example, 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.
6	In the Position box, accept the default of "1" for the starting column position of the tag in the data record.
7	Select the **Looping** tab. (Continued on next page)

(Contd) Step	Action
8	In the Min Usage box, type "1" to make the record mandatory. This box contains the minimum number of times the record must repeat. Accept the default of "1" in the Max Usage box so this record only occurs once. The Positional Record Properties dialog box should now look like this.
9	Click **OK** to create the INVCOMMENT1 record.

Creating the INVCOMMENT2 record

The INVCOMMENT2 (second comment) record contains message text. The second comment record is mandatory and occurs up to ten times per invoice translated (once for each comment line).

Procedure

Use this procedure to create the INVCOMMENT2 record.

Step	Action
1	Highlight the INVCOMMENT1 record. From the Edit menu, select **Insert**. From the submenu, select **Record** to display the Positional Properties dialog box.
2	In the Name box, type "INVCOMMENT2" as the record identification code.
3	In the Desc box, type "Header Comment 2" as a description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records. *(Continued on next page)*

(Contd) Step	Action
4	Select the **Tag** tab.
5	In the Tag box, type "CM2" as the record identification code <TAG>. For example, 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.
6	In the Position box, accept the default of "1" as the starting column position of the tag in the data record.
7	Select the **Looping** tab.
8	In the Min Usage box, type "1" to make the record mandatory. This box contains the minimum number of times the record must repeat.
9	In the Max Usage box, type "10" as the maximum number of times the record can repeat.
10	Click **OK** to create the INVCOMMENT2 record.

How to Create the INVDETAIL Record

Overview　　The INVDETAIL (detail) record is at the same level as the HEADER and INVBEGIN records and the INVCommentGroup.

The INVDETAIL record contains line item number and description, quantity shipped, quantity ordered, unit of measure, price, product code, and invoice terms. The detail record is mandatory and occurs up to 200,000 times per invoice translated (once for each line item invoiced).

Procedure　　Use this procedure to create the INVDETAIL record.

Step	Action
1	Highlight the INVCommentGroup. From the Edit menu, select **Insert**. From the submenu, select **Record** to display the Positional Record Properties dialog box.
2	In the Name box, type "INVDETAIL" as the record identification code.
3	In the Desc box, type "Detail Record" as a description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the **Tag** tab.
5	In the Tag box, type "DET" as the record identification code <TAG>. For example, 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.
6	In the Position box, accept the default of "1" as the starting column position of the tag in the data record.
7	Select the **Looping** tab.
8	In the Min Usage box, type "1" to make the record mandatory. This box contains the minimum number of times the record must repeat.
9	In the Max Usage box, type "200000" as the maximum number of times the record can repeat.
10	Click **OK** to create the INVDETAIL record.

How to Create the INVSUMMARY Record

Overview   The final record that you need to create is the INVSUMMARY (summary) record, which
is at the same level as the INVDETAIL record.

The INVSUMMARY (summary) record contains the total invoice amount and the total
transaction amount. The summary record is mandatory and occurs once per invoice
translated.

Procedure   Use this procedure to create the INVSUMMARY record.

Step	Action
1	Highlight the INVDETAIL record. From the Edit menu, select **Insert**. From the submenu, select **Record** to display the Positional Record Properties dialog box.
2	In the Name box, type "INVSUMMARY" as the record identification code.
3	In the Desc box, type "Summary Record" as a description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the **Tag** tab.
5	In the Tag box, type "SUM" as the record identification code <TAG>. For example, 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.
6	In the Position box, accept the default of "1" as the starting column position of the tag in the data record.
7	Select the **Looping** tab.
8	In the Min Usage box, type "1" to make the record mandatory. This box contains the minimum number of times the record must repeat.
9	Accept the default of "1" in the Max Usage box. This is the maximum number of times the record can repeat.
10	Click **OK** to create the INVSUMMARY record.

Now that you have created the record structure of the application file, you can define the
application fields.

Current map structure

The application side of the map (input) should now look like this:
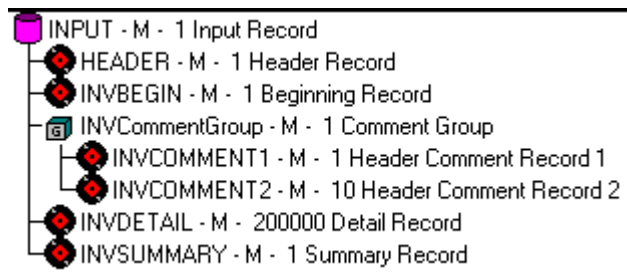
INPUT - M - 1 Input Record
    HEADER - M - 1 Header Record
    INVBEGIN - M - 1 Beginning Record
    INVCommentGroup - M - 1 Comment Group
        INVCOMMENT1 - M - 1 Header Comment Record 1
        INVCOMMENT2 - M - 10 Header Comment Record 2
    INVDETAIL - M - 200000 Detail Record
    INVSUMMARY - M - 1 Summary Record

Creating the Application Fields

Overview

Introduction   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 invoice.

The easiest way to add application fields to a record is to use the Positional Field Editor. Generally, you create the fields for each sequential record, starting with the INVBEGIN record.

Record Layout Table   This table lists and describes the application fields that you need to create.

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per document					
PARTNER_ID	X	Partner Identifier	String	1	9
STANDARD	X	Standard	String	1	1
VERSION	X	Version	String	6	6
TRANSACTION_SET	X	Transaction Set	String	1	6
TEST_PROD	X	Test/Production Flag	String	1	1
Beginning Record (INVBEGIN) 1 per document					
BILLTOACCTBEG	X	Bill to Acct Number	String	1	9
DOCTYPBEG	X	Document Type	String	1	6
INVNUMBEG	X	Invoice Number	String	1	10
INVDATE	X	Invoice Date	Dt/Tm (YYMMDD)	6	6
PONUM		Purchase Order Number	String	0	20
PODATE		Purchase Order Date	Dt/Tm (YYMMDD)	6	6
VENDNAME		Vendor Name	String	0	35

(Continued on next page)

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
VENDADD1		Vendor Address Line 1	String	0	35
VENDADD2		Vendor Address Line2	String	0	35
VENDCITY		Vendor City	String	0	30
VENDSTATE		Vendor State	String	0	2
VENDPOSTALCODE		Vendor Postal Code	String	0	9
SHIPTONAME		Ship To Name	String	0	35
SHIPTOADD1		Ship To Address Line 1	String	0	35
SHIPTOADD2		Ship To Address Line2	String	0	35
SHIPTOCITY		Ship To City	String	0	30
SHIPTOSTATE		Ship To State	String	0	2
SHIPTOPOSTAL CODE		Ship To Postal Code	String	0	9
SHIPTOLOC		Ship To Store Location	String	0	9
SHIPTODEPTNUM		Ship To Dept Number	String	0	4
REMITTONAME		Remit To Name	String	0	35
REMITTOADD1		Remit To Address Line 1	String	0	35
REMITTOADD2		Remit To Address Line 2	String	0	35
REMITTOCITY		Remit To City	String	0	30
REMITTOSTATE		Remit To State	String	0	2
REMITTOPOSTAL CODE		Remit To Postal Code	String	0	9
BOLNUM		Bill of Lading Number	String	0	12
SHIPDATE		Ship Date	Dt/Tm (YYMMDD)	6	6
TERMS_WHOLE		Terms Percent Whole	Number (N2)	0	3
TERMSNETDUE		Terms Net Due Days	Number (N0)	0	3
TERMSDISDUE		Terms Discount Due Days	Number (N0)	0	3

(Continued on next page)

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Comment Group (INVCommentGroup) 1 per document					
Comment Record 1 (INVCOMMENT1) 1 per document					
BILLTOACCTCM1	X	Bill To Acct Number	String	1	9
DOCTYPCM1	X	Document Type	String	1	6
INVNUMCM1	X	Invoice Number	String	1	10
REFNUM	X	Reference Number	String	1	30
Comment Record 2 (INVCOMMENT2) 10 per document					
BILLTOACCTCM2	X	Bill To Acct Number	String	1	9
DOCTYPCM2	X	Document Type	String	1	6
INVNUMCM2	X	Invoice Number	String	1	10
MESSAGETEXT	X	Message Text	String	1	232
Detail Record (INVDETAIL) up to 200,000 per document					
BILLTOACCTDET	X	Bill To Acct Number	String	1	9
DOCTYPDET	X	Document Type	String	1	6
INVNUMDET	X	Invoice Number	String	1	10
ITEMNUM	X	Item Number (Our Item)	String	1	6
QTYSHIP	X	Quantity Shipped	Number (N0)	1	5
QTYORD	X	Quantity Ordered	Number (N0)	1	5
(Continued on next page)					
UNITPRICE	X	Unit Price	Number (N2)	1	7
UOM	X	Unit of Measure	String	1	4
CUSTPROCODE	X	Customer Product Code	String	1	10
UPCCODE	X	UPC Code	String	1	12
ITEMDESC		Item Description	String	0	30
TERMSTYPE		Terms Type Code	String	0	2
TERMSDISP		Terms Discount Percent	Number (N2)	0	6
(Continued on next page)					

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Summary Record (INVSUMMARY) 1 per document					
BILLTOACCTSUM	X	Bill To Acct Number	String	1	9
DOCTYPSUM	X	Document Type	String	1	6
INVNUMSUM	X	Invoice Number	String	1	10
TOTALAMOUNT	X	Total Monetary Value Summary	Number (N2)	1	10
TRANSTOTAL	X	Transaction Totals	Number (N0)	1	6
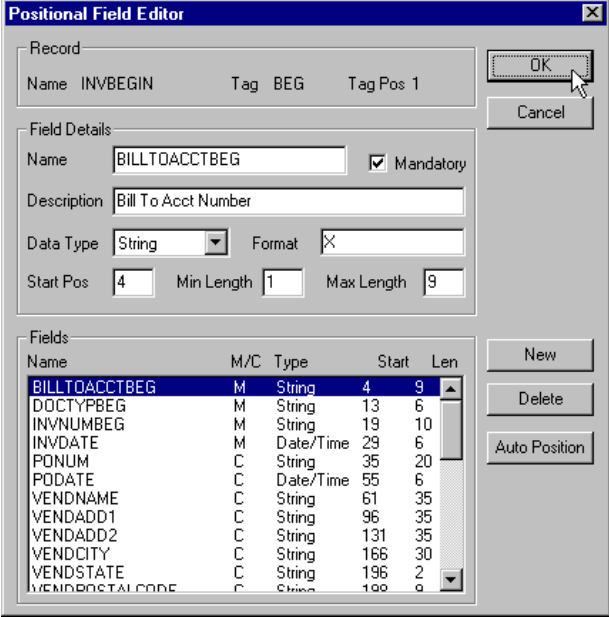
How to Create the INVBEGIN Record Application Fields

Overview   You already created the fields for the HEADER record in the system import map, and then copied the fields to the import map and removed the standard rules from those fields. Therefore, the first record for which you need to define fields, is the INVBEGIN record.

Procedure   Use this procedure to create the application fields for the INVBEGIN record.

Step	Action
1	Right-click the INVBEGIN application record to access the shortcut menu.
2	From the shortcut menu, select **Edit Fields** to access the Positional Field Editor dialog box.
3	If the field you are creating is the first field in the record, click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type "BILLTOACCTBEG", which is the name of the first field in the INVBEGIN record. 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, "BEG" for the Beginning record.
5	Select the Mandatory check box to identify it as a required field.
6	In the Description box, type "Bill To Account Number" as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select "String" as the type of the field, to specify that this field must contain alphanumeric data. **Note** If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You want to specify field start positions if, for example, you are only using a few fields but you want them positioned exactly in the record. In this tutorial, you use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see step 14 below for more information about the Auto Position function.
8	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list. *(Continued on next page)*

(Contd) Step	Action
9	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.
10	In the Max Length box, type "9" as the maximum length of the field.
11	To add the field and create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to the *Record Layout Table* on page 3 - 62. **Note** 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. (Continued on next page)

(Contd) Step	Action
13	After adding the last field, click Auto Position to automatically position the fields in the record. 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. Click Yes to acknowledge the warning message that fields are sequenced in order. **Caution** Please note that 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. The Positional Field Editor dialog box should now look like the following:
14	When you are completely finished adding fields to the INVBEGIN record, click **Close**.

How to Create the INVCOMMENT1 Record Application Fields

Procedure   Use this procedure to create the application fields for the INVCOMMENT1 record.

Step	Action
1	Right-click the INVCOMMENT1 application record to access the shortcut menu.
2	From the shortcut menu, select **Edit Fields** to access the Positional Field Editor dialog box.
3	If the field you are creating is the first field in the record, click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type "BILLTOACCTCM1", which is the name of the first field in the INVCOMMENT1 record. 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, "CM1" for the first comment record.
5	Select the Mandatory check box to identify it as a required field.
6	In the Description box, type "Bill To Account Number" as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select "String" as the type of the field, to specify that this field must contain alphanumeric data.
8	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list.
9	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.
10	In the Max Length box, type "9" as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field. *(Continued on next page)*

(Contd) Step	Action
12	Create the rest of the fields according to the *Record Layout Table* on page 3 - 62. **Note** 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.
13	After adding the last field, click **Auto Position** to automatically position the fields in the record. 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. Click Yes to acknowledge the warning message that fields are sequenced in order.
14	When you are completely finished adding fields to the INVCOMMENT1 record, click **Close**.

How to Create the INVCOMMENT2 Record Application Fields

Procedure　Use this procedure to create the application fields for the INVCOMMENT2 record.

Step	Action
1	Right-click the INVCOMMENT2 application record to access the shortcut menu.
2	From the shortcut menu, select **Edit Fields** to access the Positional Field Editor dialog box.
3	If the field you are creating is the first field in the record, click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type "BILLTOACCTCM2", which is the name of the first field in the INVCOMMENT2 record. 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, "CM2" for the second comment record.
5	Select the Mandatory check box to identify it as a required field.
6	In the Description box, type "Bill To Account Number" as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select "String" as the type of the field, to specify that this field must contain alphanumeric data.
8	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list.
9	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.
10	In the Max Length box, type "9" as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field. (Continued on next page)

(Contd) Step	Action
12	Create the rest of the fields according to the *Record Layout Table* on page 3 - 62. **Note** 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.
13	After adding the last field, click Auto Position to automatically position the fields in the record. 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. Click Yes to acknowledge the warning message that fields are sequenced in order.
14	When you are completely finished adding fields to the INVCOMMENT2 record, click **Close**.

How to Create the INVDETAIL Record Application Fields

Procedure   Use this procedure to create the application fields for the INVDETAIL record.

Step	Action
1	Right-click the INVDETAIL application record to access the shortcut menu.
2	From the shortcut menu, select **Edit Fields** to access the Positional Field Editor dialog box.
3	If the field you are creating is the first field in the record, click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type "BILLTOACCTDET", which is the name of the first field in the INVDETAIL record. 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, "DET" for the detail record.
5	Select the Mandatory check box to identify it as a required field.
6	In the Description box, type "Bill To Account Number" as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select "String" as the type of the field, to specify that this field must contain alphanumeric data.
8	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list.
9	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.
10	In the Max Length box, type "9" as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field. *(Continued on next page)*

(Contd) Step	Action
12	Create the rest of the fields according to the *Record Layout Table* on page 3 - 62. **Note** 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.
13	After adding the last field, click Auto Position to automatically position the fields in the record. 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. Click Yes to acknowledge the warning message that fields are sequenced in order.
14	When you are completely finished adding fields to the INVDETAIL record, click **Close**.

How to Create the INVSUMMARY Record Application Fields

Procedure   Use this procedure to create the application fields for the INVSUMMARY record.

Step	Action
1	Right-click the INVSUMMARY application record to access the shortcut menu.
2	From the shortcut menu, select **Edit Fields** to access the Positional Field Editor dialog box.
3	If the field you are creating is the first field in the record, click **New**. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type "BILLTOACCTSUM", which is the name of the first field in the INVSUMMARY record. 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, "SUM" for the summary record.
5	Select the Mandatory check box to identify it as a required field.
6	In the Description box, type "Bill To Account Number" as the description of the field.
7	From the Data Type list, select "String" as the type of the field, to specify that this field must contain alphanumeric data.
8	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list.
9	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.
10	In the Max Length box, type "9" as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press **Enter**. A new field with blank values ready for you to identify is positioned below the field. *(Continued on next page)*

(Contd) Step	Action
12	Create the rest of the fields according to the *Record Layout Table* on page 3 - 62. **Note** 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.
13	After adding the last field, click Auto Position to automatically position the fields in the record. 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. Click Yes to acknowledge the warning message that the fields are sequenced in order.
14	When you are completely finished adding fields to the INVSUMMARY record, click **Close**.

Mapping Your Data

Overview

Introduction

Each record in the import map contains a group of logically-related application fields. These fields contain the information that your system needs to generate the invoice. The group contains logically-related application records.

Each record (except the HEADER record that you defined for the system import map using an explicit six-field key) contains a three-field record key that ensures that the same identifying information is located at the beginning of each record. Your application file format contains a record key so that each record can be identified individually. The record key for this invoice consists of your partner's bill to account number, the document type, and the invoice number.

This section examines each application field sequentially and describes the mapping operations that you need to perform to map the appropriate data from each field.

HEADER record

For the import map, you will not map from any of the fields in the HEADER (header) record. This record contains the five fields that comprise the key in the system import map. However, since this header record is a part of your application file, it must be defined in the import map, as well. You did this when you copied the HEADER record from the system import map to this import map. This header record is mandatory and occurs only once.

BILLTOACCTBE G field

The BILLTOACCTBEG field is part of the application file format key, but does not contain data that the system needs to use to generate the invoice. Therefore, you do not perform any mapping operations on this field.

DOCTYPBEG field

The DOCTYPBEG field is part of the application file format key, but does not contain data that the system needs to use to generate the invoice. Therefore, you do not perform any mapping operations on this field.

How to Map the INVBEGIN Record

Overview

Introduction   This section describes the actions necessary for you to map from each field in the INVBEGIN (beginning) record. The INVBEGIN record contains the purchase order number and date; the vendor, ship to, and remit to address information; and the bill of lading number, ship date, and term of the invoice. The beginning record is mandatory and occurs once for each invoice.

How to Map the INVNUMBEG Field

Overview
You can map the INVNUMBEG field directly to the BIG02 (Invoice Number) field, by using the 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. The link between the two fields is visually represented with a line connecting the two fields.

Also, 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 Gentran:Server. This allows you to differentiate between documents in the document browsers in Gentran:Server.

Recommendation
We strongly recommend that you select an element or field that occurs only once in the document and has a data type of "String." For an Invoice, the document name is typically the Invoice Number.

Mapping the invoice number
Link the application field INVNUMBEG in the INVBEGIN record to the BIG02 element in the BIG segment.
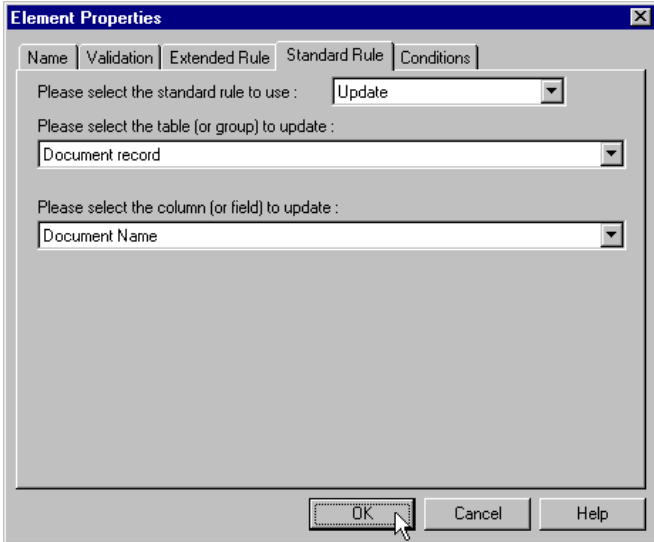
Setting up the document name
To set up a Document Name, you must select an element or field in the map that contains data that distinguishes the documents translated by this map. Then, the Name field of the document browsers that contain this document contains the data from the selected element or field.

Procedure
Use this procedure to set up the Document Name for this Invoice.

Step	Action
1	Double-click the BIG02 (0076) element to display the Element Properties dialog box
2	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 Gentran:Server for Windows.

(Contd) Step	Action
5	From the column list, select "Document Name." This indicates that you are updating the Name field in the document browsers with the contents of this element. The Standard Rule tab of the Element Properties dialog box should now look like this:
6	Click **OK** to set up the document name.

How to Map the INVDATE Field

Overview   You can map the INVDATE field directly to the BIG01 (Invoice Date) field, by using the Link function.

Procedure   Link the application field INVDATE in the INVBEGIN record to the BIG01 element in the BIG segment.

How to Map the PONUM Field

Overview   You can map the PONUM field directly to the BIG04 (Purchase Order Number) field, by using the 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. The link between the two fields is visually represented with a line connecting the two fields.

Also, we recommend that you set up a Document Reference for each map, to make the identification of a document created by this map easier in Gentran:Server for Windows. This allows you to differentiate between documents in the document browsers in Gentran:Server for Windows.

Recommendation
You can choose any element or field as the Document Reference, 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." For an Invoice, the document reference is typically the Purchase Order Number.

Procedure   Link the application field PONUM in the INVBEGIN record to the BIG04 element in the BIG segment.

How to Set Up the Document Reference

Overview   To set up a Document Reference, you must select an element or field in the map that contains data that distinguishes the documents translated by this map. If you set up the Reference in the Application Integration subsystem, the RefData field of the document browsers that contain this document in Gentran:Server for Windows contains the data from that mapped field.

Procedure   Use this procedure to set up the Document Reference for this invoice.

Step	Action
1	Double-click the BIG04 (0324) element to display the Element Properties dialog box.
2	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 Gentran:Server for Windows.
5	From the column list, select "Reference Data." This indicates that you are updating the RefData field in the document browsers with the contents of this element.
6	Click **OK** to set up the document reference.

How to Map the PODATE Field

Overview   You can map the PODATE field directly to the BIG03 (Purchase Order Date) field, by using the Link function.

Procedure   Link the application field PODATE in the INVBEGIN record to the BIG03 element in the BIG segment.

Mapping Shipping and Billing Information

Overview

Introduction   A loop is a group of two or more segments that repeat, in sequence, more than once. The N1 group contains vendor, shipping, and billing name and address information, and repeats multiple times. Your application file contains Vendor, Ship To, and Remit To information that you need to map to the N1 group. However, you can only map fields and elements that are at the same level. This means that you cannot map from a non-repeating record to a group. Specifically, you cannot map the vendor, shipping, and remittance information in the INVBEGIN record directly to the N1 group, because the INVBEGIN record does not repeat. Also, you cannot map to the N3 (address information) repeating segment from your application file Ship To fields (which are in the INVBEGIN record that does not repeat).

Therefore, to map vendor, shipping, and remittance information from your application file format, you must create three single iterations of the N1 group, so that the links between the application fields in the INVBEGIN record and the N1 groups are valid. The Split function extracts the specified number of iterations (instances) of a group (or repeating segment). This enables you to map unique data from your application file and enter a specialized definition for each N1 group (N1_Vendor, N1_ShipTo, and N1_RemitTo).

You can then map a constant to the Entity Identifier Code (in the N1 segment) for each iteration of the N1 group. If the Entity Identifier Code is "VN," that iteration of the N1 group contains shipping information. You need to map the Vendor name and address fields in your application file format to the N1_Vendor group. If the Entity Identifier Code is "ST," that iteration of the N1 group contains shipping information. You need to map the Ship To name and address fields in your application file format to the N1_ShipTo group. If the Entity Identifier Code is "RI," that iteration of the N1 group contains remittance information. You need to map the remittance name and address fields in your application file format to the N1_RemitTo group.

How to Change the Max Use of the N3 Segment

Overview   You cannot map from your application file format Vendor, Ship To, and Remit To fields (which are in the INVBEGIN record that does not repeat) to the N3 (address information) repeating segment. Therefore, before you split the N1 group, you need to change the maximum usage of the N3 segment from "2" to "1" (to make the N3 a non-repeating segment for each iteration of the group).
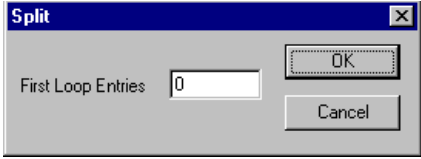
Procedure   Use this procedure to make the N3 a non-repeating segment.

Step	Action
1	Right-click the N3 segment (in the N1 group) to access the shortcut menu.
2	From the shortcut menu, select **Properties** to access the EDI Segment Properties dialog box.
3	Select the **Looping** tab.
4	In the Max Usage box, type "1" to make the N3 a non-repeating segment. This box is currently set to "2."
5	Click **OK** to accept the change to the N3 segment.

How to Split the N1 Group

Overview　　You cannot map from a non-repeating record (INVBEGIN) to a group (N1). Therefore, to map vendor, shipping, and remittance information from your application file format, you must split three single iterations of the N1 group, so that the links between the application fields in the INVBEGIN record and the N1 groups are valid.

Procedure　　Use this procedure to map to split three iterations of the N1 group and add specialized definitions.

Step	Action
1	Highlight the N1 group.
2	Click the Split button on the Main Toolbar. The Split dialog box is displayed.
3	In the First Loop Entries box, type "1" to indicate the sequential number of iterations where you want the group or repeating segment split.
4	Click **OK** to exit the Split dialog box.
5	From the Edit menu, select **Properties** to access the Group Properties dialog box for the new group.
6	In the Name box, type "N1_Vendor", which is a specialized definition to identify this group as containing vendor information.
7	Click **OK** to save the change and exit the Group Properties dialog box.
8	Highlight the original N1 group (that now repeats 199 times) again.
9	Click the Split button on the Main Toolbar to extract another single instance of the N1. The Split dialog box is displayed.
10	In the First Loop Entries box, type "1" to indicate the sequential number of iterations where you want the group or repeating segment split.
11	Click **OK** to exit the Split dialog box.
12	From the Edit menu, select **Properties** to access the Group Properties dialog box for the new group.

(Continued on next page)

(Contd) Step	Action
13	In the name box, type "N1_ShipTo", which is a specialized definition to identify this group as containing shipping information.
14	Click **OK** to save the change and exit the Group Properties dialog box.
15	Highlight the original N1 group (that now repeats 198 times) again.
16	Click the Split button on the Main Toolbar to extract another single instance of the N1. The Split dialog box is displayed.
17	In the First Loop Entries box, type "1" to indicate the sequential number of iterations where you want the group or repeating segment split.
18	Click **OK** to exit the Split dialog box.
19	From the Edit menu, select **Properties** to access the Group Properties dialog box for the new group.
20	In the Name box, type "N1_RemitTo", which is the specialized definition to identify this group as containing remittance information.
21	Click **OK** to save the change and exit the Group Properties dialog box for the new group.
22	Click the **Deactivate** button on the Main Toolbar and then click the original N1 group (that now repeats 197 times), so the system does not process it.
23	Click the REF:6 segment (in the N1_Vendor group) so the system does not process it.
24	Click the **Deactivate** button on the Main Toolbar again to turn the Deactivation mode off.
25	Double-click the N1_RemitTo group to open it.
26	Click the **Deactivate** button.
27	Click the REF:8 segment (in the N1_RemitTo group) so the system does not process it.
28	Click the **Deactivate** button on the Main Toolbar again to turn the Deactivation mode off.

How to Set the Entity Identifier Code

Overview

You need to set the Entity Identifier Code for the N1_Vendor, N1_ShipTo, and N1_RemitTo groups, so the system is able to distinguish the type of name and address information each N1 group should contain.

You already defined all the constants that you are using in this map, so now you can map the appropriate constant to each Entity Identifier Code element.

Setting the entity identifier code for N1_Vendor group

Use this procedure to set the Entity Identifier Code for the N1_Vendor group.

Step	Action
1	Double-click the N1:301 element (Entity Identifier Code) in the N1:3 segment to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.
4	From the Constants list, select the "VN" constant, to identify this iteration of the N1 group as containing vendor name and address information.
5	From the Qualifies list, select the "0093:5 NAME," to set up a qualifying relationship with the Name element. **Note** The Qualifies list contains only the other *active* elements in the same segment as the qualifying element.
6	Click **OK** and the "VN" code is loaded to the Entity Identifier Code element and the qualifying relationship between the N1:301 and N1:302 elements is established.

Setting the entity identifier code for N1_ShipTo group

Use this procedure to set the Entity Identifier Code for the N1_ShipTo group.

Step	Action
1	Double-click the N1:401 element (Entity Identifier Code) in the N1:4 segment to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.
4	From the Constants list, select the "ST" constant, to identify this iteration of the N1 group as containing shipping name and address information.

(Contd) Step	Action
5	From the Qualifies list, select the "0093:21 NAME," to set up a qualifying relationship with the Name element.
6	Click **OK** and the "ST" code is loaded in the Entity Identifier Code element and the qualifying relationship between the N1:401 and N1:402 elements is established.

Setting the entity identifier code for N1_RemitTo group

Use this procedure to set the Entity Identifier Code for the N1_RemitTo group.

Step	Action
1	Double-click the N1:501 element (Entity Identifier Code) in the N1:5 segment to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.
4	From the Constants list, select the "RI" constant, to identify this iteration of the N1 group as containing vendor name and address information.
5	From the Qualifies list, select "0093:85 NAME," to set up a qualifying relationship with the Name element.
6	Click **OK** and the "RI" code is loaded in the Entity Identifier Code element and the qualifying relationship between the N1:501 and the N1:502 elements is established.

How to Map the Vendor Application Fields

Overview　　　Now you can link the elements in your Vendor application fields in the single occurrence of the INVBEGIN record to the appropriate single occurrence of the N1 that contains vending information (N1_Vendor). You can link the elements in your Ship To application fields in the single occurrence of the INVBEGIN record to the appropriate single occurrence of the N1 that contains shipping information (N1_ShipTo). And, you can link the elements in your RemitTo application fields in the single occurrence of the INVBEGIN record to the appropriate single occurrence of the N1 that contains remittance information (N1_RemitTo).

Application fields and elements　　　This table lists the application fields in the INVBEGIN record and the elements they are mapped to.

To map the application field	Link the field to the
VENDNAME	N1:302 element in the N1:3 segment (N1_Vendor group).
VENDADD1	N3:301 element in the N3:3 segment (N1_Vendor group).
VENDADD2	N3:302 element in the N3:3 segment (N1_Vendor group).
VENDCITY	N4:301 element in the N4:3 segment (N1_Vendor group).
VENDSTATE	N4:302 element in the N4:3 segment (N1_Vendor group).
VENDPOSTALCODE	N4:303 element in the N4:3 segment (N1_Vendor group).
SHIPTONAME	N1:402 element in the N1:4 segment (N1_ShipTo group).
SHIPTOADD1	N3:401 element in the N3:4 segment (N1_ShipTo group).
SHIPTOADD2	N3:402 element in the N3:4 segment (N1_ShipTo group).
SHIPTOCITY	N4:401 element in the N4:4 segment (N1_ShipTo group).
SHIPTOSTATE	N4:40 element in the N4:4 segment (N1_ShipTo group).
SHIPTOPOSTAL CODE	N4:403 element in the N4:4 segment (N1_ShipTo group).

Mapping Shipping Application Fields

Overview

Introduction   The SHIPTOLOC (ship to location) and the SHIPTODEPTNUM (ship to department number) fields both contain additional reference information about the Ship To address. Both fields are in the INVBEGIN record that occurs once. To map both these fields to the REF:7 (reference) segment in the N1_ShipTo group, which repeats twelve times, you must promote a single instance of the REF segment, so that two target fields exists to which you can map the SHIPTOLOC and SHIPTODEPTNUM. And, because the original REF segment (REF:7) still repeats eleven times, you must change the maximum usage to one. Therefore, a link between the INVHEADER and the original REF segment (REF:7) is valid.

You must also map a constant to the Reference Number Qualifier (REF01 element) and define a qualifying relationship with the Reference Number (REF02 element), for both instances of the REF segment (REF:9 and REF:7). If the Reference Number Qualifier is "DP" (department number), that iteration of the REF segment contains the ship to department number. If the Reference Number Qualifier is "ST" (store number), that iteration of the REF segment contains the ship to location.

How to Promote the REF Segment

Overview   To map both the SHIPTOLOC and SHIPTODEPTNUM fields to the REF (reference) segment in the N1_ShipTo group, which repeats twelve times, you must promote a single instance of the REF segment, so that two target fields exists to which you can map the SHIPTOLOC and SHIPTODEPTNUM.

Procedure   Use this procedure to map to promote a single instance of the REF segment.

Step	Action
1	Highlight the REF:7 segment in the N1_ShipTo group.
2	Click the Promote button on the Main Toolbar to extract a single instance of the REF segment. The REF:9 segment that only occurs once is extracted.

How to Change the Max Use of the REF:7 Segment

Overview   The original REF segment (REF:7) still repeats eleven times, so you need to change the maximum usage to one. Therefore, a link between the INVHEADER and the original REF segment (REF:7) is valid.

Procedure   Use this procedure to make the REF:7 a non-repeating segment.

Step	Action
1	Right-click the REF:7 Segment (in the N1_ShipTo group) to access the shortcut menu.
2	From the shortcut menu, select **Properties** to access the EDI Segment Properties dialog box.
3	Select the **Looping** tab.
4	In the Max Usage box, type "1" to make the REF:7 a non-repeating segment. This box is currently set to "11."
5	Click **OK** to accept the change to the REF:7 segment.

How to Set the Reference Number Qualifier

Overview

You must map a constant to the Reference Number Qualifier (REF01 element) and define a qualifying relationship with the Reference Number (REF02 element), for both instances of the REF segment (REF:9 and REF:7).

You already defined all the constants that you are using in this map, so now you can map the appropriate constant to each Reference Number Qualifier element.

Procedure (for REF:9)

Use this procedure to set the Reference Number Qualifier for the REF:9 segment.

Step	Action
1	Double-click the REF:901 element (Reference Number Qualifier) in the REF:9 segment in the N1_ShipTo group, to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.
4	From the Constants list, select the "DP" constant, to identify this iteration of the REF segment as containing the ship to department number.
5	From the Qualifies list, select "0127:12 REFERENCE NUMBER," to set up a qualifying relationship with the Reference number element.
6	Click **OK** and the "DP" code is loaded in the Reference Number Qualifier element and the qualifying relationship between REF:901 and REF:902 elements is established.

Procedure (for REF:7)

Use this procedure to set the Reference Number Qualifier for the REF:7 segment.

Step	Action
1	Double-click the REF:701 element (Reference Number Qualifier) in the REF:7 segment in the N1_ShipTo group, to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.
4	From the Constants list, select the "SN" constant, to identify this iteration of the REF segment as containing the store number.

(Contd) Step	Action
5	From the Qualifies list, select the "0127:10 REFERENCE NUMBER," to set up a qualifying relationship with the Reference Number element.
6	Click **OK** and the "SN" code is loaded in the Reference Number Qualifier element and the qualifying relationship between the REF:701 and REF:702 elements is established.

How to Map the Shipping Reference Information

Overview

After promoting the REF segment, changing the maximum usage, mapping constants to the Reference Number Qualifier elements, and establishing the appropriate qualifying relationship, you can map the SHIPTOLOC and SHIPTODEPTNUM fields.

Mapping the SHIPTOLOC and SHIPTODEPT NUM fields

Use this procedure to map the SHIPTOLOC and SHIPTODEPTNUM fields.

Step	Action
1	Link the application field SHIPTOLOC in the INVBEGIN record to the REF7:02 element in the REF:7 segment (N1_ShipTo group).
2	Link the application field SHIPTODEPTNUM in the INVBEGIN record to the REF9:02 element in the REF:9 segment (N1_ShipTo group).

Mapping other application fields and elements

This table lists the application fields in the INVBEGIN record and the elements they are mapped to.

To map application field	You must link
REMITTONAME	N1:502 element in the N1:5 segment (N1_RemitTo group).
REMITTOADD1	N3:501 element in the N3:5 segment (N1_RemitTo group).
REMITTOADD2	N3:502 element in the N3:5 segment (N1_RemitTo group).
REMOTTOCITY	N4:501 element in the N4:5 segment (N1_RemitTo group).
REMITTOSTATE	N4:502 element in the N4:5 segment (N1_RemitTo group).
REMITTOPOSTALCODE	N4:503 element in the N4:5 segment (N1_RemitTo group).

Mapping the Bill of Lading Number

Overview

Introduction   You know that you only send one header REF segment to your trading partner (typically the REF segment repeats up to 12 times). Therefore, after you change the max usage of the REF segment to "1," the BOLNUM (bill of lading number) field can be directly mapped to the REF02 element. However, you must map a constant to the Reference Number Qualifier (REF01 element) and define a qualifying relationship with the Reference Number (REF02 element) to indicate that the REF02 contains the bill of lading number.

How to Change the Max Use of the REF Segment

Overview   You have determined that you only send your partner one header REF segment (reference information), so you must change the maximum usage allowed for that segment from "12" to "1."

Procedure   Use this procedure to make the REF a non-repeating segment.

Step	Action
1	Right-click the REF segment (in the header) to access the shortcut menu.
2	From the shortcut menu, select **Properties** to access the EDI Segment Properties dialog box.
3	Select the **Looping** tab.
4	In the Max Usage box, type "1" to make the REF a non-repeating segment. This box is currently set to "12."
5	Click **OK** to accept the change to the REF segment.

How to Set the Reference Number Qualifier

Overview   You already defined all the constants that you are using in this map, so now you can map the appropriate constant to the Reference Number Qualifier element.

Procedure   Use this procedure to set the Reference Number Qualifier for the REF segment.

Step	Action
1	Double-click the REF:01 element (Reference Number Qualifier) in the REF segment to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.
4	From the Constants list, select the "BM" constant, to identify this iteration of the REF segment as containing the bill of lading number.
5	From the Qualifies list, select the "0127 REFERENCE NUMBER," to set up a qualifying relationship with the Reference Number element.
6	Click **OK** and the "BM" code is loaded in the Reference Number Qualifier element and the qualifying relationship between the REF01 and REF02 elements is established.

How to Map the BOLNUM Field

Overview   After mapping the constant to the Reference Number Qualifier element and establishing
the appropriate qualifying relationship, you can map the BOLNUM field.

Procedure   Link the application field BOLNUM in the INVBEGIN record to the REF02 element in
the REF segment.

Mapping the Ship Date

Overview

Introduction   You know that you only send one header DTM segment to your trading partner (typically the DTM segment repeats up to 10 times). Therefore, after you change the max usage of the DTM segment to "1," the SHIPDATE field can be directly mapped to the DTM02 element (in the DTM header segment). However, you must map a constant to the Date/Time Qualifier (DTM01 element) and define a qualifying relationship with the Date (DTM02 element) to indicate that the DTM02 contains the ship date.

How to Change the Max Use of the DTM Segment

Overview   You have determined that you only send your partner one header DTM segment (date/time information), so you must change the maximum usage allowed for that segment from "10" to "1."

Procedure   Use this procedure to make the DTM a non-repeating segment.

Step	Action
1	Right-click the DTM segment (in the header) to access the shortcut menu.
2	From the shortcut menu, select **Properties** to access the EDI Segment Properties dialog box.
3	Select the **Looping** tab.
4	In the Max Usage box, type "1" to make the DTM a non-repeating segment. This box is currently set to "10."
5	Click **OK** to accept the change to the DTM segment.

How to Set the Ship Date Qualifier

Overview   You already defined all the constants that you are using in this map, so now you can map the appropriate constant to the Date/Time Qualifier element.

Procedure   Use this procedure to set the Date/Time Qualifier for the DTM segment.

Step	Action
1	Double-click the DTM:01 element (Date/Time Qualifier) in the DTM segment to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.
4	From the Constants list, select the "DTM" constant, to identify this iteration of the DTM segment as containing the bill of lading number.
5	From the Qualifies list, select the "0373:6 DATE," to set up a qualifying relationship with the Date element.
6	Click **OK** and the "DTM" code is loaded in the Date/Time Qualifier element and the qualifying relationship between the DTM01 and DTM02 elements is established.

How to Map the SHIPDATE Field

Overview   After mapping the constant to the Date/Time Qualifier element and establishing the appropriate qualifying relationship, you can map the SHIPDATE field.

Procedure   Link the application field SHIPDATE in the INVBEGIN record to the DTM02 element in the DTM segment.

Mapping the Item Detail Information

Overview

Introduction   You know that you only send one header ITD segment to your trading partner (typically the ITD segment repeats up to 5 times). Therefore, after you change the max usage of the ITD segment to "1," the TERMS_WHOLE field can be directly mapped to the ITD03 element (in the ITD header segment).

How to Change the Max Use of the ITD Segment

Overview   You have determined that you only send your partner one header ITD segment (terms of sale/deferred terms of sale), so you must change the maximum usage allowed for that segment from "5" to "1."

Procedure   Use this procedure to make the ITD segment non-repeating.

Step	Action
1	Right-click the ITD segment (in the header) to access the shortcut menu.
2	From the shortcut menu, select **Properties** to access the EDI Segment Properties dialog box.
3	Select the **Looping** tab.
4	In the Max Usage field, type "1" to make the ITD a non-repeating segment. This field is currently set to "5."
5	Click **OK** to accept the change to the ITD segment.

How to Map the Terms Fields

Mapping the TERMS_WHOLE field

Link the application field TERMS_WHOLE in the INVBEGIN record to the ITD03 element in the ITD segment (in the header).

Mapping the TERMSNETDUE field

Link the application field TERMSNETDUE in the INVBEGIN record to the ITD07 element in the ITD segment.

Mapping the TERMSDISDUE field

Link the application field TERMSDISDUE in the INVBEGIN record to the ITD05 element in the ITD segment.

The INVComment Group: INVCOMMENT1 Record

Overview

Introduction   The INVCommentGroup (comment information) groups all the comment information for the invoice, including the reference number and the message text. The INVCommentGroup occurs once per invoice translated. The information within the INVCommentGroup is further segregated by record.

This section describes the actions necessary for you to map from each field in the INVCOMMENT1 (first comment) record. The INVCOMMENT1 record contains the reference number. The first comment record is mandatory and occurs once per invoice translated (once for each comment line).

These fields are part of the application file format key, but do not contain data that the system needs to use to generate the invoice. Therefore, you do not perform any mapping operations on these fields:

◗ BILLTOACCTCM1,

◗ DOCTYPCM1, and

◗ INVNUMCM1.

How to Map the REFNUM Field

Overview The REFNUM field can be directly mapped to the N902 element (in the N9 segment in the N9 group). However, you must map a constant to the Reference Number Qualifier (N901 element) and define a qualifying relationship with the Reference Number (N902 element) to indicate that the N902 contains the reference number.

Setting the reference number qualifier You already defined all the constants that you are using in this map, so now you can map the appropriate constant to the Reference Number Qualifier element.

Procedure Use this procedure to set the Reference Number Qualifier for the N9 segment.

Step	Action
1	Double-click the N901 element (Reference Number Qualifier) in the N9 segment (N9 group), to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.
4	From the Constants list, select the "1V" constant, to identify this iteration of the REF segment as containing the reference number (related vendor order number). This number is in addition to the primary order number.
5	From the Qualifies list, select the "0127 REFERENCE NUMBER," to set up a qualifying relationship with the Reference Number element.
6	Click **OK** and the "1V" code is loaded in the Reference Number Qualifier element and the qualifying relationship between the N901 and N902 elements is established.

Mapping the reference number After mapping the constant to the Reference Number Qualifier element and establishing the appropriate qualifying relationship, you can map the REFNUM field.

Link the application field REFNUM in the INVCOMMENT1 record to the N902 element in the REF segment.

The INVCommentGroup: INVCOMMENT2 Record

Overview

Introduction   This section describes the actions necessary for you to map from each field in the INVCOMMENT2 (second comment) record. The INVCOMMENT2 record contains message text. The second comment record is mandatory and occurs up to ten times per invoice translated (once for each comment line).

These fields are part of the application file format key, but do not contain data that the system needs to generate the invoice. Therefore, you do not perform any mapping operations on these fields:

◗   BILLTOACCTCM2,

◗   DOCTYPCM2, and

◗   INVNUMCM2.

How to Map the MESSAGETEXT Field

Procedure   Link the application field MESSAGETEXT in the INVCOMMENT2 record to the
MSG01 element in the MSG segment (N9 group).

The INVCommentGroup: INVDETAIL Record

Overview

Introduction   This section describes the actions necessary for you to map from each field in the INVDETAIL (detail) record. The INVCOMMENT2 record contains line item number and description, quantity shipped, quantity ordered, unit of measure, price, product code, and invoice terms. The detail record is mandatory and occurs up to 200,000 times per invoice translated (once for each line item invoiced).
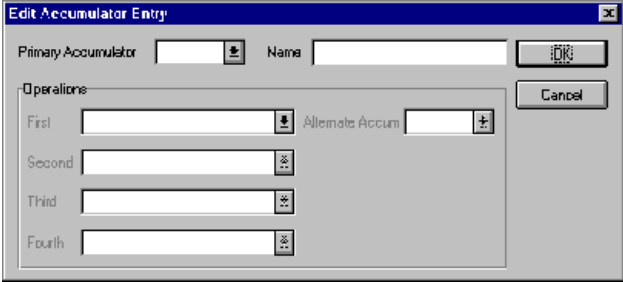
These fields are part of the application file format key, but do not contain data that the system needs to generate the invoice. Therefore, you do not perform any mapping operations on these fields:
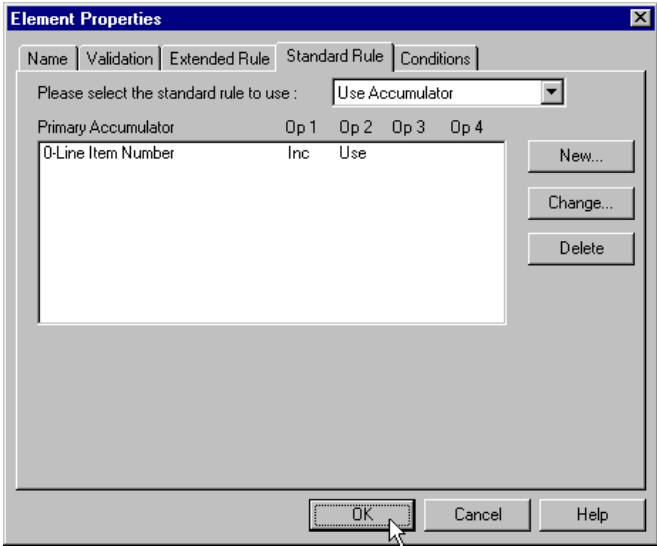
◗   BILLTOACCDET

◗   DOCTYPDET

◗   INVNUMDET

How to Map the Item Number

Overview   The ITEMNUM field can be directly mapped to the IT101 Assigned Identification element in the IT1 segment (IT1 group). However, you also need to calculate an incremental count of the number of line items, and you want to use that total value in the Number of Line Items Total field.

Procedure   Use this procedure to count line items and generate a control total.

Step	Action
1	Double-click the IT101 element (in the IT1 segment in the IT1 group). This is the field that you typically use to count the line items. The Element Properties dialog box is displayed.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Accumulator**.
4	Click **New** to access the Edit Accumulator Entry dialog box to create a new calculation for this element.
5	From the Primary Accumulator list, select "0." Before any calculations are performed on an accumulator, its content is "0" (zero). When you use an accumulator, the system adds a new accumulator to the bottom of this list. **Note** There is *only one* set of accumulators for each map. This means that the accumulator "0," whether it is used in the Primary Accumulator or Alternate Accum box 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 is the result of the earlier calculation. Any additional calculations you assign to that accumulator are performed on the contents resulting from an earlier calculation. (Continued on next page)

(Contd) Step	Action
6	In the Name box, type "Line Item Number." This is a descriptive alias that enables you to differentiate what the accumulators you create are used for.
7	From the First list, select "Increment primary." This is the first operation that the system performs. This specifies that the system increments the value in the accumulator by one for each iteration of the IT1 group. **Note** The First box is active only after you select a Primary Accumulator.
8	Click **OK** to add the accumulator. The Standard Rule tab of the Element Properties dialog box should now look like this:
9	Click **OK** on the Element Properties dialog box to add the standard rule to the IT101 element.
10	Double-click the CTT01 element (in the CTT segment). This is the field that typically contains the total number of line items. The Element Properties dialog box is displayed.
11	Select the **Standard Rule** tab.
12	From the standard rule list, select **Use Accumulator**.
13	Click **New** to access the Edit Accumulator Entry dialog box to create a new calculation for this element.
14	From the Primary Accumulator list, select primary accumulator "0." This accumulator currently contains the total number of line items.

(Contd) Step	Action
15	From the First list, select "Use primary." This specifies that the system loads the current value of the accumulator into the CTT01 (Number of Line Items Total) element. The Edit Accumulator Entry dialog box should now look like this:
16	Click **OK** to add the accumulator.
17	Click **OK** on the field Properties dialog box to add the standard rule to the CTT01 element. The CTT01 element now contains the total number of line items in the invoice.
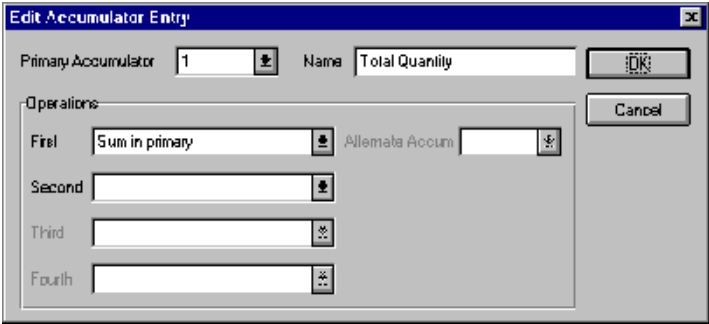
How to Map the ITEMNUM Field
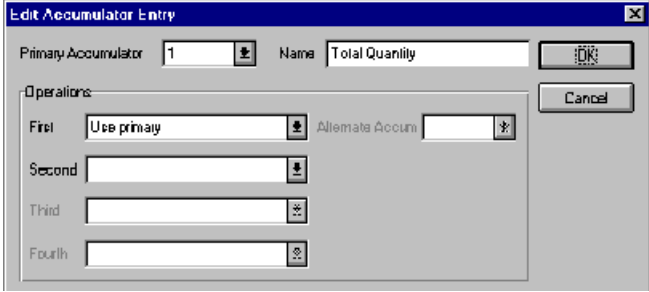
Overview   After using an accumulator to calculate the incremental count of the number of line items, you can map the ITEMNUM field directly to the IT101 element.

Procedure   Link the application field ITEMNUM in the INVDETAIL record to the IT101 element in the IT1 segment (IT1 group).
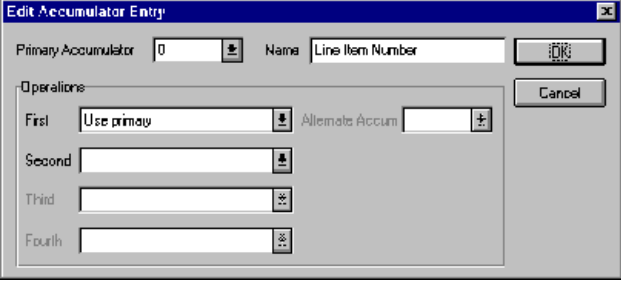
How to Map the Quantity Shipped

Overview   The QTYSHIP field can be directly mapped to the IT102 Quantity Invoiced element in the IT1 segment (IT1 group). However, you also need to count the quantity invoiced for each line item and load the total quantity in the CTT02 (Hash Total) field.

Procedure   Use this procedure to count the quantity invoiced and generate a hash total.

Step	Action
1	Double-click the IT102 element (in the IT1 segment in the IT1 group). This is the field that you typically use to count the quantity invoiced. The Element Properties dialog box is displayed.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Accumulator**.
4	Click **New** to access the Edit Accumulator Entry dialog box to create a new calculation for this element.
5	From the Primary Accumulator list, select "1." Before any calculations are performed on an accumulator, its content is "0" (zero). When you use an accumulator, the system adds a new accumulator to the bottom of this list.
6	In the Name box type "Total Quantity." This is a descriptive alias that enables you to differentiate what the accumulators you create are used for.
7	From the First list, select "Sum in primary." This is the first operation that the system performs. This specifies that the system adds the numeric value (takes the positive or negative sign of the numbers into account) of the field to the contents of the Primary Accumulator for each iteration of the PO1 group. The Edit Accumulator Entry dialog box should now look like this: *(Continued on next page)*

(Contd) Step	Action
8	Click **OK** to add the accumulator.
9	Click **OK** on the Element Properties dialog box to add the standard rule to the IT102 element.
10	Double-click the CTT02 element (in the CTT segment). This is the field that typically contains the total quantity of the purchase order. The Element Properties dialog box is displayed.
11	Select the **Standard Rule** tab.
12	From the standard rule list, select **Use Accumulator**.
13	Click **New** to access the Edit Accumulator Entry dialog box to create a new calculation for this element.
14	From the Primary Accumulator list, select primary accumulator "1." This accumulator currently contains the total quantity.
15	From the First list, select "Use primary." This specifies that the system loads the current value of the accumulator into the CTT02 (Hash Total) element. The Edit Accumulator Entry dialog box should now look like this:
16	Click **OK** to add the accumulator.
17	Click **OK** on the Field Properties dialog box to add the standard rule to the CTT02 element. The CTT02 element now contains the total quantity of the purchase order.

How to Map the QTYSHIP Field

Overview   After mapping the using an accumulator to calculate the quantity invoiced for each line item and generate a hash total, you can map the QTYSHIP field directly to the IT102 element.

Procedure   Link the application field QTYSHIP in the INVDETAIL record to the IT102 element in the IT1 segment (IT1 group).

How to Map the Quantity Ordered

Overview	You know that you only send one QTY segment (in the IT1 group) to your trading partner (typically the QTY segment repeats up to 5 times). Therefore, after you change the max usage of the QTY segment to "1," the QTYORD (quantity ordered) field can be directly mapped to the QTY02 element. However, you must map a constant to the Quantity Qualifier (QTY01 element) and define a qualifying relationship with the Quantity (QTY02 element) to indicate that the QTY02 contains the quantity ordered.

Changing the max use of the QTY segment	You have determined that you only send your partner one QTY segment (quantity), so you must change the maximum usage allowed for that segment from "5" to "1."

Making the QTY segment non-repeating

Use this procedure to make the QTY a non-repeating segment.

Step	Action
1	Right-click the QTY segment (in the IT1 group) to access the shortcut menu.
2	From the shortcut menu, select **Properties** to access the EDI Segment Properties dialog box.
3	Select the **Looping** tab.
4	In the Max Usage box, type "1" to make the QTY a non-repeating segment. This box is currently set to "5."
5	Click **OK** to accept the change to the QTY segment.

Setting the quantity qualifier	You already defined all the constants that you are using in this map, so now you can map the appropriate constant to the Quantity Qualifier element.

Procedure

Use this procedure to set the Quantity Qualifier for the REF segment.

Step	Action
1	Double-click the QTY01 element (Quantity Qualifier) in the QTY segment (IT1 group), to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.

(Contd) Step	Action
4	From the Constants list, select the "OR" constant, to identify the Quantity segment as containing the quantity ordered.
5	From the Qualifies list, select the "0380 QUANTITY," to set up a qualifying relationship with the Quantity element.
6	Click **OK** and the "OR" code is loaded in the Quantity Qualifier element and the qualifying relationship between the QTY01 and QTY02 elements is established.

How to Map the QTYORD Field

Overview   After mapping the constant to the Quantity Qualifier element and establishing the appropriate qualifying relationship, you can map the QTYORD field.

Procedure   Link the application field QTYORD in the INVDETAIL record to the QTY02 element in the QTY segment (IT1 group).

How to Map the Unit Price
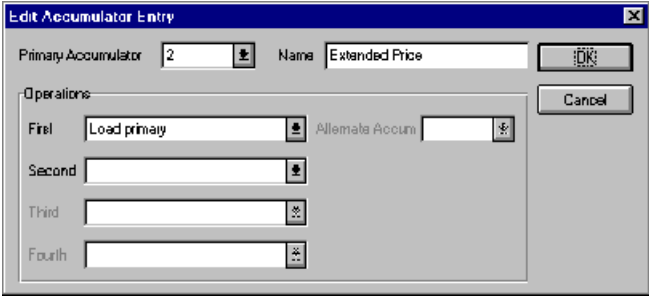
Overview
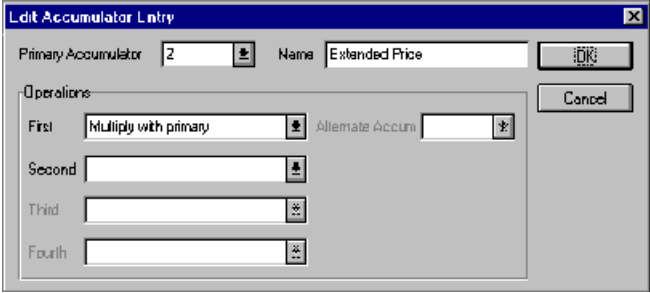
The UNITPRICE field can be directly mapped to the IT104 Unit Price element in the IT1 segment (IT1 group). However, you also need to multiply the quantity invoiced for each line item by the unit price to obtain the extended price. You also need to generate a running total of the extended price and load the final total in the TDS01 (Total Invoice Amount) element.

Obtaining the extended price

Use this procedure to multiply the quantity invoiced for each line item by the unit price to obtain the extended price.

Step	Action
1	Double-click the IT102 element (in the IT1 segment in the IT1 group). This is the field that you typically use to count the quantity invoiced. The Element Properties dialog box is displayed.
2	Select the **Standard Rule** tab. **Note** The Standard Rule list already shows that **Use Accumulator** is selected.
3	Click **New** to access the Edit Accumulator Entry dialog box to create a new calculation for this element.
4	From the Primary Accumulator list, select "2." This number is a name by which the accumulator is referred. Before any calculations are performed on an accumulator, its content is "0" (zero). When you use an accumulator, the system adds a new accumulator to the bottom of this list.
5	In the Name box, type "Extended Price." This is a descriptive alias that enables you to differentiate what the accumulators you create are used for. *(Continued on next page)*

(Contd) Step	Action
6	From the First list, select "Load primary." This is the first operation that the system performs. This specifies that the system loads the contents of the element group into the Primary Accumulator for each iteration of the IT1 group The Edit Accumulator Entry dialog box should now look like this:
7	Click **OK** to add the accumulator.
8	Click **OK** on the Element Properties dialog box to add the standard rule to the IT102 element.
9	Double-click the IT104 element (in the IT1 segment in the IT1 group). This is the field that contains the unit price for each line item. The Element Properties dialog box is displayed.
10	Select the **Standard Rule** tab.
11	From the standard rule list, select **Use Accumulator**.
12	Click **New** to access the Edit Accumulator Entry dialog box to create a new calculation for this element.
13	From the list, select primary accumulator "2." (Continued on next page)

(Contd) Step	Action
14	From the First list, select "Multiply with primary." This is the first operation that the system performs. This specifies that the system multiplies the value of the IT104 (Unit Price) element with the contents of the primary accumulator, and stores the result in the primary accumulator for each iteration of the IT1 group. The Edit Accumulator Entry dialog box should now look like this:
15	Click **OK** to add the accumulator.
16	Click **OK** on the Element Properties dialog box to add the standard rule to the IT104 element. **Tip** If there is an extended price element in your EDI file, you could load the total from the extended price calculation into that element. To do this, you need to use an accumulator on that extended price element that specifies "Use primary" for accumulator 2.

Generating a running total

Use this procedure to generate a running total of the extended price.

Step	Action
1	Double-click the IT104 element (in the IT1 segment in the IT1 group). This is the field that contains the unit price for each line item. The Element Properties dialog box is displayed.
2	Select the **Standard Rule** tab. **Note** The Standard Rule list already shows that **Use Accumulator** is selected.
3	Click **New** to access the Edit Accumulator Entry dialog box to create another calculation for this element.
4	From the Primary Accumulator list, select primary accumulator "3."

(Contd) Step	Action
5	In the Name box, type "Running Total."
6	From the First list, select "Add primary to alternate." This specifies that the system adds the contents of the primary accumulator to the contents of the alternate accumulator, and stores the result in the primary accumulator for each iteration of the IT1 group.
7	From the Alternate Accum list, select "2." This operation specifies that the system add the value of accumulator "2" (which contains the extended price for a line item) to the value of accumulator "3." The system stores the sum in accumulator "3," which therefore contains a running total of the extended price with each iteration of the IT1 group. The Edit Accumulator Entry dialog box should now look like this:
8	Click **OK** to add the accumulator.
9	Click **OK** on the Element Properties dialog box to add the standard rule to the IT104 element.
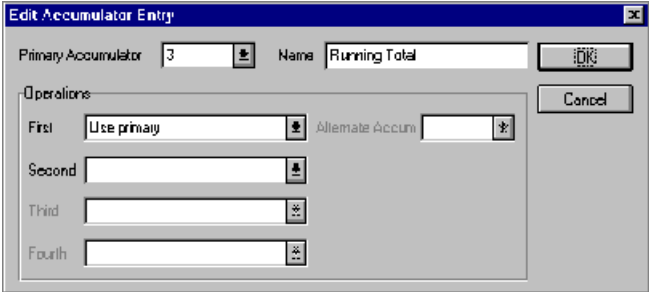
Loading the running total

Use this procedure to load the running total of the extended price into the TDS01 (Total Invoice Amount) element.

Step	Action
1	Double-click the TDS01 element (in the TDS segment). This is the field that contains the unit price for each line item. The Element Properties dialog box is displayed.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Accumulator**.
4	Click **New** to access the Edit Accumulator Entry dialog box to create a new calculation for this element.
5	From the list, select primary accumulator "3."

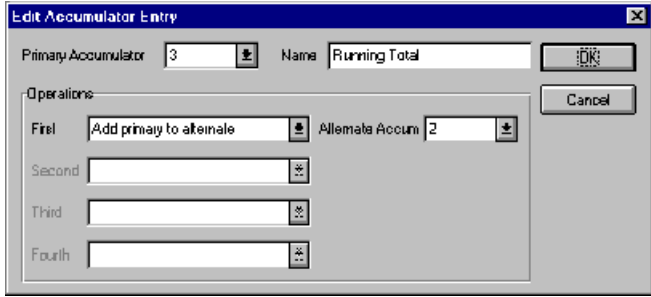| (Contd) Step	Action
6	From the First list, select "Use primary." This specifies that the system loads the contents of the primary accumulator into the TDS01 (Total Invoice Amount) element. The Edit Accumulator Entry dialog box should now look like this:
7	Click **OK** to add the accumulator.
8	Click **OK** on the Element Properties dialog box to add the standard rule to the TDS01 element.

How to Map the UNITPRICE Field

Overview   After using accumulators to multiply the quantity invoiced for each line item by the unit price to obtain the extended price and generate a running total of the extended price, you can map the UNITPRICE field directly to the IT104 element.

Procedure   Link the application field UNITPRICE in the INVDETAIL record to the IT104 element in the IT1 segment (IT1 group).

How to Map the UOM Field

Procedure   Link the application field UOM in the INVDETAIL record to the IT103 element in the IT1
segment (IT1 group).

Mapping the Customer Product Code and Your UPC Code

Overview

Introduction   You can map the CUSTPROCODE (customer product code) and UPCCODE (your UPC code) fields directly to two Product/Service ID elements IT1 segment. However, you must map a constant to each Product/Service ID Qualifier and establish a qualifying relationship with the Product Service ID. This enables you to specify which Product/Service ID element contains your partner's product code, and which contains your UPC code.

How to Set the Product/Service ID Qualifier

Overview   You must map a constant to the Product/Service ID Qualifier (IT106 and IT108 elements) and define a qualifying relationship with the Product/Service ID (IT107 and IT109 elements).

You already defined all the constants that you are using in this map, so now you can map the appropriate constant to each Product/Service ID Qualifier element.

Procedure   Use this procedure to set the Product/Service ID Qualifiers for the IT1 segment.

Step	Action
1	Double-click the IT106 element (Product/Service ID Qualifier) in the IT1 segment in the IT1 group, to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.
4	From the Constants list, select the "BP" constant (buyer's part number), to identify the IT107 element as containing the customer product code.
5	From the Qualifies list, select the "0234 PRODUCT/SERVICE ID," to set up a qualifying relationship with the Product/Service ID element.
6	Click **OK** and the "BP" code is loaded in the Product/Service ID Qualifier element and the qualifying relationship between the IT106 and IT107 elements is established.
7	Double-click the IT108 element (Product/Service ID Qualifier) in the IT1 segment in the IT1 group, to access the Element Properties dialog box.
8	Select the **Standard Rule** tab.
9	From the standard rule list, select **Use Constant**.
10	From the Constants list, select the "UI" constant (UPC identification), to identify the IT109 element as containing the UPC code.
11	From the Qualifies list, select the "0234:2 PRODUCT/SERVICE ID," to set up a qualifying relationship with the Product/Service ID element.
12	Click **OK** and the "UI" code is loaded in the Product/Service ID Qualifier element and the qualifying relationship between the IT108 and IT109 elements is established.

How to Map the CUSTPROCODE and UPCCODE Fields

Overview   After establishing the appropriate qualifying relationships, you can map the CUSTPROCODE and UPCCODE fields.

Procedure   Use this procedure to map the product code information.

Step	Action
1	Link the application field CUSTPROCODE in the INVDETAIL record to the IT107 element in the IT1 segment (IT1 group).
2	Link the application field UPCCODE in the INVDETAIL record to the IT109 element in the IT1 segment (IT1 group).

How to Map the Item Description

Overview

You know that you only send one PID:2 segment (in the PID subgroup in the IT1 group) to your trading partner (typically the PID subgroup repeats up to 1,000 times). Therefore, after you change the max usage of the PID subgroup to "1," the ITEMDESC (item description) field can be directly mapped to the PID:205 element. However, you must map a constant to the Item Description Type (PID:201 element) and define a qualifying relationship with the Description (PID:205 element) to indicate that the PID2:05 contains a free-form item description.

How to Change the Max Use of the PID Subgroup

Overview   You have determined that you only send your partner one PID:2 segment (in the PID subgroup in the IT1 group), so you must change the maximum usage allowed for that subgroup from "1,000" to "1."

Procedure   Use this procedure to change the maximum usage of the PID subgroup.

Step	Action
1	Right-click the PID subgroup (in the IT1 group) to access the shortcut menu.
2	From the shortcut menu, select **Properties** to access the Group Properties dialog box.
3	Select the **Looping** tab.
4	In the Maximum Usage box, type "1" to make the PID subgroup a non-repeating group. This box is currently set to "1,000."
5	Click **OK** to accept the change to the PID subgroup.

How to Set the Item Description Type

Overview   You must map a constant to the Item Description Type (PID:201 element) and define a qualifying relationship with the Description (PID:205 element).

You already defined all the constants that you are using in this map, so now you can map the appropriate constant to Item Description Type element.

Procedure   Use this procedure to set the Item Description Type (in the PID:2 segment in the PID subgroup in the IT1 group).

Step	Action
1	Double-click the PID:201 element (Item Description Type) in the PID:2 segment in the PID subgroup in the IT1 group, to access the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Constant**.
4	From the Constants list, select the "F" constant (free-form), to identify the PID:201 element as containing the free-form item description.
5	From the Qualifies list, select the "0352 DESCRIPTION," to set up a qualifying relationship with the Description element.
6	Click **OK** and the "F" code is loaded in the Item Description Type element and the qualifying relationship between the PID:201 and PID:205 elements is established.

How to Map the ITEMDESC Field

Procedure   Link the application field ITEMDESC in the INVDETAIL record to the PID:205 element in the PID:2 segment (PID subgroup, IT1 group).

Note
If the system prompts you with a confirmation message, click **OK**.

How to Map the Terms Type Code

Overview   You know that you only send one ITD:2 segment (in the IT1 group) to your trading partner (typically the ITD:2 segment repeats up to 2 times). Therefore, after you change the max usage of the ITD:2 segment to "1," the TERMSTYPE (terms type code) field can be directly mapped to the ITD:201 element.

How to Change the Max Use of the ITD:2 Segment

Overview   You have determined that you only send your partner one ITD:2 segment (quantity), so you must change the maximum usage allowed for that segment from "2" to "1."

Procedure   Use this procedure to make the ITD:2 a non-repeating segment.

Step	Action
1	Right-click the ITD:2 segment (in the IT1 group) to access the shortcut menu.
2	From the shortcut menu, select **Properties** to access the EDI Segment Properties dialog box.
3	Select the **Looping** tab.
4	In the Max Usage field, type "1" to make the ITD:2 a non-repeating segment. This field is currently set to "2."
5	Click **OK** to accept the change to the ITD:2 segment.

How to Map the TERMS Fields

Map the TERMSTYPE field

Link the application field TERMSTYPE in the INVDETAIL record to the ITD:201 element in the ITD:2 segment (IT1 group).

Map the TERMSDISP field

Link the application field TERMSDISP in the INVDETAIL record to the ITD:203 element in the ITD:2 segment (IT1 group).

INVSUMMARY Record

Mapping Totals

Overview   This section describes the actions necessary for you to map from each field in the INVSUMMARY (summary) record. The INVSUMMARY record contains the total invoice amount and the total transaction amount. The summary record is mandatory and occurs once per invoice translated.

These fields are part of the application file format key, but do not contain data that the system needs to use to generate the invoice. Therefore, you do not perform any mapping operations on these fields.

- ◗ BILLTOACCTSUM
- ◗ DOCTYPSUM
- ◗ INVNUMSUM

How to Map the TOTALAMOUNT Field

Overview   For the system to create output segments, there must be at least one explicit link to an element in that segment, and data to support that link. Therefore, even though you are mapping the total amount with accumulators, you must add an explicit link with an application field that contains data.

Procedure   Link the application field TOTALAMOUNT in the INVSUMMARY record to the TDS01 element in the TDS segment.

How to Map the TRANSTOTAL Field

Overview For the system to create output segments, there must be at least one explicit link to an element in that segment, and data to support that link. Therefore, even though you are mapping the transaction total with accumulators, you must add an explicit link with an application field that contains data.

Procedure Link the application field TRANSTOTAL in the INVSUMMARY record to the CTT01 element in the CTT segment.

Finalizing the Map

Overview

Introduction
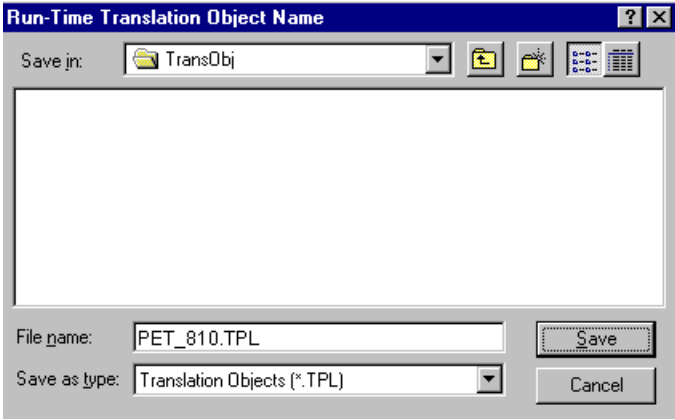
After you create the import map, define your application file format, and perform the appropriate mapping operations, you need to finalize the map. To complete the mapping process, you need to save the map, compile the translation object, print and review the Gentran:Server for Windows report, and test the map.
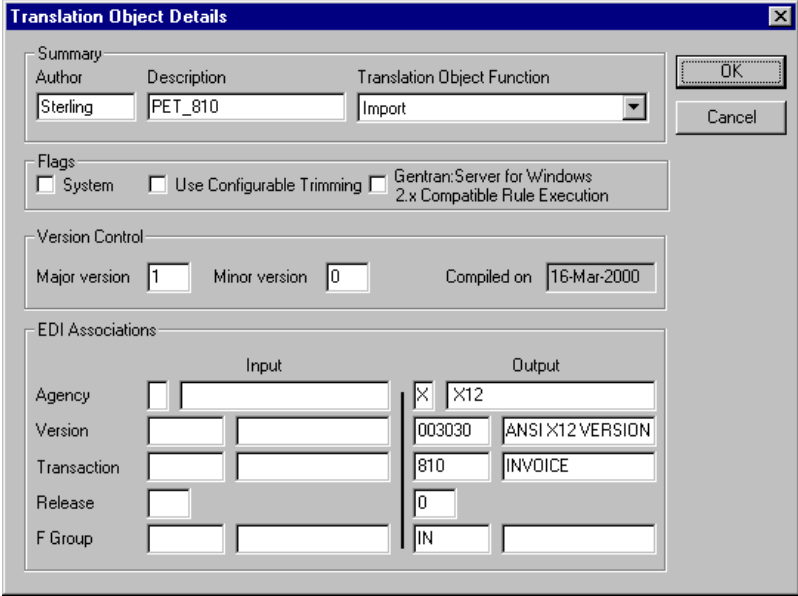
How to Compile the Map

Overview   The Compile function compiles the map and generates a translation object. The map that you created using the Application Integration subsystem (PET_810.MAP) is a *source map*. When that source map is compiled, the result is a *compiled translation object*.

After you save the translation object, you must register it with Gentran:Server before you can use it.

Procedure   Use this procedure to compile a map and generate a translation object.
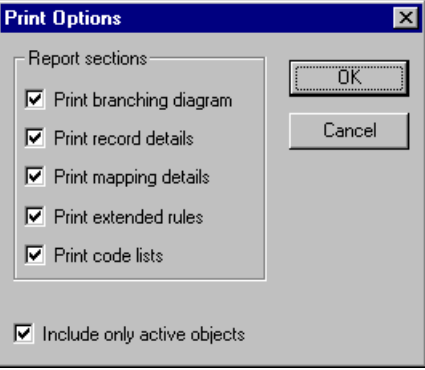
Step	Action
1	From the File menu, select **Save** to save the source map, prior to using the Compile function.
2	From the File menu, select **Compile** to display the Run-Time Translation Object Name dialog box.
3	The File name box already contains "PET_810.TPL." This is the name of the translation object, 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. **Caution** Do not overlay the source map with the compiled translation object. Use the .TPL file extension to distinguish the translation object. *(Continued on next page)*

(Contd) Step	Action
4	Change the Drive and Folder where the compiled translation object is stored, if necessary. **Caution** 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 Gentran:Server for Windows.
5	Click **Save** and the system compiles the map and generates a translation object. The Compile Error dialog box is displayed. Verify that no errors occurred and click **OK** to exit the dialog box. The date on which the translation object was compiled is automatically loaded into the Compiled on box on the Transaction Object Details dialog box.
6	From the File menu, select Save to save the source map with the Compiled on date. **Note** You must register this translation object with the Gentran:Server for Windows system before you can use it. **Reference** See the *Gentran:Server for Windows User Guide* for more information on registering a translation object.

How to Print the Report

Overview   The Gentran:Server for Windows 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 reiterate the mapping process as many times as necessary (make the modifications, save the map, recompile the translation object, and print the report again).

Procedure   Use this procedure to print the Gentran:Server report.

Step	Action
1	From the File menu or Main Toolbar, select Print to access the Print Options dialog box.
2	Accept the defaults on this dialog box (all options checked), so the report includes all sections and only the activated map components. Click **OK** and the Print dialog box is displayed.
3	Set the appropriate options. Click **OK** if you do *not* need to change Setup options, and the mapping report is printed. Click **Setup** if you need to access printer setup information (to select a specific printer, paper orientation, or size and source of paper). The Print Setup dialog box is displayed.
4	Set the appropriate options. Click **OK** if you do *not* need to change Print options (dithering, intensity control, or print true type as graphics), and you return to the Print dialog box. Click **OK** on the Print dialog box and the mapping report is printed. Click **Options** if you need to access printer setup options information. The Options dialog box is displayed.
5	Set the appropriate options. Click **OK** to return to the Print Setup dialog box. Click **OK** on the Print Setup dialog box to return to the Print dialog box. Click **OK** on the Print dialog box and the report is printed.

How to Test the Translation Object

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

Reference
See the *Gentran:Server for Windows User Guide* for more information about these topics:

◗ Registering a translation object
◗ Importing a partner
◗ Selection outbound translation objects
◗ Importing files
◗ Viewing raw EDI data

Procedure   Use this procedure to test the system import (HEADER.TPL) and import (PET_810.TPL) translation objects.

Step	Action
1	Register the HEADER.TPL and PET_810.TPL translation objects with Gentran:Server for Windows.
2	If you have not already done so, import the PETZONE.PAR partner relationship into Gentran:Server for Windows. The default full path for the partner file is: `C:\GENSRVNT\TUTORIAL\PETZONE.PAR`
3	Verify (in Partner Editor) that the PET X 3030 810 Import translation object is selected for the Outbound Relationship.
4	Ask your system administrator to add the system import translation object (HEADER.TPL) to the System Configuration program. **Reference** Please see the *Administration Guide* for more information about how your system administrator must modify the system configuration program.
5	Use the Import option in Gentran:Server for Windows to process the data file (PET_810.TXT) through the translation object. The data file is located in the TUTORIAL subfolder under the folder where Gentran:Server for Windows is installed. The default full path for the data file is: `c:\GENSRVNT\TUTORIAL\PET_810.TXT`
6	After the document is translated, it is located in the Workspace in Gentran:Server for Windows. View the EDI data to ensure that the document was translated correctly.

Glossary

?In Documents	This Gentran:Server browser contains a list of documents that were received by the system but failed compliance checking or that do not have an identifiable partner or transaction set.
?Out Documents	This Gentran:Server browser contains a list of documents that were imported into the system via unattended processing but are invalid.
acknowledgement	This term is used to indicate the ANSI 997 functional acknowledgement, the EDIA 999 acceptance/rejection advice, and the EDIFACT CNTRL document.
activation	This function enables you to make map components available for use. The system activates all the groups, segments, composites, and elements that are defined as "mandatory" (must be present) by the standard. The system does not enable you to deactivate the mandatory groups, segments, composites, and elements. 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.
AIAG	The Automotive Industry Action Group (AIAG) is the standards-setting group for the automotive industry. The standards form a subset of the ANSI X12 standard.
ANA	Article Numbering Association.
ANSI	American National Standards Institute. ANSI sets standards for many products and services, such as safety glasses and battery capacities. The ANSI X12 committee is the chief EDI standards-setting organization for the United States.
application file	If you are creating an import or export map, you must define your application to Gentran:Server. In 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 either need to extract from your partner's document (if the map is inbound) or need to send to your partner (if the map is outbound), so that your system can accurately process the data.

application system	Computer systems, outside of EDI, designed to fulfill specific business functions. These include accounting, purchasing, materials control, human resources, shipping, and other systems.
browser	A browser is a window that displays a list of items, such as documents (transaction sets), interchanges, translation objects, etc. A document browser displays only documents.
button	A button enables you to execute a command. You select a button either by clicking it or by pressing **TAB** until the button you want is selected and then pressing **ENTER**. Unavailable buttons are dimmed, and the currently selected button has a darker border than the other buttons.
code list tables	These are used by EDI standards as repositories for lists of codes. Each EDI standard provides a code list for each element that can be further defined with a code.
colours	This feature enables you to select foreground and background colours to visually define the various map components. The use of colour is optional.
communications session	Everything sent and received to/from one telephone number in one continuous period of connection. This could include sending two or three interchange envelopes to a network, each for a different trading partner.
Communicator	This is the Gentran:Server communications software. The Communicator enables you to send, receive, resend, establish communication port definitions, establish communication profiles, establish communication scripts, and view and delete communication sessions.
compile	This function compiles the map and creates a translation object. The map that you create using Gentran:Server is a *source map*. When that source map is compiled, the result is a *compiled translation object*. This translation object must be registered with the Gentran:Server system before you can use it.
compliant	This means that the document conforms to the EDI standards as defined by the translation object.
component data element	A simple data element that belongs to a composite data element. Component data elements are also called subelements.

composite data element	A 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).
conditions	See relational conditions.
constant	This standard rule enables you to move a literal constant value to the specified element or field, indicate a qualifying relationship with another element or field, and map the current date or time to the specified element or field.
cross-reference table	This table is created in Partner Editor and is used to convert your values to your trading partner's values during outbound processing, or your partner's values to your values during inbound processing.
date/time	This type of field or element contains a date or time. If you specify that a field or element is a date/time type, you must specify exactly how the date or time must be formatted.
deactivation	This function enables you to make map components unavailable for use by the system. The system does not enable you to deactivate the mandatory groups, segments, composites, and elements.
default	A predefined value – programs use these built-in values *unless* you specifically override them.
dialog box	A dialog box (or dialog) contains additional information or options that you need to act on. If you specify any options, you can select a button to execute the command. Some dialog boxes display warnings or messages indicating why a requested task cannot be accomplished. If the dialog box has a title bar, you can move it to another location on your desktop.
document	One transaction set containing actual data and treated as a single entity. The amount of data does not affect whether something is a document or not, only the fact that it is treated as a single entity. For example, regardless of whether a purchase order contains one item or ten thousand, if it is one purchase order, it is one document.
	If a trading partner sends you a communication containing 10 purchase orders, you have received 10 documents. If the communication contained 15 invoices, you have received 15 documents.

double-click	This indicates that you should position your cursor on top of the designated item and quickly press and release your primary mouse button twice. The primary mouse button is the button you access with your index finger, regardless of whether you are utilizing right-handed or left-handed mouse operation.
EANA	International Article Numbering Association, Brussels, Belgium.
EDI	Electronic Data Interchange (EDI) is the process by which companies can exchange business documents directly from application to application by computer without paper documents being produced.
EDI standard	This is the rules for turning a business document into an EDI document.
EDIA	The Electronic Data Interchange Association, formerly known as the Transportation Data Coordinating Committee (TDCC).
EDIFACT	This is the standards organization of the United Nations Economic and Social Council. The acronym is short for the Electronic Data Interchange for Administration, Commerce, and Transport.
element	The smallest piece of usable information defined by the standards. Examples might include a quantity, unit price, or description. An individual element can have somewhat different meanings depending on context. Therefore, elements are normally not considered to have useful meaning until they are combined into segments. 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.

There are three types of data elements, as illustrated in the table below:

Data Element	Definition
Simple data element	A single piece of information defined by the standards.
Composite data element	A data element that is made up of multiple component elements.
Component data element	A simple data element that belongs to a composite data element. Component data elements are also called subelements.

envelope	A way of separating information in transmissions for ease of processing. Each envelope contains a header segment and a trailer segment, which separate the envelope from other envelopes and provide information about the contents of the envelope.

There are three levels of envelopes, as follows:

Transaction Set
Each transaction set (business document) is contained within a transaction set envelope.

Functional Group
An envelope containing related business documents. The standards define which transaction sets should be placed together into a functional group envelope.

Interchange Envelope
All material being sent to one trading partner in one communication. The term Interchange Envelope is the term used by ANSI. EDIA uses the term Transmission Envelope to refer to this level of envelope. Since we use the term transmission for other uses, we will refer to Interchange Envelope only.

Note
A communications session could easily include a number of interchange envelopes. For this reason, the standards-setting bodies are considering a fourth level of envelope to cover an entire communications session. |
equalize	This function enables you to reinstate the two sides of the map with focus in equal dimensions. The use of the Equalize function is optional.
export map	This map defines how to move data from the EDI standard-formatted documents that your partners send you to your application file (flat file definition). An export map is necessary for inbound processing.
export translation object	See export map.
extended rules	These rules enable you to use a Gentran:Server proprietary programming language to perform virtually any mapping operation you require.
field	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 to move data to and from your application file.
fixed-format file	See application file.

font
This feature enables you to globally change the font that is used in the display of all maps. The default font that Gentran:Server uses is a Sans Serif 9 point. The Font feature enables you to change the font type, style, and point size. This gives you the flexibility to shrink the font if you need to view more of the map on your monitor, enlarge the font, or change the type and style to be more easily readable by you.

functional group
A group of transaction sets that the standards-setting body (such as ANSI) has defined as fitting together with other related transaction sets. An example might be a Functional Group defined as containing all purchasing transaction sets.

group
A looping structure that contains related records/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. 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 and the EDI file are both groups and therefore, they is visually represented the same way as other groups and subgroups in Gentran:Server.

import
This command enables you to import data from an external application file. Depending on the content of the file, the system may prompt you for partner, transaction, or translation object information.

import map
This 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. An import map is necessary for outbound processing.

import translation object
See import map.

In Documents
This Gentran:Server browser contains a list of documents that the system received but that have not yet been processed by the user. Once the documents are processed (printed or exported) the documents are transferred to the In Drawer.

inbound mapping
The system translates your trading partner's EDI standard formatted business document to your application file format, so you can receive documents from you partners. To translate inbound data, you need to create an export map.

In Drawer
This Gentran:Server browser contains a list of documents that were received and processed.

integer	An integer is a number that has an implied decimal point (e.g., "2.01" is formatted as "201").
interchange	The interchange contains all the functional groups of documents (transaction sets) sent from one sender to one receiver in the same transmission.
Interchanges browser	This Gentran:Server browser enables you to view a hierarchical list of all the interchanges that were sent or received. It contains status information about the interchanges, (e.g., whether or not a functional acknowledgement was received and the status of that acknowledgement). This browser also enables you to access the documents within the interchanges. The Interchanges browser can always be accessed.
link	This function 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 (hereafter referred to as fields) is visually represented with a line connecting the two fields.
literal constant	These are used by the system as a repository to store information that is used at a later point in the map. Typically, constants are used in an outbound map to generate a qualifier.
location table	This is a table created in Partner Editor that is used to contain address-related information about the partner.
lookup table	This is a table created in Partner Editor that is used to select information related to a value in inbound or outbound data.
loop count	This standard rule enables you to count the number of times a loop is repeated, if the element or field is part of a loop. If the loop is a nested loop, you can track the current loop *or* the outer loop. For example, if the Y loop is nested within the X loop, and the Y loop has cycled through 15 iterations and the X loop has cycled through 3 iterations, you can choose to count either the "15" (Y loop) or the "3" (X loop).
loop start/loop end	Certain EDI standards use Loop Start (LS) and Loop End (LE) segments. LS and LE segments differentiate between two or more loops of the same type. If the transaction contains LS and LE segments and depending on whether your map is inbound or outbound, you must define the LS and LE segments for the loops you are using in the map in one of two different ways.

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

menu
A list of items from which you can choose. Each item represents a command. Unavailable items on a menu are dimmed to inform you they are disabled.

menu bar
This contains the list of menus for the application.

network
Also known as a Third Party Network or Value-Added Network (VAN) that accepts and holds transmissions from companies until it is convenient for a trading partner to accept them.

number
This type of field or element contains either an integer or real number. If you specify that a field or element is a number type, you must designate the format by specifying a format of either "N" (integer) or "R" (real) and the number of decimal places.

ODETTE
The Organization for Data Exchange by Tele-Transmission in Europe.

Out Documents
This Gentran:Server browser contains a list of documents that are ready to be sent. After the documents are successfully sent, they are automatically transferred to the Out Drawer.

outbound mapping
The system translates your application 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 Gentran:Server.

Out Drawer
This Gentran:Server browser contains a list of documents that were successfully sent out by the system.

partner
Another firm with which your company trades documents. Also referred to as a trading partner.

Partner Editor
This Gentran:Server function enables you to define, edit, and delete all partner information for your company and all of your trading partners.

positional file
See application file.

post
This Gentran:Server function enables you to move compliant documents from the Workspace to the Out Documents.

predefined
On a data entry translation object, a default value for a particular element. You can change the value as necessary.

printout
This is produced when data received from a trading partner passes through a print translation object. You do not have to actually print the data. The printout could be to a file on your hard disk.

promote
This function extracts one iteration (instance) of a group or repeating segment. This enables you to map unique data from your application file, and/or enter a specialized definition. Gentran:Server specifies that only one-to-one (no loops) or many-to-many (loop) mapping relationships are valid.

qualifier
This is an element that has a value expressed as a code that gives a specific meaning to the function of another element. A qualifying relationship is the interaction between an element and its qualifier. The function of the element changes depending on which code the qualifier contains.

real number
A real number has an explicit decimal point (e.g., "2.01" is formatted as "2.01"), and truncates trailing zeroes.

receive
This Gentran:Server function enables you to manually initiate a communications session to receive data from your trading partner.

record
Contains a group of related fields. A record can occur once or can repeat multiple times.

relational condition
This function enables to connect fields together for syntax or compliance reasons. For example, Field A is invalid unless Field B is present. Therefore, if you set up a condition that pairs Fields A and B, the system generates a compliance error if one of those fields is not present.

respond
This Gentran:Server function enables you to create a turnaround document (using the Turn Around translation object in the partner relationship) in response to one or more selected documents in the In Drawer.

responsible agency	An organization that develops and updates standards for EDI communications. These organizations include ANSI, EDIA, AIAG, UCS, EDIFACT, ODETTE, and VICS.
segment	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. For example, a catalog price segment might consist of elements for item description, volume, price, lead time, etc. By themselves, none of these elements would communicate useful information. Together, they provide the information necessary for someone to tell if the item is what is desired, whether the price is reasonable, etc. A number of segments together form a transaction set.
select	This standard rule enables you to select entries from a location table, cross-reference table, partner table, or lookup table (all tables created in the Gentran:Server Partner Editor). You can then map the fields in those tables to one or more fields in the data. The Select function uses the value of the current field to perform the selection.
send	This Gentran:Server function enables you to manually start a communications session to send data to your trading partner. Only selected documents are enveloped and sent. If no documents are selected, ALL documents are sent. Successfully sent documents are moved to the Out Drawer.
set	See Transaction Set.
simple mapping	See link.
split	This function enables you to split (break) a group or repeating segment into two loops. Typically, you use this function when you need more than one instance of the same map component that still occurs multiple times.
standard rules	These rules give you access to functions that are necessary for mapping operations that are more complex than simple linking, but less involved than extended rules. Each of the standard rules are mutually exclusive (you can use only one on a particular field).
status bar	The status bar of an application window defines information about a selection, command, or process, defines Menu Bar items as the user highlights each item, and indicates any current keyboard-initiated modes for typing (e.g., CAP for the "Caps Lock" key or NUM for the "Num Lock" key, etc.).

string   This type of field or element contains one or more printable characters. If you specify that a field or element is a string type, you must designate the format by specifying a syntax token.

syntax token   This function enables you to designate a "token" that defines ranges of characters and/or numbers that are allowed to be used for a string-type element or field. You can then use the syntax tokens in the Format field of the Field Properties dialog. This enables you to define what type of characters should be used while compliance checking each element/field (i.e., alphanumeric within a certain range, numeric within a certain range, etc.).

system import map   The translator uses this 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. You need a system import map that is defined in EDIMGR.INI to translate outbound data.

system import translation object   See system import map.

system translation object   These translation objects control the creation and separation of interchanges, functional groups, and transaction sets. They also used to generate and reconcile functional acknowledgements. All the required system translation objects are automatically installed with the Gentran:Server system.

TDCC   See EDIA.

TDF   TDF is the Transaction Data File. This file serves as a filter between your document files and the Gentran:Server translator. For outbound processing, data is imported from a TDF file and translated to EDI format using a TDF Import translation object. The data is then ready to be posted and sent to your trading partner. For inbound processing, EDI documents that are received by Gentran:Server can be exported to a TDF-formatted file using a TDF Export translation object. This data file is then ready to be processed or converted to your internal application files.

third-party network   See Network.

title bar   The title bar of an application window contains the name of the application and enables you to move the window to another position on your desktop.

toolbar	This contains buttons that graphically represent commands available in the Menu Bar.
TRADACOMS	The U.K. standards for EDI that are published by the Article Numbering Association (UK) LTD.
trading partner	Another firm with which you company trades documents. Also referred to as a partner.
transaction set (document)	A business form as defined by the standards. Examples include an ANSI 850 purchase order or an UCS 880 invoice. The standards define each transaction set in terms of the segments and elements that make up the form, the order in which they appear, and the relationships among them. This is also known as "message" in Europe.

translation object   A predesigned layout set up to ensure that input or output for a particular transaction set exists and is presented in a usable fashion. You must specify which translation objects are used by each partner relationship.

Inbound Translation Objects:
    Turn Around: This translation object is used when a document is received, to create the natural response document that contains as many elements from the received document as possible.

    Export File: This translation object indicates that when a document is received, it is exported to a specified file format.

    Print: This translation object is used to print documents.

Outbound Translation Objects:
    Import: This translation object is used to import data from an application file.

    Print: This translation object is used to print documents.

    Data Entry: This translation object is used to enter data into the Document Editor facility.

translator	This is the engine that processes data for Gentran:Server, the Application Integration subsystem, and the Forms Integration subsystem.
transmission	See Communications Session.
transmission chain	A path an EDI communication could follow, including one company, one trading partner, and one or more network services.

turnaround document	A document into which data elements from the source document have been automatically transferred using a turnaround map.
turnaround map	A series of instructions that the system uses to create a turnaround document (a logical response document to the source) from an inbound (source) document, by transferring data from the source document to elements in the target document (translation object).
UCS	The Uniform Communications Standard is the standard used by the grocery industry.
update	This standard rule enables you to update a specific field in a document record, envelope segment, interchange, group, current partner, or document (if the map side format type is EDI), with the contents of the element or field.

Caution
Please note that this function updates the internal Gentran:Server database tables. We recommend that you use this function only if you are sure you really want to update the internal database tables.

Typically, you *only* use this function if you want to update the document name and reference in the document table. Any other use of this function could have disastrous consequences!

use accum	This standard rule gives you access to a set of numeric variables that you can manipulate via numeric operations, and then transfer to and from fields. This function enables you to add, change, or delete calculations for the element/field, including hash totals (used to accumulate numeric field values, i.e., quantity, price, etc.). This function also enables you to map the accumulated total into a control total field, and use accumulators. Accumulators are used generally for counting the occurrences of a specific element or generating increasing or sequential record or line item numbers.
use code	This standard rule enables you to match an element or field against a predefined code table, specify whether or not a compliance error is generated if the element or field does not contain one of the values in the code table, and store a code's description in another element or field.
user translation object	These translation objects control data entry, importing, exporting, document turnaround, and creating printed reports. These translation objects are created using the Application Integration or Forms Integration subsystems. The Application Integration subsystem enables you to generate import, export, and document turnaround translation objects. The Forms Integration subsystem enables you to generate data entry and print translation objects.

version   Each standards-setting body updates its standards on a regular basis. Each formal update is referred to as a version.

VICS   The Voluntary Inter-industry Communication Standards is the standards-setting body for the retail industry, a subset of ANSI X12.

Workspace   This Gentran:Server browser contains a list of outbound "Work in Progress" documents. It also contains recently imported or data entry documents.

X12   The ANSI committee that sets and publishes standards for EDI.