Gentran:Server® for Windows®

Application Integration
EDIFACT Tutorial Guide

Version 5.2

Sterling Commerce
An IBM Company

Copyright Notice

Gentran:Server for Windows
© Copyright 1995–2008
Sterling Commerce, Inc.
ALL RIGHTS RESERVED

Table of Contents

Glossary

CHAPTER

1

Getting Started

Contents

Overview

Introduction

Application integration subsystem

Welcome to the Gentran:Server® for Windows® Application Integration subsystem, Sterling Commerce's electronic commerce 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 Gentran:Server for Windows Application Integration subsystem using the EDIFACT standard. This tutorial includes an inbound mapping example (purchase order) and an outbound mapping example (invoice). Its purpose is to break the mapping process down into concise, understandable steps, using examples that will enable you to master the tasks 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 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.

This guide also assumes that you are familiar with EDI concepts, EDI standards structure, your internal application format, data mapping concepts, and Gentran:Server for Windows.

Contents

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

Chapter Number	Chapter Title	Description
1	Getting Started	Explains the content, organization, conventions and system requirements in this guide. This chapter briefly describes how to build maps using Gentran:Server for Windows Application Integration and how to customize global display options. It also introduces mapping and Gentran:Server for Windows.
2	Inbound Purchase Order	Explains the tasks and procedures necessary to create an export map for an EDIFACT inbound (ORDERS) purchase order.
		(Continued on next page)

(Contd) Chapter Number	Chapter Title	Description
3	Outbound Invoice	Explains the tasks and procedures necessary to create a system import map and an import map for an EDIFACT outbound (INVOIC) invoice.
	Glossary	Contains mapping definitions and Gentran:Server for Windows Application Integration subsystem terms and concepts.

Introducing Mapping

Mapping Overview

Introduction

A map is a set of instructions that you define in the Gentran:Server for Windows Application Integration subsystem to indicate the relationship between your application file and the EDI standards. A map also defines how the system translates 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 must create an export map in the Gentran:Server for Windows Application Integration subsystem 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, must to create an import map and a system import map. You must 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) to the EDI standard-formatted documents that your partner expects to receive from you. When an application file contains multiple documents, you must create a system import map to specify which trading relationship (established in Partner Editor) corresponds to which document in the application file so the system knows the import map to use to process the document. See the *Application Integration User's Guide* 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 instructs the translator how to convert a file from one format to another. Each function performed during translation requires a translation object. Translation objects control all Gentran:Server for Windows EDI translation functions. Translation objects can belong to the following two categories:

◗   system translation objects

◗   user translation objects

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

Translation objects (contd)

Gentran:Server for Windows Application Integration allows 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. In addition, you must establish correct trading relationships for all trading partners. To establish a trading relationship, you must specify the appropriate translation object name 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.
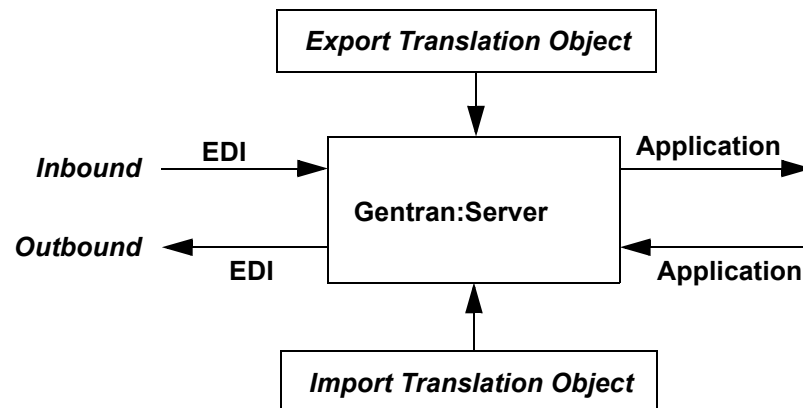
When you create an import map, you must define to Gentran:Server for Windows your application file format and the EDI standard format in which your partners expect to receive the documents. When you create an export map, you must define to Gentran:Server for Windows the EDI standard format in which your partners will send 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 the 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 field information that you either extract from your partner's document (if the map is inbound) or that you need to send to your partner (if the map is outbound).

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

Application 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, 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. 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 sub-groups. This means that you must create records and groups before you create the subordinate fields.

EDI file format

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

The Gentran:Server for Windows Application Integration subsystem generates an EDI file, 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 defined by the standards agency for the document version you selected. Please see the *Application Integration User's Guide* 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 customize the system-generated EDI file by modifying the map component properties, and using specialized Gentran:Server for Windows functions to manipulate the EDI file structure.

The specific EDI map components that you use depend on the type of map you are creating. This includes the standard, version, and transaction set (document) selected, and the groups, segments, composites, and elements your company requires. We recommend that you determine which map components you will use 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.

EDI 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.

EDI map components (contd)

Notes

◗   When an element has a mapping operation performed against it, a red checkmark is displayed over the field 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 map component properties and on 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 allows you to translate your application files to EDI standard formats for documents you send to your partners (outbound mapping) and 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 menu depends 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.

Control Menu   *Title Bar*   *Minimize/Maximize/Close Buttons*

Main Menu Bar   *Status Bar*   *Main Toolbar*
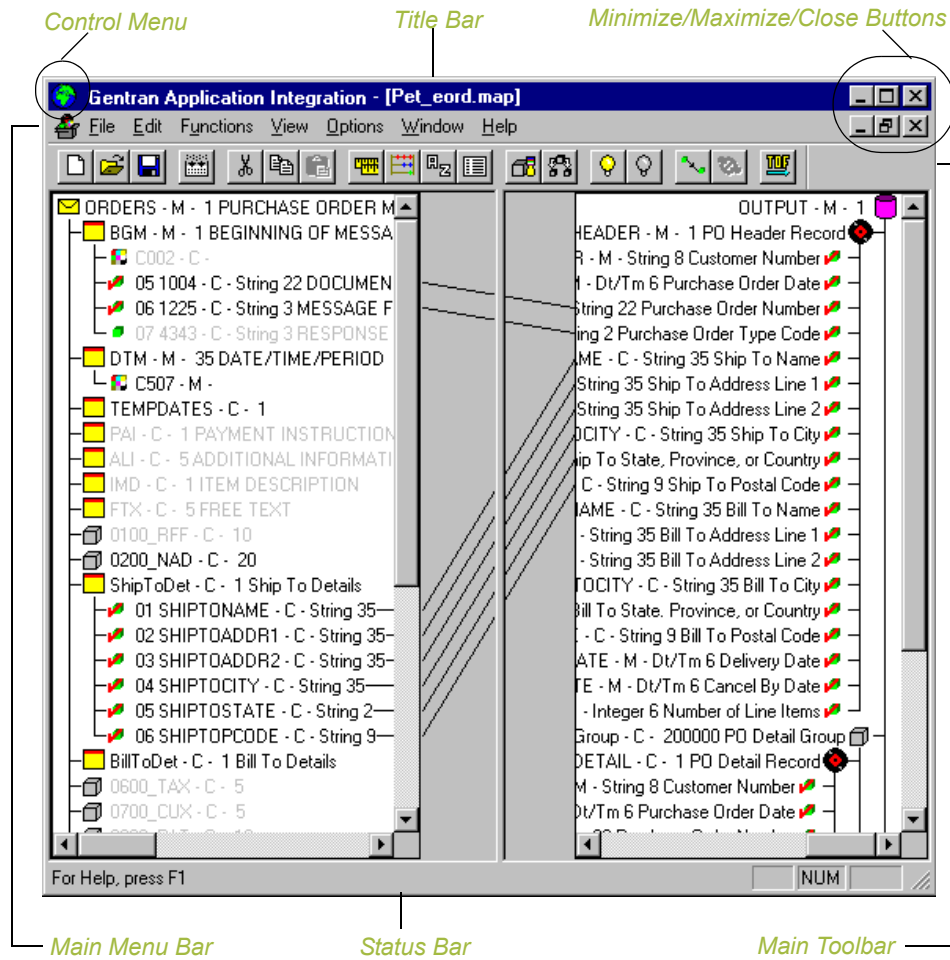
Application integration window components

This table describes the Application Integration Window components.

Part	Function
Control Menu	Contains the following commands: ◗ Restore ◗ Move ◗ Size ◗ Minimize ◗ Maximize ◗ Close
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   Gentran:Server for Windows Application Integration allows you to customize the way you display maps. The global display options that are established apply to all maps. This section explains how to perform the following:

◗ Customize global display options

◗ Display links (global option)

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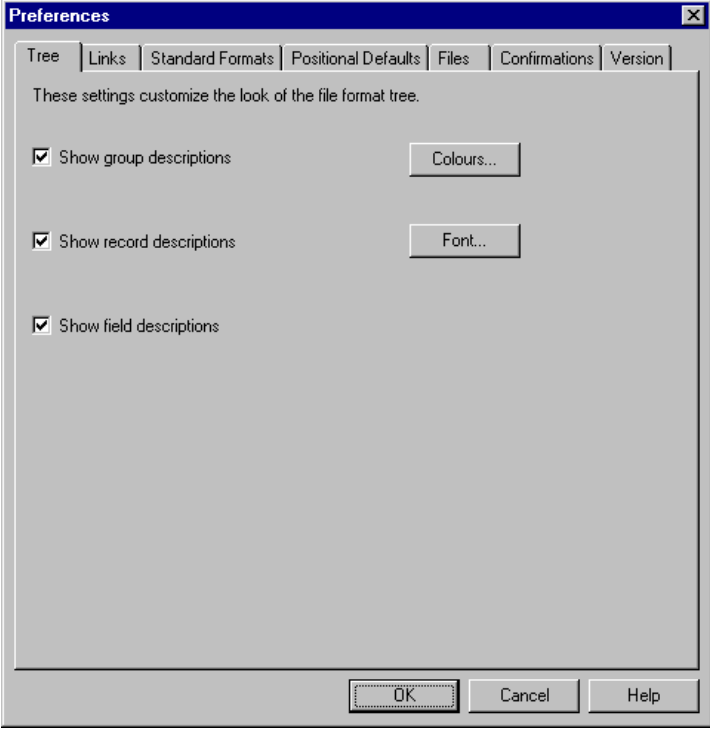
Note
You can set or change global map display options at any time.

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.
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.
2	Select the Links tab to display the linking options. The Links tab enables you to set the default display of mapping links. *(Continued on next page)*

(Contd) Step	Action
3	Select one of the following settings. ▶ Show no links: do not display mapping links. ▶ Show links to or from the currently selected element: display only the mapping links for the currently selected field (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
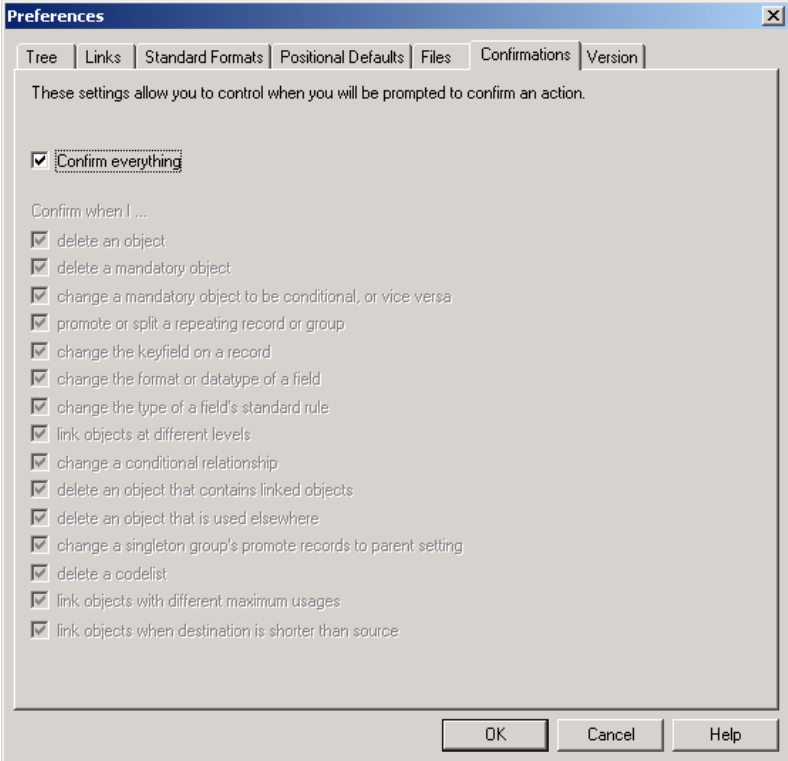
The Confirmations tab on the Preferences dialog box allows you specify when you want confirmation messages displayed.

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). *(Continued on next page)*

(Contd) Step	Action
2	Select the **Confirmation** tab. **System response** The system displays the confirmation options.
3	Set the global confirmation options by either selecting the "Confirm everything" check box (displays all confirmation messages) or by selecting individual confirmation messages by action performed.
4	Click **OK** to save the confirmation options. **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** See *Creating the Export Map* on page 2 - 15 for more information on creating, saving, and naming a map.
4	**Activate the Appropriate EDI Groups, Segments, and Elements** See *Activating EDI Map Components* on page 2 - 20 for more information on activation.
5	**Define your Application File Format** See *Defining Your Application File Format* on page 2 - 25 for more information on defining your application file format.
6	**Map the Appropriate Data for Each Application Field** See *Mapping Your Data* on page 2 - 50 for more information on mapping your data.
7	**Compile the Translation Object** See *How to Compile the Map* on page 2 - 118 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. See *How to Print the Report* on page 2 - 120 for more information on printing the mapping report.

(Contd) Stage	Description
9	**Register the Translation Object with Gentran:Server for Windows** 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 (PETZONE1) is provided on the tutorial diskettes. You need to import the partner profile (PETZONE1.PAR) into Gentran:Server for Windows, before you begin this tutorial. See the *Gentran:Server for Windows User's Guide* for more information on importing trading relationships.
11	**Test 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\PETTESTE1.INT
Outbound (system import and import)	PET_EINV.TXT	C:\GENSRVNT\TUTORIAL\PET_EINV.TXT

CHAPTER

2

Inbound Purchase Order (ORDERS)

Overview

Introduction

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

This chapter explains 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 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).

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

◗   MWT Manufacturing Co. sells these 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. One the purchase orders are translated into your application file format, they can be processed by 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 will be sending you.

Mapping Analysis

The first step in creating a map is to analyze your mapping requirements. This is the most important step in creating a successful map. If your analysis is complete, you will have all the information you need to create the map in an efficient and logical manner. If you omit this critical step, it is likely that creating the map will be a much longer and more arduous task, and the map may be invalid due to oversights or 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 to analyze your application file format, because this is the component that is probably 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, 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 Application Integration. If your company has an existing application file format, as in this tutorial, obtain the record layout from the appropriate person. If you do not have an existing application file format, you must create one by determining fields you will need to process data correctly, and then logically grouping those fields under records.

Application file layout   This is the application file layout for this tutorial.

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (POHEADER) 1 per message					
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	35
SHIPTOSTATE		Ship To State, Province, or Country	String	0	2
SHIPTOPOSTALCODE		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

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
BILLTOCITY		Bill To City	String	0	35
BILLTOSTATE		Bill To State, Province, or Country	String	0	2
BILLTOPOSTALCODE		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 200,000 per message					
Detail Record (PODETAIL) 1 per message					
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 message					
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 5 per message					
CUSTNUMAC	X	Customer Number	String	1	8

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
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 message					
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

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) to ensure 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 200,000 times per order translated. This equals a maximum order of 200,000 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."
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 five 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 purchase order requirements that you will receive from your partners.

Analyzing the EDI File

Overview   To analyze the EDI file, you must first determine what your trading partner will be sending you. You and your partner must agree on which standard, version, and message you want to use. In this tutorial, we use the EDIFACT standard, version 92.1, message ORDERS (purchase order message). It is important to know what type of information your partner will be sending you within the purchase order, and what type of data will be contained in each element used.

It is very helpful to review the EDI standard(s) that you will be using before analyzing the EDI file. Obtain an EDI standards manual for each standard and version that you will be using. Standards manuals are available from EDI standards agencies.

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

Map components   This table lists the map components that are required for this tutorial, but are not currently active following (the composite column is informational only, so you will know where to locate the appropriate element).

Groups		Segment	Group	Segment	Comp.	Element
		BGM segment (already active)				BGM05 element BGM06 element
		DTM segment (already active)			C507	DTM02 element
NAD group		NAD segment			C080 C059 C059	NAD10 element NAD16 element NAD17 element NAD19 element NAD20 element NAD21 element
LIN group		LIN segment			C212	LIN03 element
		IMD:2 segment			C273	IMD:206 element
		QTY:3 segment			C186	QTY:303 element
		MOA:5 segment			C516	MOA:502 element
	ALC group	ALC:2 segment				ALC:204 element
			MOA group	MOA:8 segment	C516	MOA:802 element
		CNT segment			C270	

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 message created by this map easier in Gentran:Server for Windows. This allows you to differentiate between messages 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 messages translated by this map. The Name column of the document browsers that contain this message contains data from the selected element or field.

Recommendation

We strongly recommend that you select an element or field that occurs only once in the message and has a data type of "String." In this tutorial, the map's document name is the purchase order number.

Next step

After you complete the analysis of the EDI file and your application, the final step in mapping analysis is to correlate 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 message		
CUSTNUMHDR	Customer Number	Extract Application ID from Partner record
PODATEHDR	Purchase Order Date	Link to DTM02 using "4" (PODATE) code
PONUMHDR	Purchase Order Number	Link to BGM05
POTYPCODE	Purchase Order Type Code	Link to BGM06
SHIPTONAME	Ship To Name	Link to NAD10 using "ST" (ShipTo) code
SHIPTOADD1	Ship To Address Line 1	Link to NAD16 using "ST" (ShipTo) code

(Contd) Field Name	Description	Mapping Operations
SHIPTOADD2	Ship To Address Line 2	Link to NAD17 using "ST" (ShipTo) code
SHIPTOCITY	Ship To City	Link to NAD19 using "ST" (ShipTo) code
SHIPTOSTATE	Ship To State, Province, or Country	Link to NAD20 using "ST" (ShipTo) code
SHIPTOPOSTALCODE	Ship To Postal Code	Link to NAD21 using "ST" (ShipTo) code
BILLTONAME	Bill To Name	Link to NAD10 using "BT" (BillTo) code
BILLTOADD1	Bill To Address Line 1	Link to NAD16 using "BT" (BillTo) code
BILLTOADD2	Bill To Address Line 2	Link to NAD17 using "BT" (BillTo) code
BILLTOCITY	Bill To City	Link to NAD19 using "BT" (BillTo) code
BILLTOSTATE	Bill To State, Province, or Country	Link to NAD20 using "BT" (BillTo) code
BILLTOPOSTALCODE	Bill To Postal Code	Link to NAD21 using "BT" (BillTo) code
DELDATE	Delivery Date	Link to DTM02 using "117" (DELDATE) code
CANCELDATE	Cancel By Date	Link to DTM02 using "153" (CANCELDATE) code
NUMLINEITEMS	Number of Line Items	Link to CNT02 using "2" (NUMLINEITEMS) code
PO Detail Group (PODetailGroup) up to 200,000 per message		
Detail Record (PODETAIL) 1 per message		
CUSTNUMDET	Customer Number	Extract Application ID from Partner record
PODATEDET	Purchase Order Date	Link to DTM02 using "4" (PODATE) code
PONUMDET	Purchase Order Number	Link to BGM05
		(Continued on next page)

(Contd) Field Name	Description	Mapping Operations
CUSTPROCODE	Customer Product Code	Link to LIN03
UPCCODE	UPC Code	Extract Partner xref by partner item from subtable "PRODCODE"
UOM	Unit of Measure	Extended rule to link to QTY:303
UNITPR	Unit Price	Link to MOA:502 using "146" (UNITPRICE) code
QTYORD	Quantity Ordered	Link to QTY:302 using "21" (QTYORD) code
Description Record (PODESC) up to 5 per message		
CUSTNUMDESC	Customer Number	Extract Application ID from Partner record
PODATEDESC	Purchase Order Date	Link to DTM02 using "4" (PODATE) code
PONUMDESC	Purchase Order Number	Link to BGM05
DESC	Description	Link to IMD:206
Allowance/Charge Record (POALLCHG) up to 5 per message		
CUSTNUMAC	Customer Number	Extract Application ID from Partner record
PODATEAC	Purchase Order Date	Link to DTM02 using "4" (PODATE) code
PONUMAC	Purchase Order Number	Link to BGM05
ACIND	Allowance or Charge Indicator	Link to ALC:201
ACMETHAND	Allowance or Change Method of Handling Code	Link to ALC:204
ACTOTAMT	Allowance or Charge Total Amount	Link to MOA:802
Summary Record (POSUMMARY) 1 per message		
CUSTNUMSUM	Customer Number	Extract Application ID from Partner record

(Contd) Field Name	Description	Mapping Operations
PODATESUM	Purchase Order Date	Link to DTM02 using "4" (PODATE) code
PONUMSUM	Purchase Order Number	Link to BGM05
NUMLINEITEMSUM	Number of Line Items	Link to CNT02 using "2" (NUMLINEITEMS) code

Cross-Reference tables

You must determine if you need code tables to convert map data. If you need code tables, you must decide whether to create a code table in Gentran:Server for Windows Application Integration that can only be used with this map, or whether to add the table to 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).

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
You do not need to create this cross-reference table – it is included with the PETZONE partner (PETZONE1.PAR) that is provided with the tutorial data when you install Gentran:Server

Next step

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

The Export Map
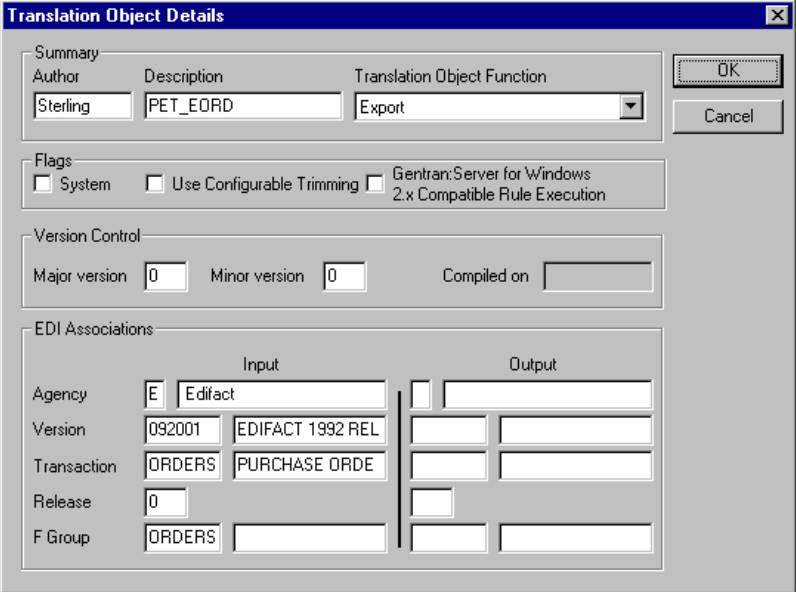
Creating the Export Map

Overview   The first task in creating a new map is to define the format that will be 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 EDIFACT ORDERS purchase order that you receive. Instead of defining the ORDERS message 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.
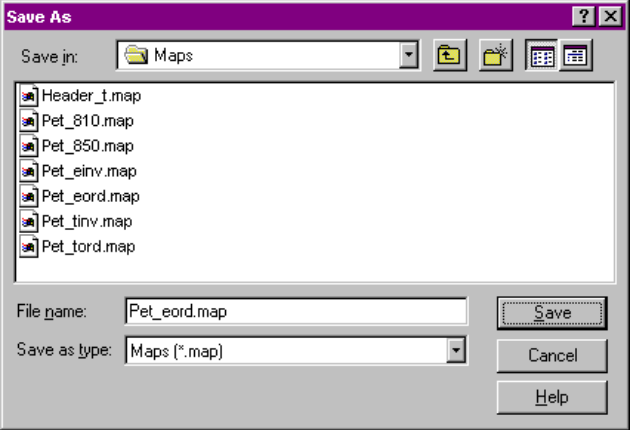
Procedure   Use this procedure to create the export map for the Inbound Purchase Order.

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_EORD** 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 **[E] EDIFACT** standards agency, b. **[092001] EDIFACT 1992 RELEASE 1 (92.1)** version, and c. **[ORDERS] PURCHASE ORDER MESSAGE** 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. (Continued on next page)

(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 E 92.1 ORDERS 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. **Save As** Save in: Maps Header_t.map Pet_810.map Pet_850.map Pet_einv.map Pet_eord.map Pet_tinv.map Pet_tord.map File name: Pet_eord.map Save Save as type: Maps (*.map) Cancel Help
14	Verify that the File name box contains pet_eord.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_T.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_T.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.

(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 have finished creating and saving the new Export map, you must define the Input and Output sides of the map.

Activating 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 (message) selected, and the segments and elements your company requires. We recommend that you determine which segments and elements you will use before starting this step.

When Gentran:Server for Windows generates the EDI side(s) of the map, the system includes all groups, segments, composites, and elements that are defined by the standards agency for the version of the message 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 gray font (dimmed).

Overview (contd)   In this example, the DTM segment and the DTM01 element are mandatory, so they have already been activated by the system. You must also use the DTM02 element (a conditional element), so you will need to activate it.
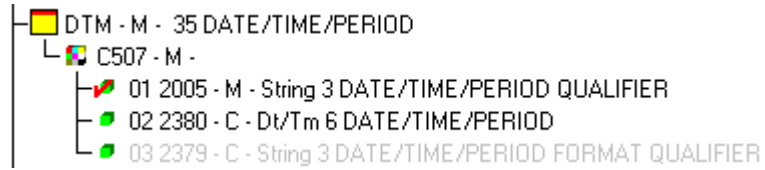
When translating data, the system does not process groups, segments, composites, and elements (or records and fields) that are not activated. Therefore, *you* must activate the groups, segments, composites, and elements that are not defined as mandatory, but that you have determined that you need to use in mapping.

Activation   This table lists the groups, subgroups, segments, and elements you must activate.

Groups		Segment	Group	Segment	Comp.	Element
		BGM segment (already active)				BGM05 element BGM06 element
		DTM segment (already active)			C507	DTM02 element
NAD group		NAD segment			C080 C059	NAD10 element NAD16 element NAD17 element NAD19 element NAD20 element NAD21 element
LIN group		LIN segment			C212	LIN03 element
		IMD:2 segment			C273	IMD:206 element
		QTY:3 segment			C186	QTY:303 element
		MOA:5 segment			C516	MOA:502 element
	ALC group	ALC:2 segment				ALC:204 element
			MOA group	MOA:8 segment	C516	MOA:802 element
		CNT segment				

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

Step	Action
1	Double-click the groups and segments listed in the previous table to open them. Opening all the necessary groups, segments, and composites before you start to activate allows you to scroll down the map and activate the segments and elements sequentially. **Note** When you activate a conditional group or segment, the mandatory subordinate segments and elements are automatically activated.
2	Click the **Activate** button on the Main Toolbar.
3	Click each inactive group listed in the table. This activates the groups. **Note** If you click a group, segment, or element that you did not intend to, right-click it to access the shortcut menu and select Deactivate from the shortcut menu.
4	Click each inactive segment listed in the table. This activates the segments.
5	Click each inactive element listed in the table. This activates the elements.
6	Once you have activated all 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 Verify EDI Delimiters
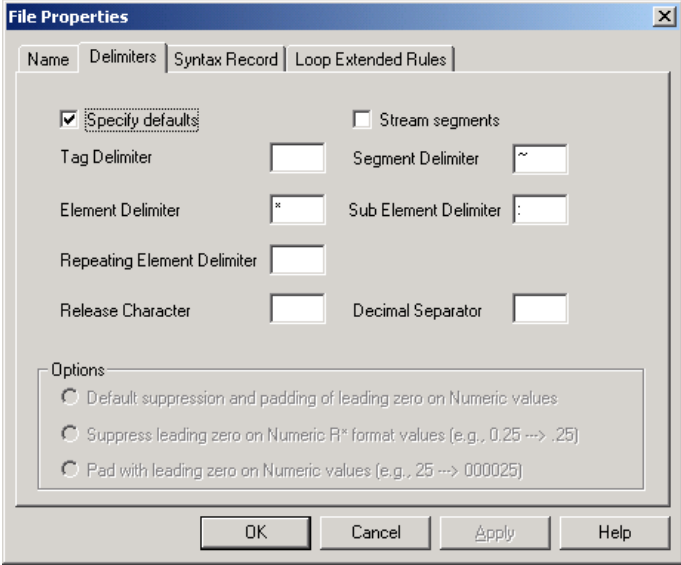
Overview　　If you are using an EDI standard that contains composite elements or sub-elements, you must verify that Gentran:Server for Windows is specifying the correct delimiters. Delimiters are flags that you define to the system as separating specific EDI components. Delimiters are necessary for all variable field-length standards, because data is compressed (and the leading zeroes and trailing blanks are removed). Since 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 sub-elements, we recommend that you perform this task regardless of which standard you use.

Procedure　　Use this procedure to verify EDI delimiters.

Step	Action
1	Right-click the ORDERS ORDER DETAILS EDI file icon and from the shortcut menu, select **Properties**. **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 that the Specify defaults check box is selected.
4	Verify the required delimiters for the EDIFACT 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 EDIFACT are the following (the tag delimiter is not used for EDIFACT):

Box	EDIFACT Default Character	EDIFACT Default Hex Value
Tag Delimiter		
Element Delimiter	+ (plus)	0x2B
Segment Delimiter	' (single quote)	0x27
Sub Element Delimiter	: (colon)	0x3A
Release Character	? (question mark)	0x3F
Decimal Separator	. (period)	0x2E

5	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 message, 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. You must arrange each map component sequentially in the order that it is most logical for the system to process. 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.

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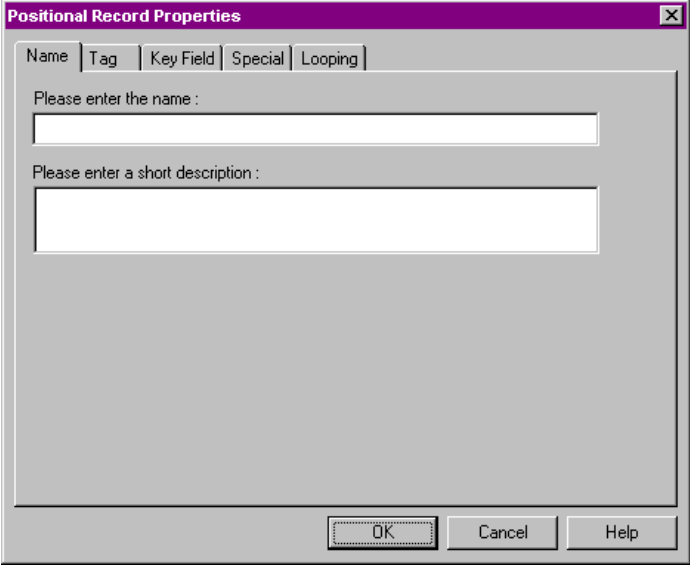
Overview   The first record in your application file is generally the header record. The first record is subordinate to the icon in the application side (output) of the map.

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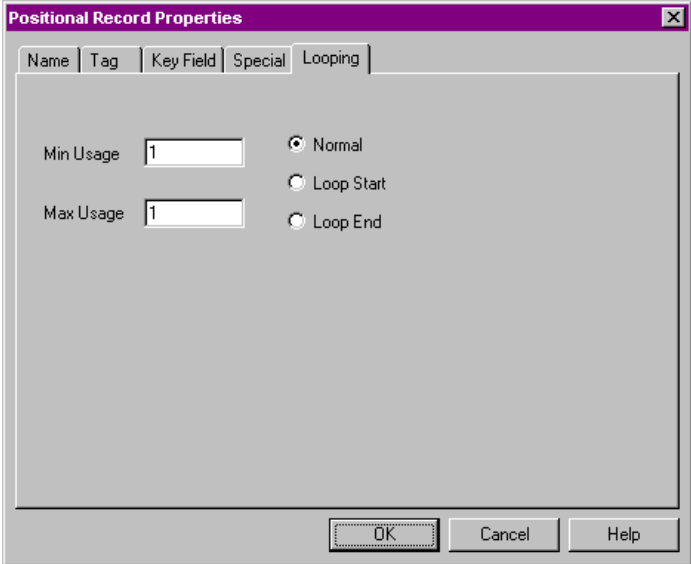
Recommendation
We recommend that you include a header record in each application file. The header record is typically mandatory and repeats only once (it is not a looping structure).

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 submenu. **System response** The system displays the Positional Record Properties dialog box.
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 to access the tag information.
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)
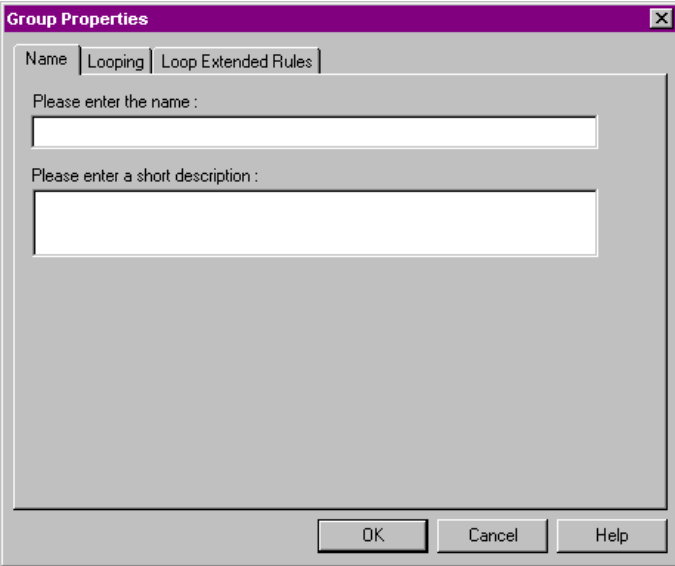
(Contd) Step	Action
8	In the **Min Usage** box, type "1" to make the header record mandatory. This is the minimum number of times the record must repeat. If you want the record to repeat (loop), type the maximum number of times it can repeat in the Max Usage box. **Note** If the Min Usage box contains 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.

How to Create the PODetailGroup

Overview

The PODetailGroup (detail information) contains all line item information for the order, including the item, unit of measure, price, quantity ordered, description, and the allowance and charge information. Each PODetailGroup occurs up to 200,000 times per order translated, allowing for a maximum order of 200,000 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	Click the POHEADER record to highlight it. 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 "PODetailGroup" to indicate that this group contains all of the detail records. This is the segment loop identification code.
3	In the Description box, type "PO Detail Group" as the loop description. This box is used to provide a brief explanation of the loop.
4	Select the **Looping** tab to access looping options. *(Continued on next page)*

(Contd) Step	Action
5	◗ 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. ◗ In the Maximum Usage box, type "200000". This indicates that this group of records can repeat up to 200,000 times. The Group Properties dialog box should now look like this:
6	Click **OK** to create the PODetailGroup.

Creating the PODetailGroup Subordinate Records

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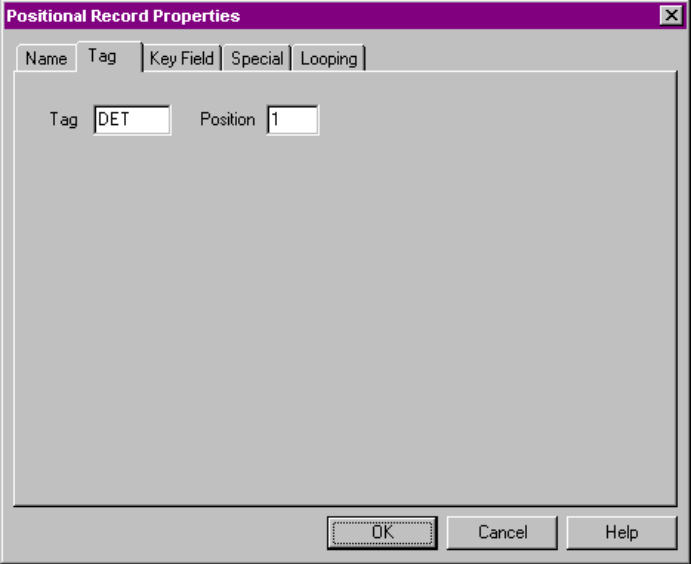
◗   PODETAIL

◗   PODESC

◗   POALLCHG.

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

How to create the   Use this procedure to create the PODETAIL record.
PODETAIL record
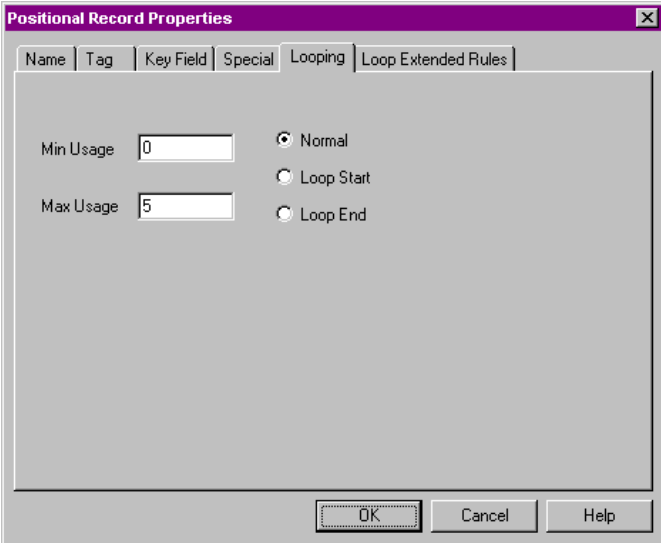
Step	Action
1	Click the PODetail Group to highlight it and select Create Sub from the Edit menu. Select Record from the submenu. **System response** The Positional Record Properties dialog box is displayed.
2	In the Name box, type "PODETAIL" as the record identification code.
3	In the Desc box, type "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 to access tag information.
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 Positional Record Properties dialog box should now look like this: **Positional Record Properties** Name | **Tag** | Key Field | Special | Looping Tag [DET] Position [1] [OK] [Cancel] [Help]
7	Click **OK** to create the PODETAIL record.

How to create the PODESC record

Use this procedure to create the PODESC record.

Step	Action
1	Highlight the PODETAIL record and select Insert from the Edit menu. From the submenu, select Record to display 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 to access tag information.
5	In the Tag box, type "DES" as the record identification code <TAG>.
6	In the Position box, accept the default of "1" as the starting column position of the tag in the data record. *(Continued on next page)*

(Contd) Step	Action
7	Select the **Looping** tab.
	The Looping section of the Positional Record Properties dialog enables you to specify the minimum and maximum number of times this record can repeat. You can also specify whether a record is mandatory or conditional. Alter this section *only* if the record is a looping structure. Otherwise, accept the defaults.
8	Accept the default of "0" in the Min Usage box. This is the minimum number of times the record repeats.
	In the Max Usage box, type "5" as the maximum amount of times the record can repeat.
	The Positional Record Properties dialog should now look like this:
9	Click **OK** to create the PODESC record.

How to create the POALLCHG record

Use this procedure to create the POALLCHG record.

Step	Action
1	Highlight the PODESC record and select Insert from the Edit menu. Select Record from the submenu to display the Positional Record Properties dialog box.
2	In the Name box, type "POALLCHG" the record identification code.

(Contd) Step	Action
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 to access tag information.
5	In the Tag box, type "ACG" as the record identification code <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.
8	In the Max Usage box, type "5" as the maximum amount of times the record can repeat. Accept the default of "0" in the Min Usage box.
9	Click **OK** to create the POALLCHG record.
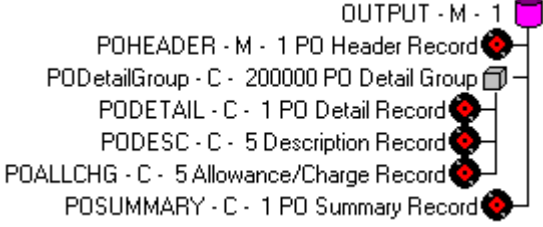
How to Create the POSUMMARY Record

Overview   The final record you must 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 and select Insert from the Edit menu. Select Record from the submenu to display 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.
4	Select the **Tag** tab to access tag information.
5	In the Tag box, type "SUM" as the record identification code <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** tag.
8	Accept the default of "0" in the Min Usage box. Also, accept the default of "1" in the Max Usage box so the summary record only occurs once (not a looping structure).
9	Click **OK** to create the POSUMMARY record. The application side of the map (output) should now look like this: OUTPUT - M - 1 POHEADER - M - 1 PO Header Record PODetailGroup - C - 200000 PO Detail Group PODETAIL - C - 1 PO Detail Record PODESC - C - 5 Description Record POALLCHG - C - 5 Allowance/Charge Record POSUMMARY - C - 1 PO 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 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. You generally create the fields for the first record in the application file (POHEADER), and then proceed with each sequential record.

Application fields   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 (POHEADER) 1 per message					
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	35
SHIPTOSTATE		Ship To State, Province, or Country	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

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
BILLTOCITY		Bill To City	String	0	35
BILLTOSTATE		Bill To State, Province, or Country	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 200,000 per message					
Detail Record (PODETAIL) 1 per message					
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 message					
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

(Continued on next page)

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Allowance/Charge Record (POALLCHG) up to 5 per message					
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 message					
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

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.
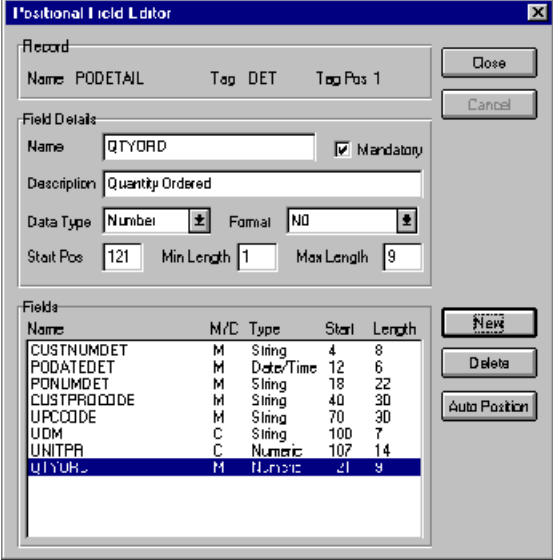
(Contd) Step	Action
7	From the Data Type list, select "String" as the type of field to specify that this field must contain alphanumeric data. **Tip** If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You want 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.
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 - 36. **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   Use this procedure to create the application fields for the PODETAIL record.

Step	Action
1	Click the PODETAIL application record with the right mouse button and select Edit Fields to access the Positional Field Editor dialog box.
2	If the field you are creating is the first field in the record, click **New**.
3	In the Name box, type "CUSTNUMDET".
4	Click the Mandatory option to identify it as a required field.
5	In the Description box, type "Customer Number" as the description of the field.
6	From the Data Type list, select the "String" as the field type.
7	If the Data Type is "Number" or "Dt/Tm", select the appropriate format from the Format list.
8	If the minimum length of the field is "0," accept that default in the Min Length box. Otherwise, type "1".
9	In the Max Length box, type "8" as the maximum field length.
10	To add the field and to create the next field positioned after it, press **Enter**.
11	Create the rest of the fields according to *Application fields* on page 2 - 36. **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)

Step	Action
12	After adding the last field, click **Auto Position** to automatically position the fields in the record. Click **Yes** to acknowledge the warning message that fields will be sequenced in order.
	The Positional Field Editor dialog box should now look like this:
13	When you are completely finished adding fields to the PODETAIL record, click **Close**.
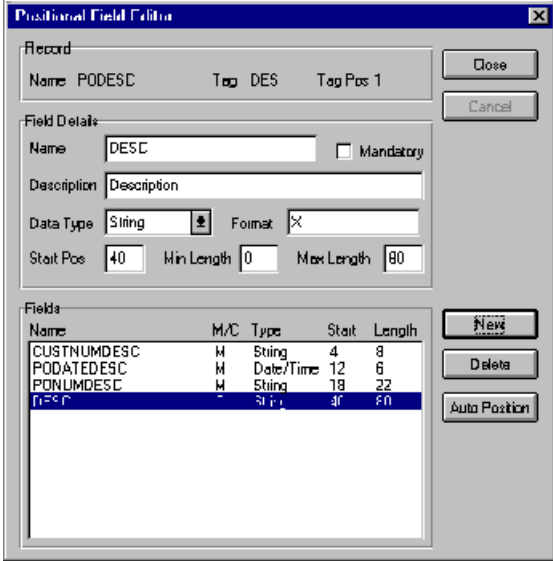
How to Create the PODESC Record Application Fields

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

Step	Action
1	Click the PODESC application record with the right mouse button and select Edit Fields to access the Positional Field Editor dialog box.
2	If the field you are creating is the first field in the record, click **New**.
3	In the Name box, type "CUSTNUMDESC".
4	Select the Mandatory check box to identify it as a required field.
5	In the Description box, type "Customer Number" as the description.
6	From the Data Type list, select "String" as the type of the field, to specify that this field must contain alphanumeric data.
7	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list.
8	If the minimum length of the field is "0," then accept that default in the Min Length box. Otherwise, type "1".
9	In the Max Length box, type "8" as the maximum length of the field.
10	To add the field and to create the next field positioned after it, press **Enter**.
11	Create the rest of the fields according to *Application fields* on page 2 - 36. **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
12	After adding the last field, click **Auto Position** to automatically position the fields in the record. Click **Yes** to acknowledge the warning message. The Positional Field Editor dialog box should now look like the following:
13	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.
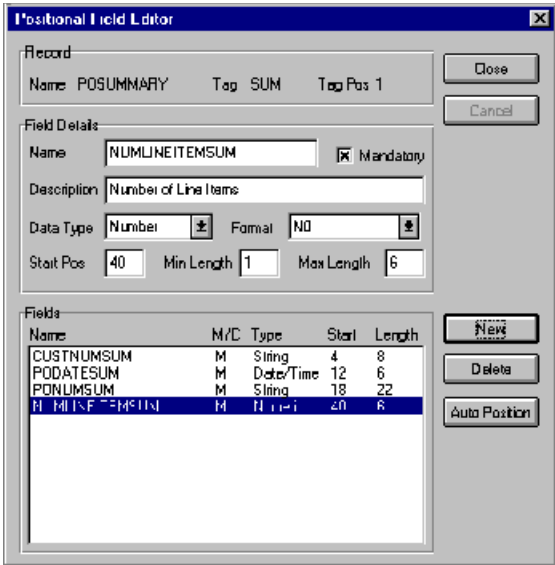
Step	Action
1	Click the POALLCHG application record with the right mouse button and select Edit Fields from the shortcut menu to access the Positional Field Editor dialog box.
2	If the field you are creating is the first field in the record, click **New**.
3	In the Name box, type "CUSTNUMAC".
4	Select the Mandatory check box to identify it as a required field.
5	In the Description box, type "Customer Number" as the description of the field.
6	From the Data Type list, select "String" as the type of field.
7	If the Data Type is "Number" or "Dt/Tm", select the appropriate format from the Format list.
8	If the minimum length of the field is "0", then accept the default in the Min Length box. Otherwise, type "1".
9	In the Max Length box, type "8" as the maximum length of the field.
10	To add the field and create the next field positioned after it, press **Enter**.
11	Create the rest of the fields according to *Application fields* on page 2 - 36. **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)

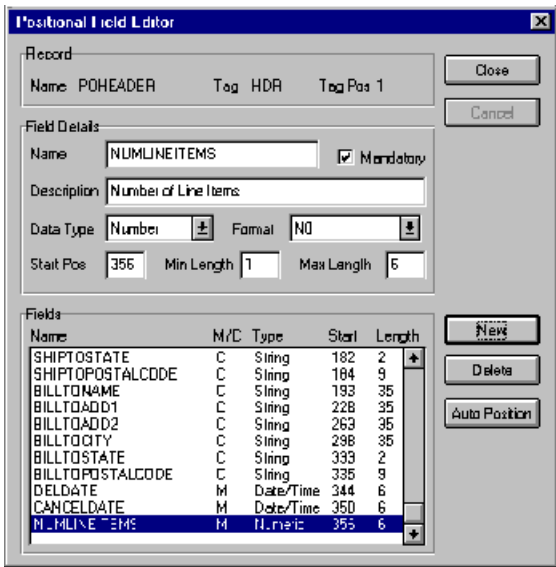| (Contd) Step	Action
12	After adding the last field, click **Auto Position** to automatically position the fields in the record. Click **Yes** to acknowledge the warning message that fields are sequenced in order. The Positional Field Editor dialog box should now look like the following:
13	When you 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	Click the POSUMMARY application record with the right mouse button to access the Positional Field Editor dialog box from the shortcut menu.
2	If the field you are creating is the first field in the record, click **New**.
3	In the Name box, type "CUSTNUMSUM", which is the name of the first field in the POSUMMARY record.
4	Select the Mandatory check box to identify it as a required field.
5	In the Description box, type "Customer Number" 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.
6	From the Data Type list, select "String" as the field type.
7	If the Data Type is "Number" or "Dt/Tm", select the appropriate format from the Format list.
8	If the minimum length of the field is "0", then accept that default in the Min Length box. Otherwise, type "1".
9	In the max Length box, type "8" as the maximum length of the field.
10	To add the field and to create the next field positioned after it, press **Enter**.
11	Create the rest of the fields according to *Application fields* on page 2 - 36. **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
12	After adding the last field, click **Auto Position** to automatically position the fields in the record. Click **Yes** to acknowledge the warning message that fields are sequenced in order. The Positional Field Editor dialog box should now look like the following:
13	When you are finished adding fields to the POSUMMARY record, click **Close**.

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

Mapping Your Data

Overview

Introduction   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 how to map each field in the POHEADER (header) record. The POHEADER (header) record contains:

◗   the purchase order type code

◗   shipping and billing information

◗   delivery and 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 by means of 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 map information that you use includes any field in a table (location, lookup, or cross-reference) or data from the partner record.

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

How to map the customer number
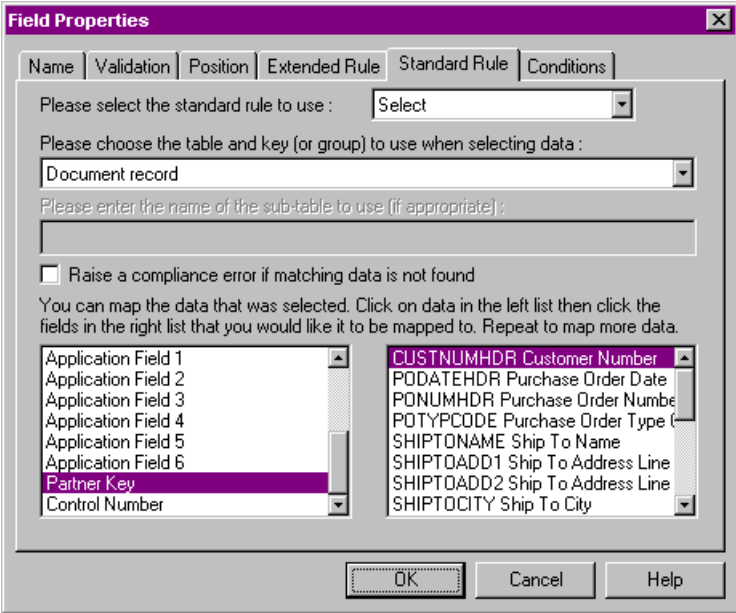
Your partner's customer number is part of the record key for each record in the application file format. It identifies which trading partner sent the purchase order. This customer number is already part of the partner definition for this partner so you can map it from the partner definition to each customer 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 to access the standard rule options.
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	From the Map From list on the left side, select "Partner Key." This is the field that contains the customer number for this partner. *(Continued on next page)*

(Contd) Step	Action
6	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:
7	Click **OK** to add the standard rule to the CUSTNUMHDR field.

PODATEHDR Field

Overview

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

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

To map the correct information from the DTM segment to the PODATEHDR field in the application header record, you must determine the data type, then extract the order date/time information.
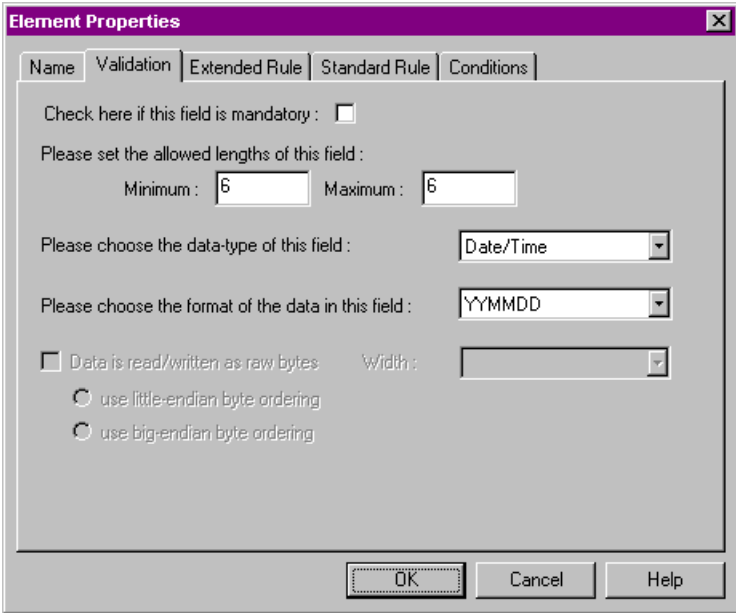
How to change the properties of the DTM02 element

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

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

Step	Action
1	Double-click the DTM02 element to display the Element Properties dialog box.
2	Select the **Validation** tab.
3	In the Minimum box, type "6" as the minimum length.
4	In the Maximum box, type "6" as the maximum length.
5	From the data-type list, select "Date/Time." This indicates that this is a date/time element. The system may prompt you with a message that changing the field's data type invalidates its format. If so, click **Yes** to acknowledge this message and proceed. *(Continued on next page)*

(Contd) Step	Action
6	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.

The Validation tab should now look like this:

 |
| 7 | Click **OK** to exit the Element Properties dialog. |

Temporary storage segments and elements

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

How to create the TEMPDATES temporary storage segment

Next, you will create the temporary storage segment details using this information:

◗ Name: TEMPDATES

◗ Tag: $$$

◗ Min Usage: 0

◗ Max Usage: 1

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**. The EDI Segment Properties dialog is displayed.
3	In the Name box, type "TEMPDATES".
4	Select the **Tag** tab.
5	In the Tag box, type "$$$". Because the system does not read a segment with the value of "$$$" in the Tag box, it will not flag this temporary storage segment as an error during compliance checking.
6	Click **OK** to create the TEMPDATES temporary storage segment.

How to create the PODATE temporary storage element

Next, you will add an element to the temporary storage segment you just created using this information:

◗ Name: PODATE

◗ Description: Purchase Order Date

◗ Min: 6

◗ Max: 6

◗ Type: Date/Time

◗ Format: YYMMDD

Use this procedure to create the PODATE element.

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.
3	In the Name box, type "PODATE".
4	In the Description box, type "Purchase Order 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 data-type list, select "Date/Time." If the system prompts you with a message box, click **Yes** to acknowledge the message prompt and proceed. *(Continued on next page)*

(Contd) Step	Action
9	From the Format list, select "YYMMDD", to indicate how the element is formatted. The Validation tab should now look like this:
10	Click **OK** to create the PODATE element.

How to use an extended rule

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

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. *(Continued on next page)*

(Contd) Step	Action

3	In the Rule list, type the following:
	
	``
	IF #2005 = "4" THEN
	BEGIN
	   $ORDERS.#PODATE=#2380;
	END
	``
	
	The rule is executed if the DTM01 element contains "4" (order date/time).

4	Click **Compile** to validate the extended rule syntax. Each rule in the map is compiled when you compile the translation object. The system allows you to compile each rule individually so that you can verify its accuracy after you create it. **Note** This compiles the rule interactively and allows you to correct any generated errors. Errors or warnings that occur during the compilation process are displayed in the Compile Errors box.
5	Click **OK** to add the extended rule to the DTM01 element.

How to map the order date/time

Use the Link function to map the PODATEHDR field directly from the PODATE temporary storage element. 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.

Use this procedure to map the order date/time.

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 PODATE temporary storage element in the TEMPDATES segment. When you click the element, the cursor changes to a Link Arrow. LINK ⇨
3	Click the application field PODATEHDR in the POHEADER record. A line is displayed, visually illustrating the link between the PODATE element and the PODATEHDR field.

PONUMHDR Field

Overview

Use the Link function to map the PONUMHDR field directly from the BGM05 (Document/Message Number). Your partner will only send you up to 22 characters for the BGM05 element (Document/Message Number), so you must change the maximum length allowed for that element.

We recommend that you set up a Document Name for each map to make identifying map messages easier. This allows you to differentiate between messages in the document browsers in Gentran:Server for Windows.

To set up a Document Name, select an element or field in the map that contains data that distinguishes the messages translated by this map. The Name field of the document browsers that contain this message contain the data from the selected element or field.

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

How to map the document/message number

To map the document/message number, link the BGM05 element in the BGM record to the application field PONUMHDR in the POHEADER record.

How to change the maximum length of the BGM05 element

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

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

Step	Action
1	Double-click the BGM05 element to display the Element Properties dialog box.
2	Select the **Validation** tab.
3	In the Maximum box, type "22" as the maximum length.
4	Continue with the next procedure.

How to set up the document name

With the Element Properties dialog box for the BGM05 element still displayed, use this procedure to set up the Document Name for this Purchase Order.

Step	Action
1	Select the **Standard Rule** tab to display the rule options.
2	From the standard rule list, select Update.
3	From the table list, select "Document record." This indicates that you are updating the document record in Gentran:Server for Windows.
4	From the column list, select "Document Name." This indicates that you are updating the Name column in the document browsers with the contents of this element. The Standard Rule tab should now look like this:
5	Click **OK** to set up the document name.

POTYPCODE Field

Overview   Use the Link function to map the POTYPCODE field directly from the BGM06 (Message Function, Coded) element.

Procedure   To map the coded message function, link the BGM06 element in the BGM record to the application field POTYPCODE in the POHEADER record.

How to Map the SHIPTONAME Information

Overview

On the EDI side of the map, the name and address information is contained in the NAD segment in the NAD group. The NAD group contains shipping and billing name and address information. It repeats multiple times. If the Party Qualifier (in the NAD segment) is "ST," that iteration of the NAD group contains shipping information. You must map the shipping information to the Ship To fields in the application header record. If the Party Qualifier is "BT," that iteration of the NAD group contains billing information. You must map the billing information to the Bill To fields in the application header record.

To map the correct information (either bill to or ship to) from the NAD 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. Because your partner will only send you up to two characters for the NAD20 element (Country Sub-Entity Identification), you must change the maximum length allowed for that element.

To map the shipping and billing information to your application file format, you must create temporary storage segments and elements on the EDI side of the map that do not repeat. Use an extended rule to extract the shipping and billing information from the NAD group and move it to the appropriate temporary storage elements. Map the shipping and billing information directly from the temporary storage elements to your application fields.

Temporary storage segments

This table lists the temporary storage segments that you will create.

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

How to create the shipping temporary storage segment

Use this procedure to create the ShipToDet temporary storage segment.

Step	Action
1	Highlight the NAD group. The two temporary storage segments are located after the NAD 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".
	(Continued on next page)

(Contd) Step	Action
5	Select the **Tag** tab.
6	On the Tag tab, type "$$$".
7	Click **OK** to create the ShipToDet temporary storage segment.

How to create the billing temporary storage 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. The EDI Segment Properties dialog box is displayed.
3	In the Name box, type "BillToDet".
4	In the Description box, type "Bill To Details".
5	Select the **Tag** tab.
6	On the Tag box, type "$$$".
7	Click **OK** to create the BillToDet temporary storage segment.

Temporary storage elements

This table lists the elements to be added to the temporary storage segments you just created.

EDI Segment	EDI Element	Max Length
ShipToDet	SHIPTONAME	35
	SHIPTOADDR1	35
	SHIPTOADDR2	35
	SHIPTOCITY	35
	SHIPTOSTATE	2
	SHIPTOPCODE	9
	(Continued on next page)	

EDI Segment	EDI Element	Max Length
BillToDet	BILLTONAME	35
	BILLTOADDR1	35
	BILLTOADDR2	35
	BILLTOCITY	35
	BILLTOSTATE	2
	BILLTOPCODE	9

How to create the shipping temporary storage 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.
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 and create the remaining Ship To elements.
7	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 for the remaining elements in the ShipToDet EDI segment listed in the table on page 2 - 64.

How to create the billing temporary storage 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 Field Properties dialog box is displayed.

(Contd) Step	Action
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 and create the remaining Bill To elements.
7	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 for the remaining elements in the BillToDet EDI segment listed in the table on page 2 - 64.

How to change the maximum length of the NAD20 element

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

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

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

Step	Action
How to use an extended rule	Use an extended rule to extract the shipping and billing information from the NAD group and move it to the appropriate temporary storage element. After the system reads an occurrence of the NAD group and executes the extended rule. Use this procedure to create the extended rule.

Step	Action
1	Right-click the NAD group and select **Extended Rules** to display the Group Properties dialog box with the Loop Extended Rules tab selected.
2	Click the On End option to specify that the rule will be executed when the loop terminates. The system loads an occurrence of the NAD group (containing the billing or shipping information) and then executes this rule.
| 3 | In the Editor box, type the following:<br><br>```
IF #3035 = "BT" THEN
BEGIN
    $ORDERS.#BILLTONAME = #3036;
    $ORDERS.#BILLTOADDR1 = #3042;
    $ORDERS.#BILLTOADDR2 = #3042:2;
    $ORDERS.#BILLTOCITY = #3164;
    $ORDERS.#BILLTOSTATE = #3229;
    $ORDERS.#BILLTOPCODE = #3251;
END
IF #3035 = "ST" THEN
BEGIN
    $ORDERS.#SHIPTONAME = #3036;
    $ORDERS.#SHIPTOADDR1 = #3042;
    $ORDERS.#SHIPTOADDR2 = #3042:2;
    $ORDERS.#SHIPTOCITY = #3164;
    $ORDERS.#SHIPTOSTATE = #3229;
    $ORDERS.#SHIPTOPCODE = #3251;
END
```<br><br>**Note**<br>If a segment/record or element/field occurs more than once in a map, it is identified by its name <ID>. Subsequent occurrences are identified by <ID>:n, where 'n' is the number of the occurrence in the map. |
| 4 | Click **Compile** to validate the syntax of the extended rule. |
| 5 | Click **OK** to add the extended rule to the NAD group. |

Mapping the ship to name information

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

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

To map POHEADER field	You need to link
SHIPTONAME	the SHIPTONAME element in the ShipToDet temporary storage segment to the application field SHIPTONAME in the POHEADER record.
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.
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.

(Contd) To map POHEADER field	You need to link
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.

DELDATE Field

Overview

The DTM segment contains all the purchase order date information. It repeats multiple times. If the DTM01 (Date/Time/Period/Qualifier) is "117", that iteration of the DTM segment contains the First Delivery Date/Time. You must map the delivery information to the DELDATE field in the application header record. If the DTM01 (Date/Time/Period/Qualifier) is "153", that iteration of the DTM segment contains the Latest Cancellation Date/Time. You must map the cancel by date information to the CANCELDATE field in the application header record.

To map the delivery and cancel by date information to your application file, you must create temporary storage elements in the TEMPDATES temporary storage segment that you have already created on the EDI side of the map. 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. Map the delivery and cancel by date information directly from the temporary storage elements to your application fields.

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

How to create temporary storage date elements

Use this procedure to create the DELDATE and CANCELDATE temporary storage elements.

Step	Action
1	Highlight the PODATE element (in the TEMPDATES segment) and select **Insert** from the Edit menu. From the submenu, select **Element** to display the Element Properties dialog box.
2	In the Name box, type "DELDATE".
3	In the Description box, type "Delivery Date".
4	Select the **Validation** tab.
5	In the Minimum box, type "6" as the minimum length.
6	In the Maximum box, type "6" as the maximum length.
	(Continued on next page)

(Contd) Step	Action
7	From the data-type list, select "Date/Time" to indicate that this is a date/ time element. If the system prompts you with a message that changing the field's data 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 a format or data type of a field" check box. Select **OK** to save.
8	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.
9	Click **OK** to create the DELDATE element. Create the CANCELDATE element next.
10	With the DELDATE element highlighted, select **Insert** from the Edit menu. From the submenu, select **Field** to display the Element Properties dialog box.
11	In the Name box, type "CANCELDATE".
12	In the Description box, type "Cancel By Date".
13	Select the **Validation** tab.
14	In the Minimum box, type "6" as the minimum length.
15	In the Maximum box, type "6" as the maximum length.
16	From the data-type list, select "Date/Time" to indicate that this is a date/ time element. *(Continued on next page)*

(Contd) Step	Action
17	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. The Element Properties dialog box should now look like this:
18	Click **OK** to create the CANCELDATE element.

How to create 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. After the system reads the DTM01 (Date/Time/Period/Qualifier), it executes the extended rule.

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. *(Continued on next page)*

(Contd) Step	Action

3	The Rule list already contains the extended rule that extracts the order date/time from the DTM segment (if the Date/Time/Period/Qualifier is "4") and moves it to the PODATE temporary storage element. In the Rules list below the existing extended rule, type the following:
	
	``IF #2005 = "117" THEN
	BEGIN
	   $ORDERS.#DELDATE = #2380;
	END
	IF #2005 = "153" THEN
	BEGIN
	   $ORDERS.#CANCELDATE = #2380;
	END```

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

How to map the DELDATE element

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

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

How to map the CANCELDATE element

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.

How to Map NUMLINEITEMS

Overview

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

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

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

How to create the TEMPCONTROL temporary storage segment

Create the temporary storage segment details using this information:

◗ Name: TEMPCONTROL

◗ Tag: $$$

◗ Min Usage: 0

◗ Max Usage: 1

Use this procedure to create the TEMPDATES temporary storage segment.

Step	Action
1	Highlight the CNT segment. The temporary storage segment is located after the CNT segment 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 "TEMPCONTROL".
4	Select the **Tag** tab.
5	In the Tag box, type "$$$".
6	Click **OK** to create the TEMPCONTROL temporary storage segment.

How to create the NUMLINEITEMS temporary storage element

Add an element to the temporary storage segment you just created. Use this information to create the temporary storage element.

◗ Name: NUMLINEITEMS

◗ Description: Number of Line Items

◗ Max: 6

◗ Type: Number

◗ Format: R9

Use this procedure to create the NUMLINEITEMS element.

Step	Action
1	Highlight the TEMPCONTROL 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.
3	In the Name box, type "NUMLINEITEMS".
4	In the Description box, type "Number of Line Items".
5	Select the **Validation** tab.
6	In the Maximum box, type "6" as the maximum length.
7	From the data-type list, select "Number." (Continued on next page)

(Contd) Step	Action
8	From the Format list, select "R9." This indicates that the element will be formatted with an explicit decimal point and contain up to nine places. The Element Properties dialog box should now look like this:
9	Click **OK** to create the NUMLINEITEMS element.

How to create an extended rule

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

Use this procedure to create the extended rule.

Step	Action
1	Right-click the CNT segment and select Extended Rules to display the EDI Segment Properties dialog box with the Loop Extended Rules tab selected.
2	Select the On End option. This specifies that the rule will be executed when the loop terminates. The system loads an occurrence of the CNT segment (containing the control value information), and executes this rule. *(Continued on next page)*

(Contd) Step	Action

3	In the Editor box, type the following:
	
	``
	IF #6069 = "2" THEN
	BEGIN
	  $ORDERS.#NUMLINEITEMS = #6066;
	END
	``

4	Click **Compile** to validate the extended rule syntax.
5	Click **OK** to add the extended rule to the CNT segment.

How to map the total number of line items

Use the Link function to map the NUMLINEITEMS field directly from the NUMLINEITEMS temporary storage element.

To map the number of line items to the application header record, link the NUMLINEITEMS temporary storage element in the TEMPCONTROL temporary storage segment to the application field NUMLINEITEMS in the POHEADER record.

Mapping the PODETAIL Record

Overview

Introduction   The PODetailGroup (detail information) contains all line item information for the order, including the item, unit of measure, price, quantity ordered, description, and the allowance and charge information. Each PODetailGroup occurs up to 200,000 times per order translated. This equals a maximum order of 200,000 items. The information within the PODetailGroup is further segregated by record:

- ◗ PODETAIL,
- ◗ PODESC, and
- ◗ POALLCHG.

This section describes how to map 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 by means of a standard rule. 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 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 pull information from the partner definition to populate the Customer Number fields on the application side of the map. Since the Customer Number is 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" to indicate that the system will update this field with selected information from the document record.
5	From the Map From list on the left, select "Partner Key." This field contains the customer number for this partner. *(Continued on next page)*

(Contd) Step	Action
6	From the Map To list on the right, select "CUSTNUMDET." Partner Editor information will be mapped to this field. The Standard Rule tab should now look like this:
7	Click **OK** to add the standard rule to the CUSTNUMDET field.

PODATEDET Field

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

You created a temporary storage segment and element on the EDI side of the map that does not repeat. You used an extended rule to extract the order date/time information from the DTM segment and move it to the PODATE temporary storage element.

The TEMPDATES temporary storage segment does not repeat. You must map the information from the PODATE temporary storage element to the PODATEDET field. The PODATEDET field is included in the PODETAIL segment in the PODetailGroup, which repeats 200,000 times.

You must 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 TEMPDATES temporary storage segment directly to your application PODATEDET field.

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

How to create an extended rule   Use this procedure to create an 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 = $ORDERS.#PODATE;` The "$ORDERS.#PODATE" is the addressing for the PODATE temporary storage element. After the system processes the PODATEDET field, it executes this rule.
4	Click **Compile** to validate the extended rule syntax.
5	Click **OK** to add the extended rule to the PODATEDET field.

PONUMDET Field

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

You can only map fields and elements that are at the same level. To map the document/message number to this application field, you must use an extended rule to extract the purchase order number from the BGM05 element and move it to the PONUMDET field.

How to create an extended rule   Use this procedure to create an 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 = $ORDERS.#1004;` The "$ORDERS.#1004" is the addressing for the BGM05 element. After the system processes the PONUMDET field, it executes this rule.
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 by means of 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.

You must map information from a cross-reference table for the PETZONE1 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. 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 must map the customer product code from the LIN03 element (Item Number) to the customer product code field (CUSTPROCODE) on the application side of the map. In order to achieve this, you must create a cross-reference table to translate the customer product code to your UPC code.

Note
Before you create the cross-reference table, you and your partner must determine what the equivalent customer product code is for each UPC code. Then you can create the cross-reference table in Partner Editor and use that information in your map.

Finally, you must use a standard rule to map the translated value from the CUSTPROCODE field to the UPCCODE field.

How to map the customer product code

To map the customer product code, link the LIN03 element in the LIN segment (in the LIN group) to the application field CUSTPROCODE in the PODETAIL record.

How to create a cross-reference table

In the Partner Editor, create a cross-reference table named "PRODCODE". See the Gentran:Server for Windows *User's Guide* or online Help for more information on creating a cross-reference table.

Note
The cross-reference table you create will only contain six entries. This is an example – an actual table would contain all of the items your company manufactures that your partner purchases.

How to create a cross-reference table

Use the information in the following table when you create 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	5 Kg. Bag
34567890123456	555666908	Small Beef Basted Rawhide Bones	Bag of 12
45678901234567	555666321	X-Large Food Bowl	Black
56789012345678	555666287	FatKat Kitty Bed	Blue with White Trim
67890123456789	555666413	Catnip Mouse Toy	

How to translate CUSTPROCODE to UPCCODE

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 CUSTPROCODE 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 list, select "Partner cross-reference by partner item." This indicates that you are using a cross-reference table and translating by your partner's item.
5	In the Sub Table box, type "PRODCODE". This is the name of the cross-reference table you created.
6	Select the raise a compliance error check box to signal the generate a compliance error if the translation fails.
7	From the Map From list on the left, select "My Item" to map the contents from this field.

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

UOM Field

Overview

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

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

How to create the TEMPLINEITEM temporary storage segment

Use this procedure to create the TEMPLINEITEM temporary storage segment.

Step	Action
1	Highlight the LIN segment (in the LIN group). From the Edit menu, select Insert. From the submenu, select Segment. **System response** The EDI Segment Properties dialog box is displayed.
2	In the Name box, type "TEMPLINEITEM".
3	Select the **Tag** tab.
4	In the Tag box, type "$$$".
5	Click **OK** to create the TEMPLINEITEM temporary storage segment.

How to create the UNITMEAS temporary storage element

Use this procedure to create the UNITMEAS temporary storage element.

Step	Action
1	Highlight the TEMPLINEITEM segment. From the Edit menu, select **Create Sub** to create a subordinate element. From the submenu, select **Element**. **System response** The Element Properties dialog box is displayed.
2	In the Name box, type "UNITMEAS".
3	In the Description box, type "Unit of Measurement".
4	Select the **Validation** tab.

(Contd) Step	Action
5	In the Maximum box, type "3" as the maximum length.
6	Click **OK** to create the UNITMEAS element.

How to create an extended rule

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

Use this procedure to create an extended rule.

Step	Action
1	Double-click the QTY:303 element to access the Element Properties dialog box.
2	Select the **Extended Rule** tab.
3	In the Rule box, type the following: ```IF #6063 = "21" THEN\nBEGIN\n $2500_LIN.#UNITMEAS = #6411:2;\nEND``` The system verifies that the QTY:303 element does not contain "21" (quantity ordered) and executes this rule.
4	Click **Compile** to validate the syntax of the extended rule.
5	Click **OK** to add the extended rule to the QTY:303 element.

Mapping the unit of measurement

Use the Link function to map the UOM (Unit of Measurement) field directly from the UNITMEAS temporary storage element.

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

UNITPR Field

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

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

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

**How to create the
UNITPRICE
temporary storage
element**

Use this procedure to create the UNITPRICE temporary storage element.

Step	Action
1	Highlight the UNITMEAS temporary storage element (in the TEMPLINEITEM segment). From the Edit menu, select **Insert** to create a subordinate element. From the submenu, select Element. **System response** The Element Properties dialog box is displayed.
2	In the Name box, type "UNITPRICE".
3	In the Description box, type "Unit Price".
4	Select the **Validation** tab.
5	In the Maximum box, type "14" as the maximum length.
6	From the data-type list, select "Number." (Continued on next page)

(Contd) Step	Action
7	From the Format list, select "R9" to indicate that the element will be formatted with an explicit decimal point and contain up to nine places. The Element Properties dialog box should now look like this:
8	Click **OK** to create the UNITPRICE element.

How to create an extended rule

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

Step	Action
1	Double-click the MOA:501 element to access the Element Properties dialog box.
2	Select the **Extended Rule** tab.

3	In the Rule box, type the following:
	
	``
	IF #5025 = "145" THEN
	BEGIN
	  $2500_LIN.#UNITPRICE = #5004;
	END
	``
	
	The system verifies that the MOA:501 element does not contain "146" (unit price) and executes this rule.
	
	(Continued on next page)

(Contd) Step	Action
4	Click **Compile** to validate the syntax of the extended rule.
5	Click **OK** to add the extended rule to the MOA:501 element.

Mapping the unit price

Map the UNITPR (Unit Price) field directly from the UNITPRICE temporary storage element, by using the Link function.

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

QTYORD Field

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

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

Use an extended rule to extract the quantity ordered from the QTY:3 segment and move it to the appropriate temporary storage element. Then map the quantity ordered directly from the temporary storage element to the QTYORD application field.

How to create the QTYORD temporary storage element   Use this procedure to create the QTYORD temporary storage element.

Step	Action
1	Highlight the UNITPRICE temporary storage element (in the TEMPLINEITEM segment). From the Edit menu, select **Insert** to create a subordinate element. From the submenu, select **Element**. **System response** The Element Properties dialog box is displayed.
2	In the Name box, type "QTYORD".
3	In the Description box, type "Quantity Ordered".
4	Select the **Validation** tab.
5	In the Maximum box, type "9" as the maximum length.
6	From the data-type list, select "Number." *(Continued on next page)*

(Contd) Step	Action
7	From the Format list, select "R9" to indicate that the element will be formatted with an explicit decimal point and contain up to nine places. The Validation tab should now look like this:
8	Click **OK** to create the QTYORD element.

How to create an extended rule

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

Use this procedure to create the extended rule.

Step	Action
1	Double-click the QTY:301 element to access the Element Properties dialog box.
2	Select the **Extended Rule** tab. *(Continued on next page)*

(Contd) Step	Action

3	In the Rule box, type the following:
	
	``
	IF #6063 = "21" THEN
	BEGIN
	   $2500_LIN.#QTYORD = #6060;
	END
	``
	
	The system verifies that the QTY:301 element does contain "21" (quantity ordered) and executes this rule.

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

Mapping the quantity ordered

Use the Link function to map the QTYORD (quantity ordered) directly from the QTYORD temporary storage element.

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

Mapping the PODESC Record

Overview

Introduction   This section describes how to map each field in the PODESC (description) record. The PODESC (description) record contains descriptive information about the line item ordered. The PODESC record occurs 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 by means of 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" to update this field with information from the document record.
5	Select the raise a compliance error check box to generate a compliance error if the system does not find a valid entry.
6	From the Map From list on the left, select "Partner Key." This 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." Partner Editor information is mapped to this field. The Standard Rule tab should now look like this:
8	Click **OK** to add the standard rule to the CUSTNUMDESC field.

PODATEDESC Field

Overview

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

You have already created a temporary storage segment and element on the EDI side of the map that do not repeat. You used an extended rule to extract the order date/time information from the DTM segment and move it to the PODATE temporary storage element.

The TEMPDATES temporary storage segment does not repeat. You must map the information from the PODATE temporary storage element to the PODATEDESC field. The PODATEDESC field is included in the PODESC segment in the PODetailGroup, which repeats 200,000 times.

You 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 TEMPDATES temporary storage segment directly to your application PODATEDESC field.

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

How to create an extended rule

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 box, type the following: `#PODATEDESC = $ORDERS.#PODATE;` After the system processes the PODATEDESC field, it executes this rule.
4	Click **Compile** to validate the extended rule syntax.
5	Click **OK** to add the extended rule to the PODATEDESC field.

PONUMDESC Field

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

You 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 document/message number in the BGM segment directly to your application PONUMDESC field.

To map the document/message number to this application field, you must use an extended rule to extract the purchase order number from the BGM05 element and move it to the PONUMDESC field.

How to create an extended rule   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 box, type the following: `#PONUMDESC = $ORDERS.#1004;` After the system processes the PONUMDESC field, it executes this rule.
4	Click **Compile** to validate the syntax of the extended rule.
5	Click **OK** to add the extended rule to the PONUMDESC field.

How to map the DESC field   The DESC field is available so your partner can describe the item in a free-form format. You need to change the maximum usage of the IMD:2 segment, because you know that your partner will only send you five iterations of the segment.

Changing the max usage
Use this procedure to change the properties of the IMD:2 segment.

Step	Action
1	Right-click the IMD:2 segment (LIN group). From the shortcut menu, select **Properties** to access the EDI Segment Properties dialog box.
2	Select the **Looping** tab.

(Contd) Step	Action
3	In the Max Usage box, type "5".
4	Click **OK** to exit the EDI Segment Properties dialog box.

Linking the DESC field
Use the Link function to map the DESC (description) field directly from the IMD:206 element.

Link the IMD:206 element in the IMD:2 segment (in the LIN group) to the application field DESC in the PODESC record.

Mapping the POALLCHG Record

Overview

Introduction

This section describes how to map 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 occurs 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 by means of 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 CUSTNUMAC field (in the POALLCHG record) to display the Field Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, choose **Select**.
4	From the table and key list, select "Document record."
5	Select the raise a compliance error check box.
6	From the Map From list on the left, select "Partner Key." This contains the customer number for this partner. *(Continued on next page)*

(Contd) Step	Action
7	From the Map To list on the right, select "CUSTNUMAC." The Standard Rule tab should now look like this:
8	Click **OK** to add the standard rule to the CUSTNUMAC field.

PODATEAC Field

Overview   On the EDI side of the map, the date information is contained in the DTM segment. The DTM segment contains date information for the header part of the map, and repeats multiple times. If the DTM01 (Date/Time/Period Qualifier) is "4," that iteration of the DTM segment contains the order date/time. You need to map the order date/time to the PODATE field in each the of the application records (PODATEHDR, PODATEDET, PODATEDESC, PODATEAC, and PODATESUM).

You have already created a temporary storage segment and element on the EDI side of the map that do not repeat. And you used an extended rule to extract the order date/time information from the DTM segment and move it to the PODATE temporary storage element.

The TEMPDATES temporary storage segment does not repeat. You must map the information from the PODATE temporary storage element to the PODATEAC field. The PODATEAC field is included in the POALLCHG segment in the PODetailGroup, which repeats 200,000 times.

To map the purchase order date to this application field, you must use an extended rule to extract the purchase order date from the PODATE temporary storage element and move it to the PODATEAC field.

How to create an extended rule   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 = $ORDERS.#PODATE;` After the system processes the PODATEAC field, it executes this rule.
4	Click **Compile** to validate the extended rule syntax.
5	Click **OK** to add the extended rule to the PODATEAC field.

PONUMAC Field

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

To map the document/message number to this application field, you must use an extended rule to extract the purchase order number from the BGM05 element and move it to the PONUMAC field.

How to create an extended rule
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 = $ORDERS.#1004;` After the system processes the PONUMAC field, it executes this rule.
4	Click **Compile** to validate the extended rule syntax.
5	Click **OK** to add the extended rule to the PONUMAC field.

ACIND Field

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

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

Step	Action
1	Double-click the ALC:201 element to display the Element Properties dialog box.
2	Select the **Standard Rule** tab.
3	From the standard rule list, select **Use Code**.
4	Click **Edit** to access the Edit Code List dialog box. The Table ID contains the element for which this code list table is used. The Description box contains the element description for which this code list table is used. *(Continued on next page)*

(Contd) Step	Action
5	◗ Click **Load** to access the Select Data Source Name dialog box. ◗ Select the **Machine Data Source** tab. ◗ Select the Data Source Name that the system uses to access the EDI message standards (probably Gentran Standards). ◗ Click **OK**. **System response** The system displays the Load Code List dialog box.
6	In the Codes in Standard list, click the "A Allowance" code. Press the **SHIFT** key and click the "C Charge" code to highlight both codes. Click **Add** to move both codes to the Codes Selected list.
7	Click **OK** to load the selected codes and exit the Load Code List dialog box. The Edit Code List dialog box should now look like this.
8	Click **OK** to exit the Edit Code List dialog box. On the Standard Rule tab, verify that the code list contains "5463 ALLOWANCE OR CHARGE QUALIFIER."
9	Select the raise a compliance error check box to specify that, for compliance reasons, the element *must* contain one of two codes that you loaded from the code table list.
10	Click **OK** to exit the Element Properties dialog box.

**How to map the
allowance or
charge indicator**

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

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

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

ACMETHAND Field

Overview

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

How to map the allowance or charge method of handling code

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

Note

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

ACTOTAMT Field

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

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

You can map the allowance or charge total amount directly from the MOA:802 element to the ACTOTAMT application field.

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

Step	Action
1	Right-click the MOA group (ALC group, LIN group). From the shortcut menu, select **Properties** to access the Group Properties dialog box.
2	Select the **Looping** tab.
3	In the Maximum Usage box, type "1" to make the MOA group a non-repeating group. *(Continued on next page)*

(Contd) Step	Action
4	Select the Promote records to parent check box to specify that when this group is compiled, the subordinate segments will be extracted and placed in the parent group. This locates the MOA:8 segment at the same level as the POALLCHG record on the application side of the map. The Group Properties dialog box should now look like this: **Group Properties** ✕ Name | Looping | Minimum Usage [0] ☑ Promote records to parent Maximum Usage [1] [OK] [Cancel] [Help]
5	Click **OK** to exit the Group Properties dialog box.

How to map the allowance or charge total amount

Map the ACTOTAMT (allowance or charge total amount) field directly from the MOA:802 element, which indicates the handling method for the allowance or charge.

Link the MOA:802 element in the MOA:8 segment (ALC group, LIN group) to the application field ACTOTAMT in the POALLCHG record.

Note

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

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 Partner Windows Editor in your maps by means of 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 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 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."
5	Select the raise a compliance error check box.
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." The Standard Rule tab should now look like this:
8	Click **OK** to add the standard rule to the CUSTNUMSUM field.

PODATESUM Field

Overview
On the EDI side of the map, the date information is contained in the DTM segment. The DTM segment contains date information for the header part of the map, and repeats multiple times. If the DTM01 (Date/Time/Period Qualifier) is "4," that iteration of the DTM segment contains the order date/time. You need to map the order date/time to the PODATE field in each the of the application records (PODATEHDR, PODATEDET, PODATEDESC, PODATEAC, and PODATESUM).

To map the correct information from the DTM segment to the PODATESUM field in the application header record, you must determine the data type and extract the order date/time.

You have already created a temporary storage segment and element on the EDI side of the map that does not repeat. To map the order date/time information to your application file format, you used an extended rule to extract the order date/time information from the DTM segment and move it to the appropriate temporary storage element. You can map the order date/time directly from the temporary storage element to the PODATESUM application field.

How to map the order date/time
Use the Link function to map the PODATESUM field directly from the PODATE temporary storage element.

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

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

PONUMSUM Field

Overview　　You previously determined that your partner will only send you up to 22 characters for the BGM05 element (Document/Message Number), so you have already changed the maximum length allowed for that element. Use the Link function to map the PONUMSUM field directly from the BGM05 (Document/Message Number) field. The BGM05 element is already mapped (linked) to the PONUMHDR field in the POHEADER application record, so you will be adding an additional link from this element to the PONUMSUM field in the POSUMMARY record.

Procedure　　To map the document/message number, link the BGM05 element in the BGM record to the application field PONUMSUM in the POSUMMARY record.

NUMLINEITEMSUM Field

Overview

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

To map the correct information from the CNT segment to the NUMLINEITEMSUM field in the application summary record, you must extract the number of line items after determining the type of control value. You have already created a temporary storage segment (TEMPCONTROL) and element (NUMLINEITEMS) on the EDI side of the map that do not repeat. You used an extended rule to extract the number of line items from the CNT segment and move it to the appropriate temporary storage element.

You can map the number of line items directly from the temporary storage element to the NUMLINEITEMS application field. The NUMLINEITEMS temporary storage element is already mapped (linked) to the NUMLINEITEMS field in the POHEADER application record so you will be adding an additional link from this element to the NUMLINEITEMSUM field in the POSUMMARY record.

How to map the total number of line items

Link the NUMLINEITEMS temporary storage element in the TEMPCONTROL temporary storage segment 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 (except the UPCCODE field in the PODETAIL application record). The red checkmark indicates that a mapping operation has been performed against the field.

Finalizing the Map

Overview

Introduction   To complete the mapping process, you must save the map, compile the translation object, print and review the Gentran:Server report, and test the map.

How to Compile the Map

Overview   The Compile function generates a translation object. The map that you created (PET_EORD.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 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.
2	From the File menu, select **Compile** to display the Run-Time Translation Object Name dialog box.
3	The File Name box contains "pet_eord.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 as 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 will remain 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 will be 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**. The system compiles the map and generates a translation object. The Compile Errors dialog box is displayed. Verify that no errors occurred (scroll down to the bottom of the list). Click **OK** to exit the dialog box.
6	From the Edit menu, select **Details**. The Translation Object Details dialog box is displayed. The date on which the translation object was compiled is automatically loaded into the Compiled on box on the Translation Object Details dialog box. Click **OK** to exit the dialog box.
7	From the File menu, select **Save** to save the source map. **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's 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 and modify the map. If you review the report and determine that the map is incorrect, 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	Select the appropriate options and click **OK**.

Testing the Translation Object

Overview　　After you compile the map, print and verify the report, and register the translation object with Gentran:Server, test the translation object to verify that the data is translated correctly. To test the compiled translation object, obtain test data from your partner(s) and process the data. Verify acknowledgement processing (if applicable) and communications with your network.

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

Step	Action
1	Register the PET_EORD.TPL translation object with Gentran:Server for Windows.
2	Import the PETZONE1.PAR partner relationship into Gentran:Server for Windows.
3	Verify (in Partner Editor) that the PET E 92.1 ORDERS 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 (PETTSTE1.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 path for the data file is: `C:\GENSRVNT\TUTORIAL\PETTSTE1.INT`
5	After the message is translated, it is located in the In Documents in Gentran:Server for Windows. Export the message to ensure that it was translated correctly.

Reference

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

Outbound Invoice (INVOIC)

Overview

Introduction

In this chapter   This tutorial defines the tasks necessary to create the map used to translate your
application file to EDIFACT invoices (INVOIC). After you translate your application file
to EDIFACT, it is ready to send to your trading partner.

This chapter explains the processes associated with creating a map for an outbound
invoice. The processes include:

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

By completing each task in sequential order, you will learn individual mapping techniques and a logical approach to methods you will utilize 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 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. After shipping an order to Pet Zone, generate the invoice.

You need to create a map that enables you go generate the invoice and translate your application file into an EDI invoice to send 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. From discussions with Pet Zone, you know the requirements for the EDI invoices that you will be sending them.

Mapping analysis

The first and most important step in creating a successful map is to analyze your mapping requirements. If your mapping analysis is complete, you will have all the information you need to create the map in an efficient, logical manner. If you omit this critical step, it is likely that creating the map will be a much longer and more arduous task, and the map may be invalid due to oversights and omissions.

This table describes the steps used in mapping analysis.

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

Analyzing Your Application File Format

Overview

The first step of mapping analysis is to analyze your application file format, because this is the component that is probably most familiar to you. For an outbound map, your application file contains the data that you must convert to an EDI file so your system can generate the message correctly. In this tutorial, your application file contains all the information that you need to generate the invoice that you will send your partner.

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

System import map application file layout

The application file layout for the system import map in this tutorial is provided in this table.

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per message					
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

Import map application file layout

The application file layout for the import map in this tutorial is provided in this table.

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per message					
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 message					
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	35
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	35
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

(Continued on next page)

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
REMITTOADD1		Remit To Address Line 1	String	0	35
REMITTOADD2		Remit To Address Line 2	String	0	35
REMITTOCITY		Remit To City	String	0	35
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
Comment Group (INVCommentGroup) 10 per message					
Comment Record 1 (INVCOMMENT1) 1 per message					
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) 1 per message					
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 message					
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)

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
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
Summary Record (INVSUMMARY) 1 per message					
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 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 to ensure 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 individually identified. The record key for this invoice consists of:

◗ your partner's bill to account number

◗ the document type

◗ invoice number

Record/Group definitions

This table provides the Record/Group definitions.

Record/Group	Definition
HEADER (header)	Contains the five-field key (partner identifier, standard, version, transaction set (message), 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 and ship date. 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 message text. The INVCommentGroup occurs up to ten times 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 once per invoice translated (once for each comment line).
INVDETAIL (detail)	Contains line item number and description, quantity shipped, quantity ordered, unit of measure, price, and product code. 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 invoice requirements that you will send to your partner.

Analyzing the EDI File

Overview   To analyze the EDI file, determine what you want to send your trading partner and what your partner expects to receive. You and your partner must agree on which standard, version, and message to use. In this tutorial, we use an EDIFACT version 92.1 INVOIC. It is important to know what type of information your partner expects you to send within the invoice and what data will be 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 will use. Standards manuals are available from EDI standards 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).

EDI map components   This table lists the map components that are required, but are not currently active.

Groups		Segment	Comp.	Element
		BGM segment (already active)		BGM05 element
		DTM segment (already active)	C507	DTM02 element
		FTX segment	C108	FTX06 element FTX07 element
RFF group		RFF segment	C506	RFF02 element
		DTM:2 segment	C507	DTM:202 element
NAD group		NAD segment	C080 C059 C059	NAD10 element NAD16 element NAD17 element NAD19 element NAD20 element NAD21 element
		LOC segment	C517	LOC05 element
	CTA group	CTA segment	C056	CTA03 element
				(Continued on next page)

Groups		Segment	Comp.	Element
LIN group		LIN segment	C212 C212	LIN01 element LIN03 element LIN04 element
		PIA segment	C212 C212	PIA02 element PIA03 element
		IMD:2 segment	C273	IMD:206 element
		QTY:2 segment	C186	QTY:303 element
	MOA group	MOA:5 segment	C516	MOA:502 element
		CNT segment	C270	
MOA group (already active)		MOA:10 segment	C516	MOA:1002 element

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. This allows you to differentiate between messages in the document browsers.

Document name and reference

To set up a Document Name, select a map element or field that contains distinguishing data for messages translated by this map. We strongly recommend that you select an element or field that occurs only once in the message and has a data type of "String." In this tutorial, the system import document name is the document/message number.

You can also establish a Document Reference for each map to distinguish messages. When you set up the Reference in the Application Integration subsystem, the RefData field of the document browser that contains this message contains data from that mapped field. In this tutorial, the system import document reference is the reference (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 your analysis is determining in which elements you must define and use qualifiers and then determining which elements they qualify. You must use constant values that are compliant with the EDI standard.

Defining constants (contd)

This table lists the constants you must define to create the required qualifier elements.

Element Using Constant	Constant ID	Constant Value	Element it Qualifies
DTM:2201	Invoice date/time	3	DTM:2202
RFF:1101 (PONUM_RFF)	Order number (purchase)	ON	RFF:1102 (PONUM_RFF)
DTM:2501 (PONUM_RFF)	Order date/time	4	DTM:2502 (PONUM_RFF)
NAD:301 (Vend_NAD)	Vendor	VN	NAD:310 (Vend_NAD)
NAD:501 (RemitTo_NAD)	RemitTo	RE	NAD:510 (RemitTo_NAD)
NAD:401 (ShipTo_NAD)	ShipTo	ST	NAD:410 (ShipTo_NAD)
LOC:1301 (ShipTo_NAD)	Goods receipt place	4	LOC:1305 (ShipTo_NAD)
RFF:1201 (BOL_RFF)	Bill of lading number	BM	RFF:1202 (BOL_RFF)
DTM:2601 (BOL_RFF)	Despatch date/time	11	DTM:2602 (BOL_RFF)
FTX01	Invoice instruction	INV	FTX06/FTX07
QTY:501	Despatch quantity	12	QTY:502
QTY:601	Ordered quantity	21	QTY:602
MOA:1301	Unit price	146	MOA:1302
LIN04	Buyer's part number	BP	LIN03
PIA:203	UPC Code	UP	PIA:202
PIA:201	Additional Identification	1	PIA:202
UNS01	Summary	S	
CNT:201	Number of line items in message	2	CNT:202

(Continued on next page)

(Contd) Element Using Constant	Constant ID	Constant Value	Element it Qualifies
CNT:301	Total quantity	1	CNT:302
MOA:1401	Total amount	128	MOA:1402

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

Correlating Your Application File and the EDI File

Overview   To translate outbound data, you must build an import map and a system import map. The Application Integration subsystem uses the system import map to determine which import map to use. The system import map builds the key that the translator uses to find the partner relationship. The system import map's sole function is to identify the appropriate partner relationship; it does *not* map data.

To build the key in system import map header record, you can use one of two methods:

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

◗   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

◗   message

◗   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 that you use this method because it is flexible. You typically 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 three-field key method is easier to build because the header record only requires the 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. Because only three fields combine to build the key, this method is not as flexible as the five-field key method. You typically use the three-field key method when you use a legacy (existing) application definition and when you do not want to add EDI-specific information.

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

You must promote several groups and segments that repeat multiple times to extract one iteration (instance) of the group or repeating segment. This is necessary to map multiple application fields to the same segment or group.

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

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

Field Name	Description	Mapping Operations
Header Record (HEADER) 1 per message		
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

Import map operations   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 message		
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 message		
BILLTOACCTBEG	Bill to Acct Number	No mapping required
DOCTYPBEG	Document Type	No mapping required
INVNUMBEG	Invoice Number	Link to BGM05 and UNS01
INVDATE	Invoice Date	Link to DTM:2202 using "3" (invoice date/time) code
PONUM	Purchase Order Number	Link to RFF:1102 using "ON" (order number) code
PODATE	Purchase Order Date	Link to DTM:2502 using "4" (order date/time) code
VENDNAME	Vendor Name	Link to (Vend_NAD) NAD:310
VENDADD1	Vendor Address Line 1	Link to (Vend_NAD) NAD:316
VENDADD2	Vendor Address Line2	Link to (Vend_NAD) NAD:317
VENDCITY	Vendor City	Link to (Vend_NAD) NAD:319
VENDSTATE	Vendor State, Province, or Country	Link to (Vend_NAD) NAD:320

(Continued on next page)

(Contd) Field Name	Description	Mapping Operations
VENDPOSTALCODE	Vendor Postal Code	Link to (Vend_NAD) NAD:321
SHIPTONAME	Ship To Name	Link to (ShipTo_NAD) NAD:410
SHIPTOADD1	Ship To Address Line 1	Link to (ShipTo_NAD) NAD:416
SHIPTOADD2	Ship To Address Line2	Link to (ShipTo_NAD) NAD:417
SHIPTOCITY	Ship To City	Link to (ShipTo_NAD) NAD:419
SHIPTOSTATE	Ship To State, Province, or Country	Link to (ShipTo_NAD) NAD:420
SHIPTOPOSTALCODE	Ship To Postal Code	Link to (ShipTo_NAD) NAD:421
SHIPTOLOC	Ship To Store Location	Link to (ShipTo_NAD) LOC:1305 using "4" (goods receipt place) code
SHIPTODEPTNUM	Ship To Dept Number	Link to (ShipTo_NAD) CTA:603
REMITTONAME	Remit To Name	Link to (RemitTo_NAD) NAD:510
REMITTOADD1	Remit To Address Line 1	Link to (RemitTo_NAD) NAD:516
REMITTOADD2	Remit To Address Line2	Link to (RemitTo_NAD) NAD:517
REMITTOCITY	Remit To City	Link to (RemitTo_NAD) NAD:519
REMITTOSTATE	Remit To State, Province, or Country	Link to (RemitTo_NAD) NAD:520
REMITTOPOSTALCODE	Remit To Postal Code	Link to (RemitTo_NAD) NAD:521
BOLNUM	Bill of Lading Number	Link to RFF:1202 using "BM" (bill of lading number) code

(Contd) Field Name	Description	Mapping Operations
SHIPDATE	Ship Date	Link to DTM:2602 using "11" (despatch date/time) code
Comment Group (INVCommentGroup) 10 per message		
Comment Record 1 (INVCOMMENT1) 1 per message		
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 FTX06 using "INV" (invoice instruction) code
Comment Record 2 (INVCOMMENT2) 1 per message		
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 FTX07
Detail Record (INVDETAIL) up to 200,000 per message		
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 LIN01
QTYSHIP	Quantity Shipped	Link to QTY:502 using "12" (despatch quantity) code
QTYORD	Quantity Ordered	Link to QTY:602 using "21" (ordered quantity) code
UNITPRICE	Unit Price	Link to MOA:1302 using "146" (unit price) code
UOM	Unit of Measure	Link to QTY:503 and QTY:603
CUSTPROCODE	Customer Product Code	Link to LIN03 using "BP" (buyer's part number) code
		(Continued on next page)

(Contd) Field Name	Description	Mapping Operations
UPCCODE	UPC Code	Link to PIA:202 using "UP" (UPC code) and "1" (additional information) codes
ITEMDESC	Item Description	Link to IMD:306
Summary Record (INVSUMMARY) 1 per message		
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 MOA:1402 using "128" (total amount) code
TRANSTOTAL	Transaction Totals	Link to CNT:202 using "2" (number of line items in message) code

Next step   After you correlate your application file and EDI file on paper, you can begin to create the system import and import maps.

The System Import Map

Creating 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 partner to whom you send messages and one import map for each message type.

The system uses a system import map to find the partner relationship for a message (flat file definition) and 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.

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. (Continued on next page)

(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 you create the system import map, you must define the header record for the system import map. Use the six-field key 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. The HEADER record is mandatory and occurs only once.

Procedure   Use this procedure to create the HEADER record.

Step	Action
1	Highlight the Positional File icon on the input side of the map.
2	From the Edit menu, select **Create Sub\Record**. The Positional Record Properties dialog box is displayed. **Positional Record Properties** Name | Tag | Key Field | Special | Looping Please enter the name : Please enter a short description : OK Cancel Help
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". This provides a brief explanation that allows you to differentiate it from similar records.
5	Select the **Tag** tab.

(Contd) Step	Action
6	In the Tag box, type "HDR" as the record identification code <TAG>. A record is recognized by the system by the following syntax: `<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 "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.

How to Create the HEADER Fields

Overview   To use the preferred method of building the system import key, you must 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 to define the key that the translator uses to find the partner relationship.

This table lists the fields you must create.

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per message					
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 and select **Edit Fields** from the shortcut menu to access the Positional Field Editor dialog box.
2	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 must fill in the field values in the Field Details section.
3	In the Name box, type "PARTNER_ID", which is the name of the first field in the HEADER record.
4	Click the Mandatory check box to identify it as a required field. For the system import map, each field in the key is mandatory.
5	In the Description box, type "Partner Identifier".
6	From the Data Type list, select "String."
	(Continued on next page)

(Contd) Step	Action
7	In the Start Pos box, type "4".
8	In the Min Length box, type "1" as the minimum field length.
9	In the Max Length box, type "9" as the maximum field length.
10	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.
11	Create the rest of the fields according to the record layout table on page 27. **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
12	After adding the last field, click **Auto Position** to automatically position the fields in the record. This calculates the start positioning 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 will be 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 are defining every field in the record in the sequence that each field occurs, as you are in this tutorial. The Positional Field Editor dialog box should now look like the following:
13	Click **OK** to add the fields to the HEADER record.

Defining 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.

How to define the partner key   Use this procedure to define the partner key.

Step	Action
1	Double-click the HEADER record to display the fields you just created.
2	Double-click the PARTNER_ID field to display the Field Properties dialog box.
3	Select the **Standard Rule** tab.
4	From the standard rule list, choose **Select**.
5	From the Table list, select "Partner by EDI Code" (you could also select "Partner by Alternate code," or "Partner by Partner Key") to look up the indicated partner key in the partner database and update this field. The Standard Rule tab should now look like this:
6	Click **OK** to add the standard rule to the partner key field.

How to define the 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, select "Document record" to indicate 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, select "Agency" to indicate that you are updating the agency (standard) field in the system buffer. The Standard Rule tab should now look like this:
6	Click **OK** to add the standard rule to the standard field.

How to define the version field

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**.

Step	Action
4	From the table list, select "Document record."
5	From the column list, select "Version" to indicate that you are updating the standard version field in the system buffer. The Standard Rule tab should now look like this:
6	Click **OK** to add the standard rule to the version field.

How to define the transaction field   Use this procedure to define the transaction set (message).

Step	Action
1	Double-click the transaction set 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, select "Document record." (Continued on next page)

Step	Action
5	From the column list, select "Transaction Set ID" to indicate that you are updating the transaction set field in the system buffer. The Standard Rule tab should now look like this:
6	Click **OK** to add the standard rule to the transaction set field.

How to define the test/production field

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

Step	Action
1	Double-click the test/production 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, select "Document record." (Continued on next page)

(Contd) Step	Action
5	From the column list, select "Test Mode" to indicate that you are updating the release field in the system buffer. The Standard Rule tab should now look like this:
6	Click **OK** to add the standard rule to the test or production status field.

How to Compile the Map

Overview   The Compile function generates a translation object. The map that you created (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.
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 file extension and our recommendation that you name the translation object (.TPL file) the same file name 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 will remain 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.
5	Click **Save**. The system compiles the map and generates a translation object. The Compile Error dialog box is displayed. Verify that no errors occurred, and click **OK** to exit the dialog box.

(Contd) Step	Action
6	From the Edit menu, select **Details**. The Translation Object Details dialog box is dispalyed. The date on which the translation object was compiled is automatically loaded into the Compiled on box on the Translation Object Details dialog box. Click **Cancel** to exit the dialog box.
7	From the File menu, select **Save** to save the source map. **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's Guide* for more information about 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, 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, select **Print** to access the Print Options dialog box.
2	Accept the defaults on this dialog box (all options checked) so the report will include all sections and only the activated map components. Click **OK** and the Print dialog box is displayed.
3	Select the appropriate options. Click **OK**. Click **Setup** to select a specific printer, paper orientation, or size and source of paper. The Print Setup dialog box is displayed.

Testing the Translation Object

Overview

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

Reference

Test the system import translation object (HEADER.TPL) when you test the import map (see *Testing the Translation Object* on page 3 - 137).

Creating the Import Map

How to Create the Import Map

Introduction   The first task in creating a new map is to define the format that will be 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. The input side of the map must reflect your application file. The output side of the map must reflect the EDIFACT 92.1 invoice (INVOIC) that you will send to your partner. Instead of defining the INVOIC message 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_EINV** 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 **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 **[E] Edifact** standards agency, b. **[092001] EDIFACT Release 1 (92.1)** version, c. **[INVOIC] INVOICE MESSAGE** transaction set. Click **Next**.
7	Click **Finish**. **System response** The system loads the selected transaction set and returns to the New Map Wizard - Output Format dialog box.
7	Click **Next**. **System response** The system displays the New Map Wizard.
4	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.
5	From the Edit menu, select **Details**. **System response** The system displays the Translation Object Details dialog box.
6	In the Description field, type **PET E 92.1 INVOIC Import**.

(Contd) Step	Action
7	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).
8	Click **OK** to save the translation object details.
9	From the File menu, select **Save**. **System response** The system displays the Save As dialog box. (Continued on next page)

(Contd) Step	Action
10	Verify that the File name box contains pet_einv.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_EINV.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_EINV.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.
11	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 have finished creating and saving the new Import map, you must define the Input and Output sides of the map.

Activating 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 (message) selected, and which segments and elements your company requires. We recommend that you determine which segments and elements you will use before starting this step.

When Gentran:Server for Windows generates the EDI side(s) of the map, the system includes all the groups, segments, composites, and elements that are defined by the standards agency for the message version 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.

Overview (contd)   By default, Gentran:Server for Windows displays active map components with a black font and inactive map components with a gray font (dimmed). In the example below, the DTM segment and the DTM01 element are mandatory, so they have already been activated by the system. You must also activate the DTM02 element (a conditional element).

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.

Required conditional groups, segments, and elements   This table lists the required conditional groups, segments, and elements that must be activated.

Groups		Segment	Comp.	Element
		BGM segment (already active)		BGM05 element
		DTM segment (already active)	C507	DTM02 element
		FTX segment	C108	FTX06 element FTX07 element
RFF group		RFF segment	C506	RFF02 element
		DTM:2 segment	C507	DTM:202 element
NAD group		NAD segment	C080 C059	NAD10 element NAD16 element NAD17 element
				NAD19 element NAD20 element NAD21 element
		LOC segment	C517	LOC05 element
	CTA group	CTA segment	C056	CTA03 element
				(Continued on next page)

(Contd) Groups		Segment	Comp.	Element
LIN group		LIN segment	C212	LIN01 element LIN03 element LIN04 element
		PIA segment	C212	PIA02 element PIA03 element
		IMD:2 segment	C273	IMD:206 element
		QTY:2 segment	C186	QTY:303 element
	MOA group	MOA:5 segment	C516	MOA:502 element
		CNT segment	C270	
MOA group (already active)		MOA:10 segment	C516	MOA:1002 element

Procedure　　Complete the following steps to activate the required conditional groups, segments, and elements:

Step	Action
1	Double-click the segments and groups that are listed in the previous table to open them. These are the groups, subgroups, segments, and elements you must activate. Opening all necessary groups, segments, and composites before you start to activate allows you to scroll down the map and activate the segments and elements sequentially. **Note** When you activate a group or segment, the mandatory subordinate segments and elements are automatically activated.
2	Click the **Activate** button on the Main Toolbar.
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.
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. *(Continued on next page)*

Step	Action
6	After 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.

Defining Literal Constants

Overview   The system uses literal constants 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 element(s) with which it is associated. Qualifying relationships are typically defined in outbound maps. You define a qualifying relationship using a standard rule. This process is explained later in this tutorial.

Since you are creating an import map to translate outbound data, you must use constants to define the qualifying relationships on the EDI (output) side of the map. First, determine which constants you want to use in a map, then define each to save time.

For this tutorial, you must define these constants.

Constant ID	Constant Type	Constant Value
Invoice date/time	String	3
Order number (purchase)	String	ON
Order date/time	String	4
Vendor	String	VN
RemitTo	String	RE
ShipTo	String	ST
Goods receipt place	String	4
Bill of lading number	String	BM
Despatch date/time	String	11
Invoice instruction	String	INV
Despatch quantity	String	12
Ordered quantity	String	21
Unit price	String	146
Buyer's part number	String	BP
UPC Code	String	UP
Additional Identification	String	1
		(Continued on next page)

(Contd) Constant ID	Constant Type	Constant Value
Summary	String	S
Number of line items in message	String	2
Total quantity	String	1
Total amount	String	128

Procedure   Use this procedure to create the literal constants that you will use in the import map.

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. ![Edit Constant dialog box with ID, Type, Value fields and OK, Cancel buttons]
3	In the ID box, type "Invoice Date/Time", which is the first literal constant identifier in the table. This is typically a description of the field or element in which the constant will be used.
4	From the Type list, select "String" as the category of this literal constant.
5	In the Value box, type "3" 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 in the Overview section above.
8	Click **Close** to exit the Map Constants dialog box.

How to Map the UNS01 Element

Overview　　The UNS segment contains the section control information. Both the UNS segment and the UNS01 element are mandatory. Since you are creating an outbound (import) map, you must map data to the UNS01 element. Because this data is not part of your application file, you must use a constant to map the appropriate data to this element. You have already defined all the constants that you will use for this map.

Procedure　　Use this procedure to map the mandatory data to the UNS01 element.

Step	Action
1	Double-click the UNS01 element (Section Identification) in the UNS 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 "S" constant (summary), to identify the UNS01 element as the summary section control. The Standard Rule tab should now look like this:
5	Click **OK**. The "S" code is loaded in the Section Identification element.

Promoting Groups and Repeating Segments

Overview

The Promote function extracts one iteration (instance) of a group or repeating segment. This enables you to map unique data from your application file and enter a specialized definition. Only one-to-one (no loops) or many-to-many (loops) mapping relationships are valid.

Promote is a specialized function that guarantees the integrity of your EDI structure. You must promote several groups and repeating segments. You must also enter a specialized definition for each group you promote so that you can easily differentiate between the multiple instances of a group. You decide that the naming convention that Gentran:Server uses for multiple instances of the same *segment* is sufficient and you do not need to enter specialized definitions for the segments you promote.

Note

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

How to promote the DTM segment

The DTM segment contains the date and time information for the map header and repeats up to 35 times. You must extract one iteration of the DTM segment so you can map the INVDATE field, which is in the INVBEGIN record that repeats once, to the DTM segment.

Use this procedure to promote one iteration of the DTM segment.

Step	Action
1	Right-click the DTM segment and select Promote from the shortcut menu to extract one iteration of the DTM segment. The new iteration is named DTM:22 and is located directly above the DTM segment.
2	Right-click the original DTM segment and select Deactivate from the shortcut menu to deactivate the DTM segment (which now repeats 34 times) so the system will not process it.

How to promote the RFF group and add specialized definitions

The RFF group contains reference information and repeats up to 10 times. You must extract two, singly occurring instances of the RFF group so you can map the purchase order number and date and the bill of lading number and ship date from the INVBEGIN record (which repeats once). You must add a specialized definition to each group you promote so you can easily differentiate between the two instances of the RFF group.

How to promote the RFF group and add specialized definitions (contd)

Use this procedure to promote two iterations of the RFF group and add specialized definitions.

Step	Action
1	Right-click the 0100_RFF group and select **Promote** from the shortcut menu to extract one iteration of the RFF group. The new iteration is named 0100_RFF:2 and is located directly above the 0100_RFF group.
2	Right-click the 0100_RFF group (which now repeats nine times) and select **Promote** from the shortcut menu to extract one iteration of the RFF group. The new iteration is named 0100_RFF:3 and is located directly above the 0100_RFF group.
3	Right-click the original 0100_RFF group and select **Deactivate** from the shortcut menu to deactivate the 0100_RFF group, which now repeats eight times.
4	Right-click the 0100_REF:2 group and select **Properties** to access the Group Properties dialog box.
5	In the Name box, type "PONUM_RFF", which is a specialized definition that identifies this group as containing purchase order number information. The Group Properties dialog box should now look like this: ![Group Properties dialog box] Group Properties Name │ Looping │ Please enter the name : PONUM_RFF Please enter a short description : OK　　Cancel　　Help
6	Click **OK** to save the change and exit the Group Properties dialog box.
7	Right-click the 0100_REF:3 group and select **Properties** to access the Group Properties dialog box. *(Continued on next page)*

(Contd) Step	Action
8	In the Name box, type "BOL_REF", which is a specialized definition that identifies this group as containing bill of lading information. The Group Properties dialog box should now look like this:
9	Click **OK** to save the change and exit the Group Properties dialog box.

How to promote the DTM:23 segment

The DTM:23 segment (in the PONUM_RFF group) contains the date and time information for the purchase order number reference and repeats up to five times. You must extract one iteration of the DTM:23 segment so you can map the PODATE field (which is in the INVBEGIN record that repeats once) to the DTM:23 segment.

Use this procedure to promote one iteration of the DTM:23 segment.

Step	Action
1	Right-click the DTM:23 segment in the PONUM_RFF group and select **Promote** from the shortcut menu to extract one iteration of the DTM:23 segment. The new iteration is named DTM:25 and is located directly above the DTM:23 segment.
2	Right-click the original DTM:23 segment and select **Deactivate** from the shortcut menu to deactivate the DTM:23 segment, which now repeats four times.

How to promote the DTM:24 segment

The DTM:24 segment in the BOL_RFF group contains the date and time information for the bill of lading reference and repeats up to five times. You must extract one iteration of the DTM:24 segment so you can map the SHIPDATE field (which is in the INVBEGIN record that repeats once) to the DTM:24 segment.

Use this procedure to promote one iteration of the DTM:24 segment.

Step	Action
1	Right-click the DTM:24 segment (in the BOL_RFF group) and select **Promote** from the shortcut menu to extract one iteration of the DTM:24 segment. The new iteration is named DTM:26 and is located directly above the DTM:24 segment.
2	Right-click the original DTM:24 segment and select **Deactivate** from the shortcut menu to deactivate the DTM:24 segment, which now repeats four times.

How to promote the NAD group and add specialized definitions

The NAD group contains name and address information and repeats up to 20 times. You cannot map from a non-repeating record (INVBEGIN) to a repeating group (NAD). To map vendor, shipping, and remittance information from your application file format, you must extract three single iterations of the NAD group so the links between the application fields in the INVBEGIN record and the NAD groups are valid. You must also add a specialized definition to each group you promote so you can easily differentiate between the three instances of the NAD group.

Use this procedure to promote three iterations of the NAD group and add specialized definitions.

Step	Action
1	Right-click the 0200_NAD group and select **Promote** from the shortcut menu to extract one iteration of the NAD group. The new iteration is named 0200_NAD:2 and is located directly above the 0200_NAD group.
2	Right-click the 0200_NAD group, which now repeats 19 times, and select **Promote** from the shortcut menu to extract one iteration of the NAD group. The new iteration is named 0200_NAD:3 and is located directly above the 0200_NAD group.
3	Right-click the 0200_NAD group, which now repeats 18 times, and select **Promote** from the shortcut menu to extract one iteration of the NAD group. The new iteration is named 0200_NAD:4 and is located directly above the 0200_NAD group.
4	Right-click the 0200_NAD:2 group. From the shortcut menu, select **Properties** to access the Group Properties dialog box.

(Contd) Step	Action
5	In the Name box, type "Vend_NAD", which is a specialized definition that identifies this group as containing vendor information.
6	Click **OK** to save the change and exit the Group Properties dialog box.
7	Right-click the 0200_NAD:3 group. From the shortcut menu, select **Properties** to access the Group Properties dialog box.
8	In the Name box, type "ShipTo_NAD", which is a specialized definition that identifies this group as containing shipping information.
9	Click **OK** to save the change and exit the Group Properties dialog box.
10	Right-click the 0200_NAD:4 group. From the shortcut menu, select **Properties** to access the Group Properties dialog box.
11	In the Name box, type "RemitTo_NAD", which is a specialized definition that identifies this group as containing shipping information.
12	Click **OK** to save the change and exit the Group Properties dialog box.
13	Double-click the Vend_NAD group *and* the RemitTo_NAD group to open them.
14	Click the Deactivate button on the Main Toolbar. Click the original 0200_NAD group, which now repeats 17 times.
15	In the Vend_NAD group, click the LOC:10 segment *and* the 0500_CTA:2 group.
16	In the RemitTo_NAD group, click the LOC:12 segment *and* the 0500_CTA:4 group.
17	Click the Deactivate button on the Main Toolbar again to turn Deactivation mode off.

How to promote the LOC:11 segment

The LOC:11 segment (in the ShipTo_NAD group) contains the shipping location information and repeats up to 25 times. You must extract one iteration of the LOC:11 segment so you can map the SHIPTOLOC field (which is in the INVBEGIN record that repeats once) to the LOC:11 segment.

How to promote the LOC:11 segment (contd)

Use this procedure to promote one iteration of the LOC:11 segment.

Step	Action
1	Right-click the LOC:11 segment (in the ShipTo_NAD group) and select Promote from the shortcut menu to extract one iteration of the LOC:11 segment. The new iteration is named LOC:13 and is located directly above the LOC:11 segment.
2	Right-click the original LOC:11 segment and select Deactivate from the shortcut menu to deactivate the LOC:11 segment, which now repeats 24 times.

How to promote the CTA:3 group

The CTA:3 group (in the ShipTo_NAD group) contains the contact information and repeats up to five times. You must extract one iteration of the CTA:3 group so you can map the SHIPTODEPTNUM field (which is in the INVBEGIN record that repeats once) to the CTA:3 group.

Use this procedure to promote one iteration of the CTA:3 group.

Step	Action
1	Right-click the CTA:3 group (in the ShipTo_NAD group) and select Promote from the shortcut menu to extract one iteration of the CTA:3 group. The new iteration is named CTA:5 and is located directly above the CTA:3 group.
2	Right-click the original CTA:3 group and select Deactivate from the shortcut menu to deactivate the CTA:3 group, which now repeats four times.

How to promote the PIA segment

The PIA segment (in the LIN group) contains additional product information and repeats up to 25 times. You must extract one iteration of the PIA segment so you can map your UPC code for the line item.

Use this procedure to promote one iteration of the PIA segment.

Step	Action
1	Right-click the PIA segment and select Promote from the shortcut menu to extract one iteration of the PIA segment. The new iteration is named PIA:2 and is located directly above the PIA segment.
2	Right-click the original PIA segment (that now repeats 24 times) and select Deactivate from the shortcut menu to deactivate the PIA segment.

How to promote the IMD:2 segment

The IMD:2 segment (in the LIN group) contains the Item description information and repeats up to ten times. You must extract one iteration of the IMD:2 segment so you can map the item description for the line item.

Use this procedure to promote one iteration of the IMD:2 segment.

Step	Action
1	Right-click the IMD:2 segment and select Promote from the shortcut menu to extract one iteration of the IMD:2 segment. The new iteration is named IMD:3 and is located directly above the IMD:2 segment.
2	Right-click the original IMD:2 segment (which now repeats nine times) and select Deactivate from the shortcut menu to deactivate the IMD:2 segment.

How to promote the QTY:2 segment

The QTY segment contains the quantity information and repeats up to five times. You must extract two iterations of the QTY:2 segment so you can store the quantity shipped and the quantity ordered.

Use this procedure to promote two iterations of the QTY:2 segment.

Step	Action
1	Right-click the QTY:2 segment and select Promote from the shortcut menu to extract one iteration of the QTY:2 segment. The new iteration is named QTY:5 and is located directly above the QTY:2 segment.
2	Right-click the original QTY:2 segment, which now repeats four times, and select Promote from the shortcut menu to extract one iteration of the QTY:2 segment. The new iteration is named QTY:6 and is located directly above the QTY:2 segment.
3	Right-click the original QTY:2 segment and select Deactivate from the shortcut menu to deactivate the QTY:2 segment, which now repeats three times.

How to promote the MOA group

The MOA group in the detail section of the map (LIN group) contains the line item unit price information and repeats up to five times. You must extract one iteration of the MOA group so you can map the unit price to the MOA group. Since you are only extracting one instance of the MOA group and then deactivating the original group, you do not need to add a specialized definition to the new iteration of the group.

Use this procedure to promote one iteration of the MOA group.

Step	Action
1	Right-click the 2300_MOA group (in the LIN group) and select Promote from the shortcut menu to extract one iteration of the MOA group. The new iteration is named 2300_MOA:2 and is located directly above the 2300_MOA group.
2	Right-click the original 2300_MOA group and select Deactivate from the shortcut menu to deactivate the 2300_MOA group, which now repeats four times.

How to promote the CNT segment

The CNT segment contains the control total information and repeats up to 10 times. You must extract two iterations of the CNT segment so you can store the total number of line items and the total quantity invoiced.

Use this procedure to promote two iterations of the CNT segment.

Step	Action
1	Right-click the CNT segment and select Promote from the shortcut menu to extract one iteration of the CNT segment. The new iteration is named CNT:2 and is located directly above the CNT segment.
2	Right-click the original CNT segment (that now repeats nine times) and select Promote from the shortcut menu to extract one iteration of the CNT segment. The new iteration is named CNT:3 and is located directly above the CNT segment.
3	Right-click the original CNT segment and select Deactivate from the shortcut menu to deactivate the CNT segment, which now repeats eight times.

How to promote the MOA group

The MOA group in the summary section of the map contains the total monetary amount information for the and repeats up to 100 times. You must extract one iteration of the MOA group so you will be able to map the TOTALAMOUNT field (which is in the INVSUMMARY record that repeats once), to the MOA group. Since you are only extracting one instance of the MOA group (and will then deactivate the original group), you do not need to add a specialized definition to the new iteration of the group.

How to promote the MOA group (contd)

Use this procedure to promote one iteration of the MOA group.

Step	Action
1	Right-click the 4500_MOA group and select Promote from the shortcut menu to extract one iteration of the MOA group. The new iteration is named 4500_MOA:2 and is located directly above the 4500_MOA group.
2	Right-click the original 4500_MOA group and select Deactivate from the shortcut menu to deactivate the 4500_MOA group, which now repeats 99 times.

How to Verify EDI Delimiters

Overview   If you are using an EDI standard that contains composite elements or sub-elements, you must verify that Gentran:Server for Windows is specifying the correct 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 sub-elements, 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 INVOIC INVOICE MESSAGE file icon and from the shortcut menu, select **Properties**. **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 that the Specify defaults check box is selected.
4	Verify the required delimiters for the EDIFACT standard. If the delimiters differ from the defaults in the table below, type either the character or the hexadecimal value in the correct box. The default delimiters for EDIFACT are the following: table The tag delimiter is not used for EDIFACT.
5	Click **OK** to exit the File Properties dialog box.

Inner table for step 4:

Box	EDIFACT Default Character	EDIFACT Default Hex Value
Tag Delimiter		
Element Delimiter	+ (plus)	0x2B
Segment Delimiter	' (single quote)	0x27
Sub Element Delimiter	: (colon)	0x3A
Release Character	? (question mark)	0x3F
Decimal Separator	. (period)	0x2E

Next step   The next step is to define your application file layout.

Defining the Rest of Your Application File Format

Overview

Introduction   You have already defined the system import map that you will 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 message that you will send to your partner.

Before you define your application file format, obtain a layout of the necessary records, fields, and groups. 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   You can copy the system import map header record to the import map. The copied record contains subordinate fields, which will 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.
2	Highlight the HEADER record.
3	Click Copy on the Main Toolbar.
4	Open the PET_EINV.MAP import map.
5	Highlight the INPUT positional file icon.
6	Click Paste on the Main Toolbar to paste the contents of the clipboard. The HEADER record is pasted as a subordinate map component to the INPUT positional file icon.

How to Remove the HEADER Standard Rules

Overview   The HEADER record that you just copied from the system import map contains five fields. The first field (PARTNER_ID) had 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) had 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 standards 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>** to remove the standard rule from the field.
4	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 must 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, and ship date. 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 "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. Use the underscore (_) to separate words.
3	In the Desc box, type "Beginning Record". This description provides 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>. A record is recognized by the following syntax: `<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.
6	In the Position box, accept the default of "1" for the starting column position of the tag in the data record. *(Continued on next page)*

(Contd) Step	Action
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).
9	Click **OK** to create the INVBEGIN record.

How to Create the INVCOMMENT Group

Overview   The INVCommentGroup (comment information) groups all invoice comment information, including the reference number and the message text. The INVCommentGroup occurs up to 10 times 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 and select **Insert** from the Edit menu. From the submenu, select **Group**. **System response** The system displays the Group Properties dialog box.
2	In the Name box, type "INVCommentGroup" to indicate that this group will contain 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 description provides 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. In the Maximum Usage box, type "10" to indicate that this group of records will occur up to ten times.
6	Click **OK** to create the INVCommentGroup.

Creating the INVCommentGroup Subordinate Records

Overview   In this tutorial, the INVCommentGroup contains two subordinate records:

◗ INVCOMMENT1

◗ INVCOMMENT2

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

How to create 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).

Use this procedure to create the INVCOMMENT1 record.

Step	Action
1	Highlight the INVCommentGroup. From the Edit menu, select **Create Sub**. From the submenu, select **Record**. 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 "Invoice Comment Record 1".
4	Select the **Tag** tab.
5	In the Tag box, type "CM1" as the record identification code <TAG>.
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. Accept the default of "1" in the Max Usage box so this 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 INVCOMMENT1 record.

How to create the INVCOMMENT2 record

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

Use this procedure to create the INVCOMMENT2 record.

Step	Action
1	Highlight theINVCOMMENT1 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 "INVCOMMENT2" as the record identification code.
3	In the Desc box, type "Invoice Comment Record 2".
4	Select the **Tag** tab.
	(Continued on next page)

(Contd) Step	Action
5	In the Tag box, type "CM2" as the record identification code <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.
8	In the Min Usage box, type "1" to make the record mandatory. Accept the default of "1" in the Max Usage box so this record only occurs once (not a looping structure). The Positional Record Properties dialog box should now look like this: **Positional Record Properties** Name | Tag | Key Field | Special | Looping Min Usage 1 ⦿ Normal ○ Loop Start Max Usage 1 ○ Loop End OK Cancel Help
9	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, and product code. 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 record description.
4	Select the **Tag** tab.
5	In the Tag box, type "DET" as the record identification code <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.
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 must 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 record description.
4	Select the **Tag** tab.
5	In the Tag box, type "SUM" as the record identification code <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.
8	In the Min Usage box, type "1" to make the record mandatory.
9	Accept the default of "1" in the Max Usage box.
10	Click **OK** to create the INVSUMMARY record.

Next step Now that you have created the record structure of the application file, you can define the application fields. The application side of the map (input) should now look like this:


```
INPUT - M - 1
 ├─ HEADER - M - 1 Header Record
 ├─ INVBEGIN - M - 1 Beginning Record
 ├─ INVCommentGroup - M - 10 Comment Group
 │    ├─ INVCOMMENT1 - M - 1 Invoice Comment Record 1
 │    └─ INVCOMMENT2 - M - 1 Invoice Comment Record 2
 ├─ INVDETAIL - M - 200000 Detail Record
 └─ INVSUMMARY - M - 1 Summary Record
```

Creating the Application Fields

Overview

Introduction   Each record you create will contain a group of logically related application fields. These fields define the data structure and content that your system uses to process the invoice.

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

Application fields   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 message					
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 message					
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

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
VENDADD2		Vendor Address Line2	String	0	35
VENDCITY		Vendor City	String	0	35
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	35
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	35
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
Comment Group (INVCommentGroup) 10 per message					
Comment Record 1 (INVCOMMENT1) 1 per message					
BILLTOACCTCM1	X	Bill To Acct Number	String	1	9
DOCTYPCM1	X	Document Type	String	1	6
					(Continued on next page)

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
INVNUMCM1	X	Invoice Number	String	1	10
REFNUM	X	Reference Number	String	1	30
Comment Record 2 (INVCOMMENT2) 1 per message					
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 message					
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
Summary Record (INVSUMMARY) 1 per message					
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 Define the INVBEGIN Application Fields

Overview   You created the fields for the system import map HEADER record, copied the fields to the import map, and removed the standard rules. The INVBEGIN record is the first record for which you must define fields.

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

Step	Action
1	Right-click the INVBEGIN application record and select **Edit Fields** from the shortcut menu to access the Positional Field Editor dialog box.
2	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 must fill in the values in the Field Details section.
3	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 fields that occur in multiple records with a suffix that identifies the record that contains it. For example, "BEG" for the Beginning record.
4	Select the Mandatory check box to identify it as a required field.
5	In the Description box, type "Bill To Account Number". This description provides a brief explanation of the field that allows you to differentiate it from similar fields.
6	From the Data Type list, select "String" as the field type to specify that this field must contain alphanumeric data. **Note** To indicate the exact position of the field in the record, type the starting field position in the Start Pos box. Specify field start positions if, for example, you will only be using a few fields and you want them positioned exactly in the record. **Reference** In this tutorial, you will 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 about the Auto Position function.
7	If the Data Type is "Number" or "Dt/Tm", select the appropriate format from the Format list. (Continued on next page)

(Contd) Step	Action
8	If, according to the table starting on page 72, the field's minimum length is "0", then accept that default in the Min Length box. Otherwise type "1".
9	In the Max Length box, type "9" as the maximum field length.
10	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.
11	Create the rest of the fields according to the record layout table starting on page 72. **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	Click **Auto Position** to automatically position the fields in the record. This 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 will be 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 this tutorial. The Positional Field Editor dialog box should now look like this: Only the last 11 fields in the record are visible in the Fields list.

How to Define 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 and select **Edit Fields** from the shortcut menu to access the Positional Field Editor dialog box.
2	If the field you are creating is the first field in the record, click **New**.
3	In the Name box, type "BILLTOACCTCM1", which is the name of the first field in the INVOCMMENT1 record.
4	Click the Mandatory check box to identify it as a required field.
5	In the Description box, type "Bill To Account Number".
6	From the Data Type list, select "String" as the field type.
7	If the Data Type is "Number" or "Dt/Tm," select the appropriate format from the Format list.
8	If, according to *Application fields* on page 3 - 72, the field's minimum length is "0", then accept that default in the Min Length box. Otherwise type "1".
9	In the Max Length box, type "9" as the maximum field length.
10	To add the field and to create the next field positioned after it, press **Enter**. A new field with blank fields ready for you to identify is positioned below the field.
11	Create the rest of the fields according to *Application fields* on page 3 - 72. **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
12	Click **Auto Position**. Click **Yes** to acknowledge the warning message that fields will be sequenced in order. The Positional Field Editor dialog box should now look like the following:

How to Define the INVCOMMENT2 Application Fields

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

Step	Action
1	Right-click the INVCOMMENT2 application record and select **Edit Fields** from the shortcut menu to access the Positional Field Editor dialog box.
2	If the field you are creating is the first field in the record, click **New**.
3	In the Name box, type "BILLTOACCTCM2", which is the name of the first field in the INVCOMMENT2 record.
4	Click the Mandatory check box to identify it as a required field.
5	In the Description box, type "Bill To Account Number" as the field description.
6	From the Data Type list, select "String" as the field type.
7	If the Data Type is "Number" or "Dt/Tm", select the appropriate format from the Format list.
8	If, according to *Application fields* on page 3 - 72, the field's minimum length is "0", then accept that default in the Min Length box. Otherwise type "1".
9	In the Max Length box, type "9" as the maximum field length.
10	To add the field and to create the next field positioned after it, press **Enter**.
11	Create the rest of the fields according to *Application fields* on page 3 - 72. **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.

(Contd) Step	Action
12	Click **Auto Position**. Click **Yes** to acknowledge the warning message that fields will be sequenced in order. The Positional Field Editor dialog box should now look like this:

How to Define the INVDETAIL Application Fields

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

Step	Action
1	Right-click the INVDETAIL application record and select **Edit Fields** from the shortcut menu to access the Positional Field Editor dialog box.
2	If the field you are creating is the first field in the record, click **New**.
3	In the Name box, type "BILLTOACCTDET", which is the name of the first field in the INVDETAIL record.
4	Click the Mandatory check box to identify it as a required field.
5	In the Description box, type "Bill To Account Number" as the field description.
6	From the Data Type list, select "String" as the field type.
7	If the Data Type is "Number" or "Dt/Tm", select the appropriate format from the Format list.
8	If, according to the table starting on page 72, the field's minimum length is "0", then accept that default in the Min Length box. Otherwise type "1".
9	In the Max Length box, type "9" as the maximum field length.
10	To add the field and to create the next field positioned after it, press **Enter**.
11	Create the rest of the fields according to *Application fields* on page 3 - 72. **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
12	Click **Auto Position**. Click **Yes** to acknowledge the warning message that fields will be sequenced in order. The Positional Field Editor dialog box should now look like this:

How to Define the INVSUMMARY Application Fields

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

Step	Action
1	Right-click the INVSUMMARY application record and select **Edit Fields** from the shortcut menu to access the Positional Field Editor dialog box.
2	If the field you are creating is the first field in the record, click **New**.
3	In the Name box, type "BILLTOACCTSUM", which is the name of the first field in the INVSUMMARY record.
4	Click the Mandatory check box to identify it as a required field.
5	In the Description box, type "Bill To Account Number" as the field description.
6	From the Data Type list, select "String" as the field type.
7	If the Data Type is "Number" or "Dt/Tm", select the appropriate format from the Format list.
8	If, according to t*Application fields* on page 3 - 72, the field's minimum length is "0", then accept that default in the Min Length box. Otherwise type "1".
9	In the Max Length box, type "9" as the maximum field length.
10	To add the field and to create the next field positioned after it, press **Enter**.
11	Create the rest of the fields according to *Application fields* on page 3 - 72. **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
12	Click **Auto Position**. Click **Yes** to acknowledge the warning message that fields will be sequenced in order. The Positional Field Editor dialog box should now look like the following:

Next step   After defining your application file, you must map the necessary data from the EDI file to your application fields.

Mapping Your Data

Overview

Introduction	This section examines each application field sequentially and describes the mapping operations that you must 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 part of your application file, it must be defined in the import map. 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.

INVBEGIN record	This section describes how to map from each field in the INVBEGIN (beginning) 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 and ship date. The beginning record is mandatory and occurs once for each invoice.

BILLTOACCTBEG field	The BILLTOACCTBEG field is part of the application file format key, but does not contain data that the system uses to generate the invoice. You will 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 uses to generate the invoice. You will not perform any mapping operations on this field.

INVNUMBEG Field

Overview
Use the Link function to map the INVNUMBEG field directly to the BGM05 (document/Message Number) element. 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.

You can also link the INVNUMBEG field directly to the UNS01 element (Section Identification). This is a mandatory element in the mandatory UNS segment (Section Control). For Gentran:Server for Windows to create a mandatory segment in an outbound map, that segment must have at least one field mapped to it. This direct link is overwritten by mapping the constant value ("S") that you have already mapped to the UNS01 element.

We recommend that you set up a Document Name for each map to make the message identification easier when you use the document browsers.

To set up a Document Name, you must select an element or field in the map that contains data to distinguish the messages translated by this map.

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

How to map the invoice number
To map the invoice number, link the application field INVNUMBEG in the INVBEGIN record to the BGM05 element in the BGM record.

Then, link the INVNUMBEG field to the UNS01 element (Section Identification).

How to set up the document name
Use this procedure to set up the Document Name for this invoice.

Step	Action
1	Double-click the BGM05 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, select "Document record."
	(Continued on next page)

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

INVDATE Field

Overview

On the EDI side of the map, the date information is contained in the DTM:22 segment. You must change the properties of the DTM:2202 element (Date/Time/Period) to reflect the minimum and maximum length, the type, and the format that your partner will send you.

Use the Link function to map the INVDATE field directly to the DTM:2202 (Date/Time/Period) element. If the DTM:2201 (Date/Time/Period Qualifier) is "3", the DTM:22 segment contains the invoice date/time. You must map a constant to the Date/Time/Period Qualifier and establish a qualifying relationship with the Date/Time/Period. This enables you to specify that the Date/Time/Period element contains the invoice date/time.

How to change the properties of the DTM:2202 element

You informed your partner that you will send six digits for the DTM:2202 element (Date/Time/Period) and these six digits will be a Dt/Tm data type in the YYMMDD format (two-digit year, two-digit month, and two-digit day). You must change the minimum and maximum length, the type, and the format of the DTM:2202 element.

Use this procedure to change the length of the DTM2202 element.

Step	Action
1	Double-click the DTM:2202 element to display the Element Properties dialog box.
2	Select the **Validation** tab.
3	In the Minimum box, type "6" as the minimum length.
4	In the Maximum box, type "6" as the maximum length.
5	From the Type list, select "Date/Time" to indicate that this is a date/time element. The system may prompt you with a message that changing the field's data type will invalidate its format. If so, click **Yes** to acknowledge his message and proceed. *(Continued on next page)*

(Contd) Step	Action
6	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. The Validation tab should now look like this:
7	Click **OK** to exit the Element Properties dialog box.

How to set the date/time/period qualifier

You must map a constant to the Date/Time/Period Qualifier (DTM:2201 element) and define a qualifying relationship with the Date/Time/Period (DTM:2202 element).

You have already defined all the constants that you will use for this map. Now map the appropriate constant to the Date/Time/Period Qualifier element.

Use this procedure to set the Date/Time/Period Qualifier for the DTM:22 segment.

Step	Action
1	Double-click the DTM:2201 element (Date/Time/Period Qualifier) in the DTM:22 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 "3" (invoice date/time) to identify the DTM:2202 element as containing the invoice date and time. (Continued on next page)

(Contd) Step	Action
5	From the Qualifies list, select "2380:2 DATE/TIME/PERIOD," to set up a qualifying relationship with the DTM:2202 element. **Note** The Qualifies list contains only *active* elements in the same segment as the qualifying element. The Standard Rule tab should now look like this:
6	Click **OK**. The "3" code is loaded in the Date/TIme/Qualifier element. The qualifying relationship between the DTM:2201 and the DTM:2202 elements is established.

How to map the invoice date

To map the invoice date, link the application field INVDATE in the INVBEGIN record to the DTM:2202 element in the DTM:22 segment.

PONUM Field

Overview   Use the Link function to map the PONUM field directly to the RFF:1102 (Reference Number) field. If the RFF:1101 (Reference Qualifier) is "ON," the RFF:11 segment contains the order number (purchase). You must map a constant to the Reference Qualifier and establish a qualifying relationship with the Reference Number. This enables you to specify that the Reference Number element contains the order number (purchase).

Recommendation
We recommend that you set up a Document Reference for each map to make message identification easier.

To set up a Document Reference, you must select an element or field in the map that contains data to distinguish messages 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 message will contain the data from that mapped field.

How to set the reference qualifier   You must map a constant to the Reference Qualifier (RFF:1101 element) and define a qualifying relationship with the Reference Number (RFF:1102 element).

You have already defined all the constants that you will use for this map. Now map the appropriate constant to the Reference Qualifier element.

Use this procedure to set the Reference Qualifier for the RFF:11 segment.

Step	Action
1	Double-click the RFF:1101 element (Reference Qualifier) in the RFF:11 segment (PONUM_RFF 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 "ON" constant to identify the RFF:1102 element as containing the order number (purchase).
5	From the Qualifies list, select the "1154:2 REFERENCE NUMBER" to set up a qualifying relationship with the RFF:1102 element.
6	Click **OK** and the "ON" code is loaded in the Reference Qualifier element. The qualifying relationship between the RFF:1101 and RFF:1102 elements is also established.

How to map the purchase order number

To map the purchase order number, link the application field PONUM in the INVBEGIN record to the RFF:1102 element in the RFF:11 segment (PONUM_RFF group).

How to set up the document reference

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

Step	Action
1	Double-click the RFF:1102 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, select "Document record."
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. The Standard Rule tab should now look like this:
6	Click **OK** to set up the document reference.

PODATE Field

Overview

On the EDI side of the map, the purchase order date information is contained in the DTM:25 segment (in the PONUM_RFF group). You must change the properties of the DTM:2502 element (Date/Time/Period) to reflect the minimum and maximum length, type, and format that your partner will send you.

Use the Link function you to map the PODATE field directly to the DTM:2502 (Date/Time/Period) element. If the DTM:2501 (Date/Time/Period Qualifier) is "4", the DTM:25 segment contains the order date/time. You must map a constant to the Date/Time/Period Qualifier and establish a qualifying relationship with the Date/Time/Period. This enables you to specify that the Date/Time/Period element contains the order date/time.

How to change the properties of the DTM:2502 element

You have informed your partner that you will send six digits for the DTM:2502 element (Date/Time/Period) and that these six digits will be a Dt/Tm data type in the YYMMDD format (two-digit year, two-digit month, and two-digit day). You must change the minimum and maximum length, type, and format of the DTM:2502 element.

Use this procedure to change the length of the DTM:2502 element.

Step	Action
1	Double-click the DTM:2502 element to display the Element Properties dialog box.
2	Select the **Validation** tab.
3	In the Minimum box, type "6" as the minimum length.
4	In the Maximum box, type "6" as the maximum length.
5	From the Type list, select "Date/Time." Click **Yes** to acknowledge this message and proceed. (Continued on next page)

(Contd) Step	Action
6	From the Format list, select "YYMMDD." The Validation tab should now look like this:
7	Click **OK** to exit the Element Properties dialog box.

How to set the date/time/period qualifier

You must map a constant to the Date/Time/Period Qualifier (DTM:2501 element) and define a qualifying relationship with the Date/Time/Period (DTM:2502 element).

You have already defined all the constants that you will use for this map. Now you can map the appropriate constant to the Date/Time/Period Qualifier element.

Use this procedure to set the Date/Time/Period Qualifier for the DTM:25 segment.

Step	Action
1	Double-click the DTM:2501 element (Date/Time/Period Qualifier) in the DTM:25 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 "4" (order date/time) to identify the DTM:2502 element as containing the order date and time. (Continued on next page)

(Contd) Step	Action
5	From the Qualifies list, select "2380:5 DATE/TIME/PERIOD" to set up a qualifying relationship with the DTM:2502 element.
6	Click **OK**. The "4" code is loaded in the Date/Time/Period Qualifier element and the qualifying relationship between the DTM:2501 and DTM:2502 elements is established.

How to map the purchase order date

Use the Link function to map the PODATE field directly to the DTM:2502 (Date/Time/Period) element.

To map the purchase order date, link the application field PODATE in the INVBEGIN record to the DTM:2502 element in the DTM:25 segment (in the PONUM_RFF group).

Mapping the Shipping and Billing Information

Overview

Introduction

The NAD group contains vendor, shipping, and billing name and address information. It repeats multiple times. Your application file contains Vendor, Ship To, and Remit To information that you must map to the NAD 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 NAD group, because the INVBEGIN record does not repeat.

To map vendor, shipping, and remittance information from your application file format, you created three single iterations of the NAD group using the promote function, so that the links between the application fields in the INVBEGIN record and the NAD groups are valid. The Promote function extracts the one iteration (instance) of a group (or repeating segment). This enables you to map unique data from your application file to each NAD group (Vend_NAD, ShipTo_NAD, and RemitTo_NAD).

VENDNAME Field

Overview

You can map a constant to the NAD segment Party Qualifier for each iteration of the NAD group. If the Party Qualifier is "VN," that iteration of the NAD group contains shipping information. You must map the Vendor name and address fields in your application file format to the Vend_NAD group. If the Party Qualifier is "ST," that iteration of the NAD group contains shipping information. You must map the Ship To name and address fields in your application file format to the ShipTo_NAD group. If the Party Qualifier is "RE", that iteration of the NAD group contains remittance information. You must map the remittance name and address fields in your application file format to the RemitTo_NAD group.

How to set the party qualifier

You must set the Party Qualifier for the Vend_NAD, ShipTo_NAD, and RemitTo_NAD groups, so the system can distinguish the type of name and address information each NAD group should contain.

You have already defined all the constants that you will use for this map. Now you can map the appropriate constant to each Party Qualifier element.

Step	Action
1	Double-click the NAD:310 element (Party Qualifier) in the NAD: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 "VN" to identify this iteration of the NAD group as containing vendor name and address information.
5	Click **OK**. The "VN" code is loaded in the Party Qualifier element.

How to set the party qualifier for the ShipTo_NAD segment

Use this procedure to set the Party Qualifier for the ShipTo_NAD group.

Step	Action
1	Double-click the NAD:401 element (Party Qualifier) in the NAD: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 "ST" to identify this iteration of the NAD group as containing shipping name and address information.
5	Click **OK**. The "ST" code is loaded in the Party Qualifier element.

How to set the party qualifier for the RemitTo_NAD group

Use this procedure to set the Party Qualifier for the RemitTo_NAD group.

Step	Action
1	Double-click the NAD:501 element (Party Qualifier) in the NAD: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 "RE" to identify this iteration of the NAD group as containing vendor name and address information.
5	Click **OK**. The "RE" code is loaded in the Party Qualifier element.

How to link the NAD group

Link the elements in your application fields to the appropriate elements in the NAD 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	Link the field to the
VENDNAME	NAD:310 element in the NAD:3 segment (Vend_NAD group).
VENADD1	NAD:316 element in the NAD:3 segment (Vend_NAD group).
VENADD2	NAD:317 element in the NAD:3 segment (Vend_NAD group).
VENDCITY	NAD:319 element in the NAD:3 segment (Vend_NAD group).
VENDSTATE	NAD:320 element in the NAD:3 segment (Vend_NAD group).
VENDPOSTAL CODE	NAD:321 element in the NAD:3 segment (Vend_NAD group).
SHIPTONAME	NAD:410 element in the NAD:4 segment (ShipTo_NAD group).
SHIPTOADD1	NAD:416 element in the NAD:4 segment (ShipTo_NAD group).
SHIPTOADD2	NAD:417 element in the NAD:4 segment (ShipTo_NAD group).
SHIPTOCITY	NAD:419 element in the NAD:4 segment (ShipTo_NAD group).
SHIPTOSTATE	NAD:420 element in the NAD:4 segment (ShipTo_NAD group).
SHIPTOPOSTAL CODE	NAD:421 element in the NAD:4 segment (ShipTo_NAD group).

SHIPTOLOC Field

Overview

The SHIPTOLOC (ship to location) contains additional reference information about the Ship To address. This field is in the INVBEGIN record that occurs once. Map this field to the LOC:1305 element (place/location) in the LOC:13 segment in the ShipTo_NAD group.

You must also map a constant to the Place Location Qualifier (LOC:1301 element) and define a qualifying relationship with the Place/Location (LOC:1305 element). If the Place/ Location Qualifier is "4" (goods receipt place), that iteration of the LOC:13 segment contains the ship to location.

How to set the place/location qualifier

You have already defined all constants that you will use for this map. Now map the appropriate constant to each Place/Location Qualifier element.

Use this procedure to set the Place/Location Qualifier for the LOC:13 segment.

Step	Action
1	Double-click the LOC:1301 element (Place/Location Qualifier) in the LOC:13 segment in the ShipTo_NAD 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 *second* "4" constant in the list to identify this iteration of the LOC:13 segment as containing the ship to department number.
5	From the Qualifies list, select "3224:5 PLACE/LOCATION" to set up a qualifying relationship with the Place/Location element.
6	Click **OK**. The "4" code is loaded in the Place/Location Qualifier element and the qualifying relationship between the LOC:1301 and LOC:1305 elements is established.

How to map the ship to location field

After mapping the constant to the Place/Location Qualifier element and establishing the appropriate qualifying relationship, map the SHIPTOLOC field.

To map the SHIPTOLOC, link the application field SHIPTOLOC in the INVBEGIN record to the LOC:1305 element in the LOC:13 segment (ShipTo_NAD group).

How to map the SHIPTODEPTNUM field

The SHIPTODEPTNUM (ship to department number) contains additional reference information about the Ship To address. This field is in the INVBEGIN record that occurs once. Map this field to the CTA:603 element (department or employee) in the CTA:6 segment in the CTA:5 group in the ShipTo_NAD group.

To map the SHIPTODEPTNUM, link the application field SHIPTODEPTNUM in the INVBEGIN record to the CTA:603 element in the CTA:6 segment in the CTA:5 group (ShipTo_NAD 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	Link the field to the
REMITTONAME	NAD:510 element in the NAD:5 segment (RemitTo_NAD group).
REMITTOADD1	NAD:516 element in the NAD:5 segment (RemitTo_NAD group).
REMITTOADD2	NAD:517 element in the NAD:5 segment (RemitTo_NAD group).
REMITTOCITY	NAD:519 element in the NAD:5 segment (RemitTo_NAD group).
REMITTOSTATE	NAD:520 element in the NAD:5 segment (RemitTo_NAD group).
REMITTOPOSTALCODE	NAD:521 element in the NAD:5 segment (RemitTo_NAD group).

BOLNUM Field

Overview

Use the Link function to map the BOLNUM field directly to the RFF:1202 (Reference Number) field. If the RFF:1201 (Reference Qualifier) is "BM," the RFF:12 segment contains the bill of lading number. You must map a constant to the Reference Qualifier and establish a qualifying relationship with the Reference Number. This enables you to specify that the Reference Number element contains the bill of lading number.

How to set the reference qualifier

You must map a constant to the Reference Qualifier (RFF:1201 element) and define a qualifying relationship with the Reference Number (RFF:1202 element).

You have already defined all the constants that you will use for this map. Now map the appropriate constant to the Reference Qualifier element.

Use this procedure to set the Reference Qualifier for the REF:12 segment.

Step	Action
1	Double-click the RFF:1201 element (Reference Qualifier) in the RFF:12 segment (BOL_RFF 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 "BM" constant (bill of lading number) to identify the RFF:1202 element as containing the bill of lading number.
5	From the Qualifies list, select "1154:3 REFERENCE NUMBER" to set up a qualifying relationship with the RFF:1202 element.
6	Click **OK**. The "BM" code is loaded in the Reference Qualifier element and the qualifying relationship between the RFF:1201 and RFF:1202 elements is established.

How to map the bill of lading number

To map the BOLNUM (bill of lading number), link the application field BOLNUM in the INVBEGIN record to the RFF:1202 element in the RFF:12 segment (BOL_RFF group).

SHIPDATE Field

Overview

On the EDI side of the map, the purchase order date information is contained in the DTM:26 segment (in the BOL_RFF group). You must change the properties of the DTM:2602 element (Date/Time/Period) to reflect the minimum and maximum length, type, and format that your partner will send you.

Use the Link function to map the SHIPDATE field directly to the DTM:2602 (Date/Time/Period) element. If the DTM:2601 (Date/Time/Period Qualifier) is "11", the DTM:26 segment contains the despatch date/time. You must map a constant to the Date/Time/Period Qualifier and establish a qualifying relationship with the Date/Time/Period. This enables you to specify that the Date/Time/Period element contains the despatch date/time.

How to change the properties of the DTM:2602 element

You have informed your partner that you will send six digits for the DTM:2602 element (Date/Time/Period), and these six digits will be a Dt/Tm data type in the YYMMDD format (two-digit year, two-digit month, and two-digit day). You must change the minimum and maximum length, type, and format of the DTM:2602 element.

Use this procedure to change the length of the DTM:2602 element.

Step	Action
1	Double-click the DTM:2602 element to display the Element Properties dialog box.
2	Select the **Validation** tab.
3	In the Minimum box, type "6" as the minimum length.
4	In the Maximum box, type "6" as the maximum length.
5	From the Type list, select "Date/Time." *(Continued on next page)*

(Contd) Step	Action
6	From the Format list, select "YYMMDD." The Validation tab should now look like this:
7	Click **OK** to exit the Element Properties dialog box.

How to set the date/time/period qualifier

You must map a constant to the Date/Time/Period Qualifier (DTM:2601 element) and define a qualifying relationship with the Date/Time/Period (DTM:2602 element).

You have already defined all the constants that you will use for this map. Now you map the appropriate constant to the Date/Time/Period Qualifier element.

Use this procedure to set the Date/Time/Period Qualifier for the DTM:26 segment.

Step	Action
1	Double-click the DTM:2601 element (Date/Time/Period Qualifier) in the DTM:26 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 "11" (dispatch date/time) to identify the DTM:2602 element as containing the ship date and time. (Continued on next page)

(Contd) Step	Action
5	From the Qualifies list, select "2380:6 DATE/TIME/PERIOD" to set up a qualifying relationship with the DTM:2602 element.
6	Click **OK**. The "11" code is loaded in the Date/Time/Period Qualifier element and the qualifying relationship between the DTM:2601 and DTM:2602 elements is established.

How to map the purchase order date

Use the Link function to map the SHIPDATE field directly to the DTM:2602 (Date/Time/Period) element.

To map the purchase order date, link the application field SHIPDATE in the INVBEGIN record to the DTM:2602 element in the DTM:26 segment (in the BOL_RFF group).

The INVCommentGroup: INVCOMMENT1 Record

Overview

Introduction   The INVCommentGroup (comment information) groups all the invoice comment information, including the reference number and the message text. The INVCommentGroup occurs up to 10 times per invoice translated. The information within the INVCommentGroup is further segregated by record.

Mapping Fields in the INVCOMMENT1 Record

Overview

This section describes how 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).

BILLTOACCTC M1 field

The BILLTOACCTCM1 field is part of the application file format key but does not contain data that the system uses to generate the invoice. You will not perform any mapping operations on this field.

DOCTYPCM1 field

The DOCTYPCM1 field is part of the application file format key but does not contain data that the system uses to generate the invoice. You will not perform any mapping operations on this field.

INVNUMCM1 field

The INVNUMCM1 field is part of the application file format key but does not contain data that the system uses to generate the invoice. You will not perform any mapping operations on this field.

How to Map the REFNUM Field

Overview
The FTX segment contains free form or coded text information. It repeats up to 10 times. The REFNUM field can be directly mapped to the FTX06 element in the FTX segment, but you must map a constant to the Text Subject Qualifier (FTX01 element) and define a qualifying relationship with the Free Text (FTX06) element to indicate that the FTX06 contains the reference number.

How to set the text subject qualifier
You have already defined all the constants that you will use for this map. Now map the appropriate constant to the Text Subject Qualifier element.

Use this procedure to set the Text Subject Qualifier for the FTX segment.

Step	Action
1	Double-click the FTX01 element (Text Subject Qualifier) in the FTX 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 "INV" constant to identify this iteration of the FTX segment as containing the reference number (invoice instruction). This number is in addition to the primary order number.
5	From the Qualifies list, select "4440 FREE TEXT" to set up a qualifying relationship with the Free Text element.
6	Click **OK**. The "INV" code is loaded in the Reference Number Qualifier element and the qualifying relationship between the FTX01 and FTX06 elements is established.

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

To map the shipping reference information, link the application field REFNUM in the INVCOMMENT1 record to the FTX06 element in the REF segment.

The INVCommentGroup: INVCOMMENT2 Record

Overview

Introduction

This section describes how to map from each field in the INVCOMMENT2 (first comment) record. The INVCOMMENT2 record contains message text. The second comment record is mandatory and occurs once per invoice translated (once for each comment line).

BILLTOACCTC M2 field

The BILLTOACCTCM2 field is part of the application file format key but does not contain data that the system uses to generate the invoice. You will not perform any mapping operations on this field.

DOCTYPCM2 field

The DOCTYPCM2 field is part of the application file format key but does not contain data that the system uses to generate the invoice. You will not perform any mapping operations on this field.

INVNUMCM2 field

The INVNUMCM2 field is part of the application file format key but does not contain data that the system uses to generate the invoice. You will not perform any mapping operations on this field.

How to map the MESSAGETEXT field

To map the message text, link the application field MESSAGETEXT in the INVCOMMENT2 record to the FTX07 element in the FTX segment.

The INVCommentGroup: INVDETAIL Record

Overview

Introduction

This section describes how 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, and product code. The detail record is mandatory and occurs up to 200,000 times per invoice translated (once for each line item invoiced).

BILLTOACCTDE T field

The BILLTOACCTDET field is part of the application file format key but does not contain data that the system needs to use to generate the invoice. You will not perform any mapping operations on this field.

DOCTYPDET field

The DOCTYPDET field is part of the application file format key but does not contain data that the system needs to use to generate the invoice. You will not perform any mapping operations on this field.

INVNUMDET field

The INVNUMDET field is part of the application file format key but does not contain data that the system needs to use to generate the invoice. You will not perform any mapping operations on this field.

ITEMNUM Field

Overview   The ITEMNUM field can be directly mapped to the LIN01 Line Item Number element in the LIN segment (LIN group). However, you must calculate an incremental count of the number of line items. Use that total value in the Transaction Totals element (CNT:202).

How to calculate the number of line items   Use this procedure to count line items and generate a control total.

Step	Action
1	Double-click the LIN01 element (in the LIN segment in the LIN group). This field is typically used to count 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 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 will be the result of the earlier calculation. Any additional calculations you assign to that accumulator will be performed on the contents resulting from an earlier calculation.

(Contd) Step	Action
6	In the Name box, type "Line Item Number". This descriptive alias enables you to differentiate how the accumulators you create are used.
7	From the First list, select "Increment primary." This is the first operation that the system performs. This specifies that the system increases the value in this accumulator by one for each LIN 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 should now look like this:
9	Click **OK** on the Element Properties dialog box to add the standard rule to the LIN01 element.
10	Double-click the CNT:202 element (in the CNT:2 segment). This field 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. *(Continued on next page)*

(Contd) Step	Action
15	From the First list, select "Use primary." This specifies that the system will load the current value of the accumulator into the CNT:202 (Control Value) element.
16	Click **OK** to add the accumulator. The Standard Rule tab should now look like this.

| 17 | Click **OK** on the Element Properties dialog box to add the standard rule to the CNT:202 element. The CNT:202 element now contains the total number of line items in the invoice. |

How to map the item number

After using an accumulator to calculate the incremental count of the number of line items, map the ITEMNUM field directly to the LIN01 element.

To map the item number, link the application field ITEMNUM in the INVDETAIL record to the LIN01 element in the LIN segment (LIN group).

QTYSHIP Field

Overview
You have already promoted two iterations of the QTY segment in the LIN group. You can map the QTYSHIP field directly to the QTY:502 Quantity element in the QTY:5 segment in LIN group. However, you must map a constant to the Quantity Qualifier (QTY:501 element) and define a qualifying relationship with the Quantity (QTY:502 element). If the Quantity Qualifier is "12" (despatch quantity), that iteration of the QTY:5 segment contains the quantity shipped.

You must also count the quantity shipped for each line item and load the total quantity in the CNT:302 (Hash Total) field. Additionally, you must map a constant to the Control Qualifier (CNT:301 element) and define a qualifying relationship with the Monetary Amount (CNT:302) element. If the Control Qualifier is "1" (total quantity), that iteration of the CNT:3 segment contains the total quantity.

How to set the quantity qualifier
Map a constant to the Quantity Qualifier (QTY:501 element) and define a qualifying relationship with the Quantity (QTY:502) element.

You have already defined the constants that you will use for this map. Now map the appropriate constant to each Quantity Qualifier element.

Use this procedure to set the Quantity Qualifier for the QTY:5 segment.

Step	Action
1	Double-click the QTY:501 element (Quantity Qualifier) in the QTY:5 segment in the LIN 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 "12" to identify this iteration of the QTY:5 segment as containing the quantity shipped.
5	From the Qualifies list, select "6060:2 QUANTITY" to set up a qualifying relationship with the Quantity element.
6	Click **OK**. The "12" code is loaded in the Quantity Qualifier element and the qualifying relationship between the QTY:501 and QTY:502 elements is established.

How to calculate the number of line items

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

Step	Action
1	Double-click the QTY:502 element in the QTY:5 segment in the LIN group. This field is typically used 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 "1."
6	In the Name box, type "Total Quantity". This is a descriptive alias that enables you to differentiate how the accumulators you create are used.
7	From the First list, select "Sum in primary." This is the first operation that the system performs and specifies that the system adds the numeric value (takes the positive or negative sign of 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. The Standard Rule tab should now look like this:
9	Click **OK** on the Element Properties dialog box to add the standard rule to the QTY:502 element.
10	Double-click the CNT:302 element (in the CNT:3 segment). This field 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. (Continued on next page)

(Contd) Step	Action
15	From the First list, select "Use primary" to load the current value of the accumulator into the CNT:302 (Control Value) element. The Edit Accumulator Entry dialog box should now look like this:
16	Click **OK** to add the accumulator. The Standard Rule tab should now look like this;
17	Click **OK** on the Element Properties dialog box to add the standard rule to the CNT:302 element. The CNT:302 element now contains the total quantity of the purchase order.

How to set the control qualifier

You must map a constant to the Control Qualifier (CNT:301 element) and define a qualifying relationship with the Control Value (CNT:302) element.

You have already defined the constants that you will use for this map. Now map the appropriate constant to the Control Qualifier element.

(Continued on next page)

How to set the control qualifier (contd)

Use this procedure to set the Control Qualifier for the CNT:3 segment.

Step	Action
1	Double-click the CNT:301 element (Control Qualifier) in the CNT: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 "1" to identify this iteration of the CNT:3 segment as containing the total quantity.
5	From the Qualifies list, select the "6066:3 CONTROL VALUE" to set up a qualifying relationship with the Control Value element.
6	Click **OK**. The "1" code is loaded in the Control Qualifier element and the qualifying relationship between the CNT:301 and CNT:302 elements is established.

How to map the quantity shipped

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 QTY:502 element.

To map the QTYSHIP, link the application field QTYSHIP in the INVDETAIL record to the QTY:502 element in the QTY:5 segment (LIN group).

QTYORD Field

Overview
You have already promoted two iterations of the QTY segment (in the LIN group). The QTYORD field can be directly mapped to the QTY:602 Quantity element in the QTY:6 segment (LIN group). However, you must map a constant to the Quantity Qualifier (QTY:601 element) and define a qualifying relationship with the Quantity (QTY:602 element). If the Quantity Qualifier is "21" (ordered quantity), that iteration of the QTY:6 segment contains the quantity ordered.

How to set the quantity qualifier
You must map a constant to the Quantity Qualifier (QTY:601 element) and define a qualifying relationship with the Quantity (QTY:602) element.

You have already defined all the constants that you will use for this map. Now map the appropriate constant to each Quantity Qualifier element.

Use this procedure to set the Quantity Qualifier for the QTY:6 segment.

Step	Action
1	Double-click the QTY:601 element (Quantity Qualifier) in the QTY:6 segment in the LIN 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 "21" to identify this iteration of the QTY:6 segment as containing the quantity ordered.
5	From the Qualifies list, select "6060:3 QUANTITY" to set up a qualifying relationship with the Quantity element.
6	Click **OK**. The "21" code is loaded in the Quantity Qualifier element and the qualifying relationship between the QTY:601 and QTY:602 elements is established.

How to map the quantity ordered
After mapping the using an accumulator to calculate each line item's quantity invoiced and to generate a hash total, map the QTYORD field directly to the QTY:602 element.

To map the QTYORD, link the application field QTYORD in the INVDETAIL record to the QTY:602 element in the QTY:6 segment (LIN group).

UNITPRICE Field

Overview

You can map the UNITPRICE field directly to the MOA:1302 Monetary Amount element in the MOA:13 segment (MOA:2 group, LIN group). However, you must map a constant to the Monetary Amount Type Qualifier (MOA:1301 element) and define a qualifying relationship with the Monetary Amount (MOA:1302) element. If the Monetary Amount Type Qualifier is "146" (unit price), that iteration of the MOA:13 segment contains the unit price.

You must multiply the quantity invoiced for each line item by the unit price to obtain the extended price. You must also generate a running total of the extended price and load the final total in the MOA:1402 (Total Invoice Amount) element.

How to set the monetary amount type qualifier

You must map a constant to the Monetary Amount Type Qualifier (MOA:1301 element) and define a qualifying relationship with the Monetary Amount (MOA:1302) element.

You have already defined the constants that you will use for this map. Now map the appropriate constant to the Monetary Amount Type Qualifier element.

Use this procedure to set the Monetary Amount Type Qualifier for the MOA:13 segment.

Step	Action
1	Double-click the MOA:1301 element (Monetary Amount Type Qualifier) in the MOA:13 segment in the MOA:2 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 "146" to identify this iteration of the MOA:13 segment as containing the unit price.
5	From the Qualifies list, select the "5004:2 MONETARY AMOUNT" to set up a qualifying relationship with the Quantity element.
6	Click **OK**. The "146" code is loaded in the Monetary Amount Type Qualifier element and the qualifying relationship between the MOA:1301 and MOA:1302 elements is established.

How to calculate the value total

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

Step	Action
1	Double-click the QTY:502 element (in the QTY:5 segment in the LIN group). This field is typically used to count the quantity invoiced. The Element Properties dialog box is displayed.
2	Select the **Standard Rule** tab.
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."
5	In the Name box, type "Extended Price".
6	From the First list, select "Load primary." This is the first operation that the system performs and specifies that the system loads the contents of the element into the Primary Accumulator for each iteration of the QTY:5 group. The Edit Accumulator Entry dialog box should now look like this: *(Continued on next page)*

(Contd) Step	Action
7	Click **OK** to add the accumulator. The Standard Rule tab should now look like this:
8	Click **OK** on the Element Properties dialog box to add the standard rule to the QTY:502 element.
9	Double-click the MOA:1302 element (in the MOA:13 segment in the MOA:2 group in the LIN 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 Primary Accumulator list, select "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 MOA:1302 (Unit Price) element with the contents of the primary accumulator and stores the result in the primary accumulator for each iteration of the MOA:2 group.

The Edit Accumulator Entry dialog box should now look like this:

 |
| 15 | Click **OK** to add the accumulator. The Standard Rule tab should now look like this:

 |
| 16 | Click **OK** on the Element Properties dialog to add the standard rule to the MOA:1302 element.

Note
If you have an extended price element in your EDI file, load the total from the extended price calculation into that element. To do this, use an accumulator on that extended price element that specifies "Use primary" for accumulator 2. |

How to generate a running total

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

Step	Action
1	Double-click the MOA:1302 element (in the MOA:13 segment in the MOA:2 group in the LIN 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.
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."
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 MOA:2 group.
7	From the Alternate Accum list, select "2." This specifies that the system adds the value of accumulator "2," which contains the extended price for a line item, to the value of accumulator "3," which contains a running total of the extended price with each iteration of the MOA:2 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. The Standard Rule tab should now look like this:
9	Click **OK** on the Element Properties dialog box to add the standard rule to the MOA:1302 element.

How to load the running total

Use this procedure to load the running total of the extended price into the MOA:1402 (Monetary Amount) element.

Step	Action
1	Double-click the MOA:1402 element (in the MOA:14 segment in the MOA:2 group in the summary section of the map). This is the field that contains the total amount. 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 "3." (Continued on next page)

(Contd) Step	Action
6	From the First list, select "Use primary." This specifies that the system will load the contents of the primary accumulator into the MOA:1402 element. The Edit Accumulator Entry dialog box should now look like this:
7	Click **OK** to add the accumulator. The Standard Rule tab should now look like this:
8	Click **OK** on the Element Properties dialog box to add the standard rule to the MOA:1402 element.

How to map the unit price

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, map the UNITPRICE field directly to the MOA:1302 element.

To map the UNITPRICE, link the application field UNITPRICE in the INVDETAIL record to the MOA:1302 element in the MOA:13 segment (in the MOA:2 group in the LIN group).

How to map the   You have already mapped the quantity shipped to the QTY:5 segment and the quantity
UOM field   ordered to the QTY:6 segment. Map the unit of measure to both segments.

To map the UOM to both segments, Link the application field UOM in the INVDETAIL
record to the QTY:503 element in the QTY:5 segment (LIN group). Then link the
application field UOM in the INVDETAIL record to the QTY:603 element in the QTY:6
segment (LIN group).

CUSTPROCODE Field

Overview
You can map the CUSTPROCODE (customer product code) field directly to the LIN03 element in the LIN segment in the LIN group. However, you must map a constant to the LIN04 element (Item Number Type, Coded) and establish a qualifying relationship with the Item Number (LIN03). This enables you to specify that the LIN03 element contains your partner's product code.

How to set the line item type, coded
You must map a constant to the LIN04 element (Item Number Type, Coded) and establish a qualifying relationship with the Item Number (LIN03).

You have already defined the constants that you will use for this map. Now map the appropriate constant to the Item Number Type, Coded element.

Use this procedure to set a constant for the Item Number Type, Coded element.

Step	Action
1	Double-click the LIN04 element (Item Number Type, Coded) in the LIN segment in the LIN 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 LIN04 element as containing the customer product code.
5	From the Qualifies list, select "7140 ITEM NUMBER" to set up a qualifying relationship with the Item Number element.
6	Click **OK**. The "BP" code is loaded in the Item Number Type, Coded element and the qualifying relationship between the LIN03 and LIN04 elements is established.

How to map the customer product code
After establishing the appropriate qualifying relationship, map the CUSTPROCODE field.

To map the product code information, link the application field CUSTPROCODE in the INVDETAIL record to the LIN03 element in the LIN segment (LIN group).

UPCCODE Field

Overview   You can map the UPCCODE (your UPC code) field directly to the PIA:202 element in the PIA:2 segment (in the LIN group). However, you must map a constant to the PIA:203 element (Item Number Type, Coded) and establish a qualifying relationship with the Item Number (PIA:202). This enables you to specify that the PIA:202 element contains your UPC code.

Additionally, you must set the Product ID Function Qualifier for the PIA:2 segment (PIA:201 element).

How to set the line item type, coded   You must map a constant to the PIA:203 element (Item Number Type, Coded) and establish a qualifying relationship with the Item Number (PIA:202).

You have already defined the constants that you will use for this map. Now map the appropriate constant to the Item Number Type, Coded element.

Use this procedure to set a constant for the Item Number Type, Coded element.

Step	Action
1	Double-click the PIA:203 element (Item Number Type, Coded) in the PIA:2 segment in the LIN 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 "UP" constant (UPC Code) to identify the PIA:202 element as containing your UPC Code.
5	From the Qualifies list, select the "7140:7 ITEM NUMBER" to set up a qualifying relationship with the Item Number element.
6	Click **OK**. The "UP" code is loaded in the Item Number Type, Coded element and the qualifying relationship between the PIA:202 and PIA:203 elements is established.

How to set the product ID function qualifier	You must also map a constant to the PIA:201 element (Product ID Function Qualifier) and establish a qualifying relationship with the Item Number (PIA:202). You have already defined the constants that you will use for this map. Now map the appropriate constant to the Product ID Function Qualifier element. Use this procedure to set a constant for the Product ID Function Qualifier element.

Step	Action
1	Double-click the PIA:201 element (Product ID Function Qualifier) in the PIA:2 segment in the LIN 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 second "1" constant (additional information) to identify the PIA:202 element as containing additional product information.
5	From the Qualifies list, select the "7140:7 ITEM NUMBER" to set up a qualifying relationship with the Item Number element.
6	Click **OK**. The "1" code is loaded in the Product ID Function Qualifier element and the qualifying relationship between the PIA:201 and PIA:202 elements is established.

How to map your UPC code	After establishing the appropriate qualifying relationship, map the UPCCODE field. To map the product code information, link the application field UPCCODE in the INVDETAIL record to the PIA:202 element in the PIA:2 segment (LIN group).

How to map the ITEMDESC field	Since you have already promoted one instance of the IMD segment in the LIN group, you can map the ITEMDESC (item description) field directly to the IMD:306 element. To map the item description, link the application field ITEMDESC in the INVDETAIL record to the IMD:306 element in the IMD:3 segment (LIN group).

The INVSUMMARY Record

Overview

Introduction

This section describes how 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.

BILLTOACCTSUM field

The BILLTOACCTSUM field is part of the application file format key but does not contain data that the system uses to generate the invoice. You will not perform any mapping operations on this field.

DOCTYPSUM field

The DOCTYPSUM field is part of the application file format key but does not contain data that the system uses to generate the invoice. You will not perform any mapping operations on this field.

INVNUMSUM field

The INVNUMSUM field is part of the application file format key but does not contain data that the system uses to generate the invoice. You will not perform any mapping operations on this field.

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. Though you are mapping the total amount with accumulators, you must add an explicit link with an application field that contains data.

You must also map a constant to the Monetary Amount Type Qualifier (MOA:1401 element) and define a qualifying relationship with the Monetary Amount (MOA:1402) element. If the Monetary Amount Type Qualifier is "128" (total amount), that iteration of the MOA:14 segment contains the total invoice amount.

How to set the monetary amount type qualifier   You must map a constant to the Monetary Amount Type Qualifier (MOA:1401 element) and define a qualifying relationship with the Monetary Amount (MOA:1402) element.

You have already defined the constants that you will use for this map. Now map the appropriate constant to the Monetary Amount Type Qualifier element.

Use this procedure to set the Monetary Amount Type Qualifier for the MOA:14 segment.

Step	Action
1	Double-click the MOA:1401 element (Monetary Amount Type Qualifier) in the MOA:14 segment in the MOA:2 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 "128" to identify this iteration of the MOA:14 segment as containing the total invoice amount.
5	From the Qualifies list, select "5004:3 MONETARY AMOUNT" to set up a qualifying relationship with the Quantity element.
6	Click **OK**. The "128" code is loaded in the Monetary Amount Type Qualifier element and the qualifying relationship between the MOA:1401 and MOA:1402 elements is established.

How to map the total monetary value summary   To map the total monetary value summary, link the application field TOTALAMOUNT in the INVSUMMARY record to the MOA:1402 element in the MOA:14 segment (in the MOA:2 group).

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.

Additionally, you must map a constant to the Control Qualifier (CNT:201 element) and define a qualifying relationship with the Monetary Amount (CNT:202) element. If the Control Qualifier is "2" (number of line items in message), that iteration of the CNT:2 segment contains the total number of line items.

How to set the control qualifier

You must map a constant to the Control Qualifier (CNT:201 element) and define a qualifying relationship with the Control Value (CNT:202) element.

You have already defined all the constants that you will use for this map. Now map the appropriate constant to the Control Qualifier element.

Use this procedure to set the Control Qualifier for the CNT:2 segment.

Step	Action
1	Double-click the CNT:201 element (Control Qualifier) in the CNT:2 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 "2" to identify this iteration of the CNT:2 segment as containing the total number of line items.
5	From the Qualifies list, select "6066:2 CONTROL VALUE" to set up a qualifying relationship with the Control Value element.
6	Click **OK**. The "2" code is loaded in the Control Qualifier element and the qualifying relationship between the CNT:201 and CNG:202 elements is established.

How to map the total transaction amount

To map the transaction totals, link the application field TRANSTOTAL in the INVSUMMARY record to the CNT:202 element in the CNT:2 segment.

Finalizing the Map

Overview

Introduction   To complete the mapping process, you must 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 generates a translation object. The map that you created (PET_EINV.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.
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_EINV.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 as you named the map (.MAP file). **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 will be stored, if necessary. **Caution** Do not store the compiled translation object i the GENSRVNT\RegTransObj subfolder. This subfolder is reserved for storing a copy of each translation object you register with Gentran:Server for Windows. *(Continued on next page)*

(Contd) Step	Action
5	Click **Save**. The system compiles the map and generates a translation object. The Compile Errors dialog box is displayed. Verify that no errors occurred and click **OK** to exit the dialog box.
6	From the Edit menu, select **Details**. The Translation Object Details dialog box is displayed. The date on which the translation object was compiled is automatically loaded into the Compiled on box on the Translation Object Details dialog box. Click **Cancel** to exit the dialog box.
7	From the File menu, select **Save** to save the source map. **Note** You must register this translation object with the Gentran:Server system before you can use it. **Reference** See the Gentran:Server for Windows *User's 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, 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. **Print Options** Report sections ☑ Print branching diagram ☑ Print record details ☑ Print mapping details ☑ Print extended rules ☑ Print code lists ☑ Include only active objects OK Cancel
2	Accept the defaults on this dialog box (all options checked) so the report will include all sections and only the activated map components. Click **OK**. The Print dialog box is displayed.
3	Select the appropriate options and click **OK**. Click **Setup** to select a specific printer, paper orientation, or size and source of paper. The Print Setup dialog box is displayed.

Testing the Translation Object

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

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

Step	Action
1	Register the HEADER.TPL and PET_EINV.TPL translation objects with Gentran:Server for Windows.
2	If you have not already done so, import the PETZONE1.PAR partner relationship into Gentran:Server for Window. The default path for the partner file is: `C:\GENSRVNT\TUTORIAL\PETZONE1.PAR`
3	Verify (in Partner Editor) that the PET E 92.1 INVOIC 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.
5	Use the Import option in Gentran:Server for Windows to process the data file (PET_EINV.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 path for the data file is: `C:\GENSRVNT\TUTORIAL\PET_EINV.TXT`
6	After the message is translated, it is located in the Workspace in Gentran:Server for Windows. View the EDI data to ensure that the message was translated correctly.

Reference
See the Gentran:Server for Windows *User's Guide* for more information on registering a translation object, importing a partner, selecting outbound translation objects, and viewing raw EDI data.

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.
check box	A check box enables you to select or clear options from a list. You can select as many options from the list as you need. A selected check box contains a checkmark. The labels of unavailable options are dimmed.
click	Click indicates that you should position your cursor on top of the designated item and press and release your primary mouse button. 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.
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 will need to 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 sub-elements.
composite data element	A contains two or more component data elements or sub-elements. 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 will be 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.

element (contd)

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 sub-elements.

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 sub-group), 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 sub-groups 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.

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.
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 will be transferred to the In Drawer.
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.
list box	A list box displays a list of choices. Scroll bars are provided if there are more choices than can fit in the box.
literal constant	These are used by the system as a repository to store information that will be 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 will need to define the LS and LE segments for the loops you will be 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 will be 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.

radio buttons Radio buttons indicate a group of mutually exclusive options. You can only select one option at a time. Select an option by clicking the radio button or pressing the SPACE BAR until the option you want is selected. The selected option contains a black dot. The labels of unavailable choices are dimmed.

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 will generate 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 document(s) 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 will be 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. You will typically 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 will be 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 will know 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 EDIFACT INVOIC (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 object(s) will be 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 will be 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 element or field contents.

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 will be 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. |