

Gentran:Server® for Windows®

Application Integration TRADACOMS 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

Sterling Commerce Software Trade Secret Notice

THE GENTRAN:SERVER FOR WINDOWS SOFTWARE ("STERLING COMMERCE SOFTWARE") IS THE CONFIDENTIAL AND TRADE SECRET PROPERTY OF STERLING COMMERCE, INC., ITS AFFILIATED COMPANIES OR ITS OR THEIR LICENSORS, AND IS PROVIDED UNDER THE TERMS OF A LICENSE AGREEMENT. NO DUPLICATION OR DISCLOSURE WITHOUT PRIOR WRITTEN PERMISSION. RESTRICTED RIGHTS.

This documentation, the Sterling Commerce Software it describes, and the information and know-how they contain constitute the proprietary, confidential and valuable trade secret information of Sterling Commerce, Inc., its affiliated companies or its or their licensors, and may not be used for any unauthorized purpose, or disclosed to others without the prior written permission of the applicable Sterling Commerce entity. This documentation and the Sterling Commerce Software that it describes have been provided pursuant to a license agreement that contains prohibitions against and/or restrictions on their copying, modification and use. Duplication, in whole or in part, if and when permitted, shall bear this notice and the Sterling Commerce, Inc. copyright notice.

As and when provided to any governmental entity, government contractor or subcontractor subject to the FARs, this documentation is provided with RESTRICTED RIGHTS under Title 48 CFR 52.227-19. Further, as and when provided to any governmental entity, government contractor or subcontractor subject to DFARs, this documentation and the Sterling Commerce Software it describes are provided pursuant to the customary Sterling Commerce license, as described in Title 48 CFR 227-7202 with respect to commercial software and commercial software documentation.

These terms of use shall be governed by the laws of the State of Ohio, USA, without regard to its conflict of laws provisions. If you are accessing the Sterling Commerce Software under an executed agreement, then nothing in these terms and conditions supersedes or modifies the executed agreement.

Product names mentioned herein may be trademarks and/or registered trademarks of their respective companies. Gentran and Gentran:Server are registered trademarks of Sterling Commerce, Inc.

Third Party Software:

Portions of the Sterling Commerce Software may include products, or may be distributed on the same storage media with products, ("Third Party Software") offered by third parties ("Third Party Licensors").

Warranty Disclaimer

This documentation and the Sterling Commerce Software which it describes are licensed either "AS IS" or with a limited warranty, as set forth in the Sterling Commerce license agreement. Other than any limited warranties provided, NO OTHER WARRANTY IS EXPRESSED AND NONE SHALL BE IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE OR FOR A PARTICULAR PURPOSE. The applicable Sterling Commerce entity reserves the right to revise this publication from time to time and to make changes in the content hereof without the obligation to notify any person or entity of such revisions or changes.

The Third Party Software is provided 'AS IS' WITHOUT ANY WARRANTY AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. FURTHER, IF YOU ARE LOCATED OR ACCESSING THIS SOFTWARE IN THE UNITED STATES, ANY EXPRESS OR IMPLIED WARRANTY REGARDING TITLE OR NON-INFRINGEMENT ARE DISCLAIMED.

Table of Contents

Chapter 1 Getting Started

Overview	1-2
▶ Introduction	1-2
▶ In This Guide	1-3
Introducing Mapping	1-5
▶ Mapping Overview	1-5
▶ File Formats	1-7
The Application Integration Subsystem	1-10
▶ Overview	1-10
▶ Application Integration Window	1-11
▶ Customising Global Display Options	1-13
▶ Building a Map with Application Integration	1-18
▶ Using the Tutorial	1-20

Chapter 2 Inbound Purchase Order (ORDERS)

Overview	2-3
▶ Introduction	2-3
▶ Getting Started	2-4
▶ Analysing Your Application File Format	2-5
▶ Analysing the EDI File	2-9
▶ Reconciling Your Application File and the EDI File	2-12
The Export Map	2-16
▶ How to Create the Export Map	2-16
▶ How to Load the ORDHDR Message	2-21
▶ Activating EDI Map Components	2-23
▶ How to Verify EDI Delimiters	2-27
Application File Format	2-29
▶ Defining Your Application File Format	2-29
▶ How to Create the POHEADER Record	2-30
▶ How to Create the PODetailGroup	2-33
▶ Creating the PODetailGroup Subordinate Records	2-35
▶ How to Create the POSUMMARY Record	2-39
Creating the Application Fields	2-41
▶ Overview	2-41
▶ How to Create the POHEADER Record Application Fields	2-44

• How to Create the PODETAIL Record Application Fields	2-47
• How to Create the PODESC Record Application Fields	2-49
• How to Create the POSUMMARY Record Application Fields	2-51
Mapping Your Data	2-54
• Overview	2-54
Mapping the POHEADER Record	2-55
• Overview	2-55
• CUSTNUMHDR Field	2-56
• PODATEHDR Field	2-58
• PONUMHDR Field	2-59
• POTYPECODE Field	2-61
• Mapping the Shipping Information	2-62
• Mapping Customer Information	2-65
• DELDATE Field	2-68
• NUMLINEITEMS Field	2-69
Mapping the PODetailGroup Records: PODETAIL Record	2-70
• Overview	2-70
• CUSTNUMDET Field	2-71
• PODATEDET Field	2-73
• PONUMDET Field	2-74
• CUSTPROCEDURE and UPCCODE Fields	2-75
• CONSUMERUNITS Field	2-78
• UOM Field	2-80
• UNITPR Field	2-81
• QTYORD Field	2-82
PODESC Record	2-84
• Overview	2-84
• CUSTNUMDESC Field	2-85
• PODATEDESC Field	2-87
• PONUMDESC Field	2-88
• DESC Field	2-89
POSUMMARY Record	2-90
• Overview	2-90
• CUSTNUMSUM Field	2-91
• PODATESUM Field	2-93
• PONUMSUM Field	2-94
• NUMLINEITEMSUM Field	2-95
Finalising the Map	2-96
• Overview	2-96
• Compiling the Map	2-97
• Printing the Report	2-99
• Testing the Translation Object	2-100

Chapter 3 Outbound Invoice (INVOIC)

Overview	3-4
▶ Introduction	3-4
▶ Getting Started	3-5
▶ Analysing Your Application File Format	3-7
▶ Analysing the EDI File	3-12
▶ Correlating Your Application File and the EDI File	3-14
The System Import Map	3-19
▶ Creating the System Import Map	3-19
▶ Creating the HEADER Record	3-23
▶ Creating the HEADER Fields	3-25
▶ Defining the Key	3-29
▶ Compiling the Map	3-33
▶ Printing the Report	3-35
▶ Testing the Translation Object	3-36
The Import Map	3-37
▶ Creating the Import Map	3-37
▶ Activating EDI Map Components	3-42
▶ Verifying EDI Delimiters	3-46
Defining the Rest of Your Application File Format	3-48
▶ Overview	3-48
▶ Copying the HEADER Record	3-49
▶ Creating the INVBEGIN Record	3-51
▶ Creating the INVCOMMENT Group	3-53
▶ Creating the INVCOMMENT Record	3-55
▶ Creating the INVDETAIL Record	3-57
▶ Creating the TEMPVATSTL Record	3-59
▶ Creating the INVSUMMARY Record	3-61
Creating the Application Fields	3-63
▶ Overview	3-63
▶ How to Define the INVBEGIN Record Application Fields	3-66
▶ How to Define the INVCOMMENT Record Application Fields	3-69
▶ How to Define the INVDETAIL Record Application Fields	3-71
▶ How to Define the TEMPVATSTL Record Application Fields	3-73
▶ How to Define the INVSUMMARY Record Application Fields	3-75
Mapping Your Data	3-78
▶ Overview	3-78
▶ Mapping the INVNUMBEG Field	3-79
▶ How to Map the INVDATE Field	3-81
▶ Mapping the PONUM Field	3-82
▶ How to Map the PODATE Field	3-86

Mapping the Shipping Information	3-87
▶ How to Map the ShipTo Fields	3-87
The INVCommentGroup: INVCOMMENT Record	3-92
▶ Overview	3-92
▶ Mapping the MESSAGETEXT Information	3-93
The INVCommentGroup: INVDETAIL Record	3-95
▶ Overview	3-95
▶ How to Map the ITEMNUM Field	3-96
▶ Mapping the QTYSHIP Field	3-97
▶ How to Map the UNITPRICE Field	3-98
▶ How to Map the UOM Field	3-99
▶ How to Map the CUSTPROCEDURE Field	3-100
▶ How to Map the UPCCODE Field	3-101
▶ How to Map the ITEMDESC Field	3-102
▶ Mapping the VATCODE Field	3-103
▶ How to Map the TEMPVATRATE Field	3-105
▶ Mapping the TEMPLEXC Field	3-106
The INVCommentGroup: TEMPVATSTL Record	3-112
▶ Overview	3-112
▶ How to Map the SEQA Field	3-113
▶ How to Map the VATRATE Field	3-114
▶ How to Map the VATRATECODE Field	3-115
▶ How to Map the NUMLINES Field	3-116
▶ How to Map the LVLA Field	3-117
▶ How to Map the VATA Field	3-118
The INVCommentGroup: INVSUMMARY Record	3-119
▶ Overview	3-119
▶ How to Map the TOTALAMOUNT Field	3-120
▶ Producing the VAT Summary	3-121
▶ Producing the Invoice Trailer	3-127
Finalising the Map	3-130
▶ Overview	3-130
▶ How to Compile the Map	3-131
▶ How to Print the Report	3-133
▶ Testing the Translation Object	3-134
▶ How to Populate the SDT and CDT Segments	3-135

Glossary

Getting Started

Contents	Overview.....	1 - 2
	▶ Introduction.....	1 - 2
	▶ In This Guide.....	1 - 3
	Introducing Mapping	1 - 5
	▶ Mapping Overview	1 - 5
	▶ File Formats.....	1 - 7
	The Application Integration Subsystem	1 - 10
	▶ Overview	1 - 10
	▶ Application Integration Window.....	1 - 11
	▶ Customising Global Display Options	1 - 13
	▶ Building a Map with Application Integration.....	1 - 18
	▶ Using the Tutorial	1 - 20

Overview

Introduction

**Gentran:Server
for Windows
Application
Integration**

Welcome to Gentran:Server[®] for Windows[®] Application Integration subsystem, Sterling Commerce's electronic data interchange (EDI) 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 messages you send to your partners (outbound mapping), and to translate EDI standard formats to your application format for messages that you receive from your partners (inbound mapping).

In This Guide

Overview

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

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

Note

This manual 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 dialogue boxes, icons, windows, and buttons.
- Using the Windows Program Manager.

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

Contents

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

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

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

Introducing Mapping

Mapping Overview

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

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

Outbound mapping For outbound mapping, the system translates your application file format to EDI standard formats, so you can send messages to your partners. To translate outbound data, you need to create an import map and a system import map in the Gentran:Server for Windows Application Integration subsystem. You need to create an import map because the system is *importing from* your application file. The import map defines how to move data from your application file (flat file definition), which may contain multiple messages, to the EDI standard-formatted messages that your partners expect to receive from you. You need to create a system import map to determine which trading relationship (established in Partner Editor) corresponds to each message in the application file, so the system will know which import map to use to process the message. Please see *Creating the System Import Map* on page 3 - 19 for more information on creating a system import map.

Translation objects After you create a map in the Application Integration subsystem, this map is the *source* from which you compile a translation object. A translation object is a set of rules that instruct the translator how to convert a file from one format to another. Every function performed during translation requires a translation object. Translation objects are used to control all the functions of Gentran:Server for Windows EDI translation. Translation objects can belong to the following two categories: system translation objects or user translation objects. 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, message turnaround, and creating printed reports. All the required system translation objects are automatically installed with the Gentran:Server for Windows system.

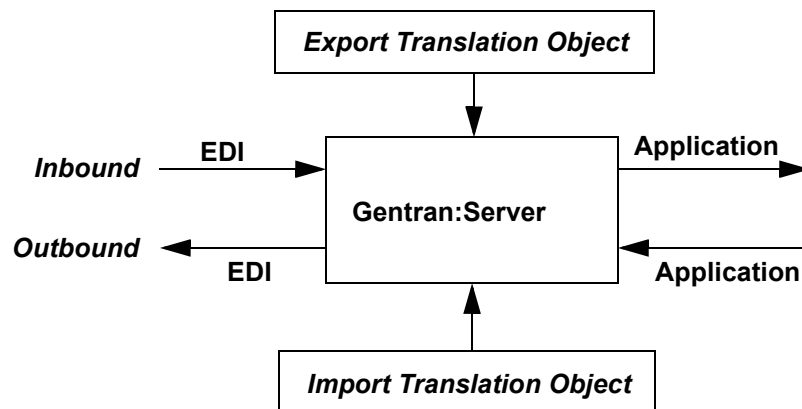
(Continued on next page)

Translation objects (contd)

The Gentran:Server for Windows Application Integration subsystem enables you to generate import, export, and message turnaround translation objects. The Gentran:Server for Windows Forms Integration subsystem enables you to generate data entry and print translation objects. Once you generate a translation object using the Gentran:Server for Windows Application Integration subsystem (or the Gentran:Server for Windows Forms Integration subsystem), you must register the translation object with Gentran:Server for Windows (see the Gentran:Server for Windows *User's Guide* for more information). In addition, you must establish correct trading relationships for all trading partners from which you will receive or to which you will send data (see the Gentran:Server for Windows *User's Guide* for more information). During the process of establishing a trading relationship, you need to specify the name of the appropriate translation object that the translator will use to translate the data (see the Gentran:Server for Windows *User's Guide* for more information).

Diagram: Inbound and outbound translation process

This diagram illustrates the inbound and outbound translation process.



Inbound and outbound translation

When you create an import map, you need to define to Gentran:Server for Windows your application file format (your data that you will send to your partners) and the EDI standard format in which your partners expect to receive the messages. When you create an export map, you need to define to Gentran:Server for Windows the EDI standard format in which your partners will send messages to you 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 Application Integration subsystem. In Gentran:Server for Windows terminology, your application file is also referred to as a *fixed-format file* or a *positional file*. Your application file must contain all the information that you either need to extract from your partner's message (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 components

This table describes the application components used 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.

(Continued on next page)

Application components (contd)**Notes**

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

Before you define your application file format, you should obtain a layout of the necessary records, fields, and groups. Each map component will be arranged sequentially in the order that it is most logical for the system to process. Therefore, each level of your application file must be created sequentially. For example, your application file contains records and groups. The records contain fields and the groups contain records and/or 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 all the information that you expect to receive from your partner (if the map is inbound) or need to send to your partner (if the map is outbound), so that your system can accurately process the data.





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

However, if you want to use a specialised version 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.

Regardless of whether the system generates the EDI file or you define it, the specific EDI map components that you will use will depend on the type of map you are creating. This includes the standard, version, and transaction set (message) selected, and which groups, segments, composites, and elements your company requires. We recommend that you determine which map components you will be using before generating or defining an EDI file.

EDI components

This table describes the EDI components Gentran:Server for Windows uses to define the EDI file format.

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.

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 customising the properties of the map components, and using specialised Gentran:Server for Windows functions to manipulate the EDI file structure.

The Application Integration Subsystem

Overview

Introduction

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

Navigation

The Gentran:Server for Windows Application Integration Window allows you to access its functionality in four different ways, as follows:

- 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 that contains all the functions that are currently available for that map component. The shortcut menus enable you to quickly and easily access available functionality. The content of the shortcut menus will vary, depending on the type and level of the selected map component.

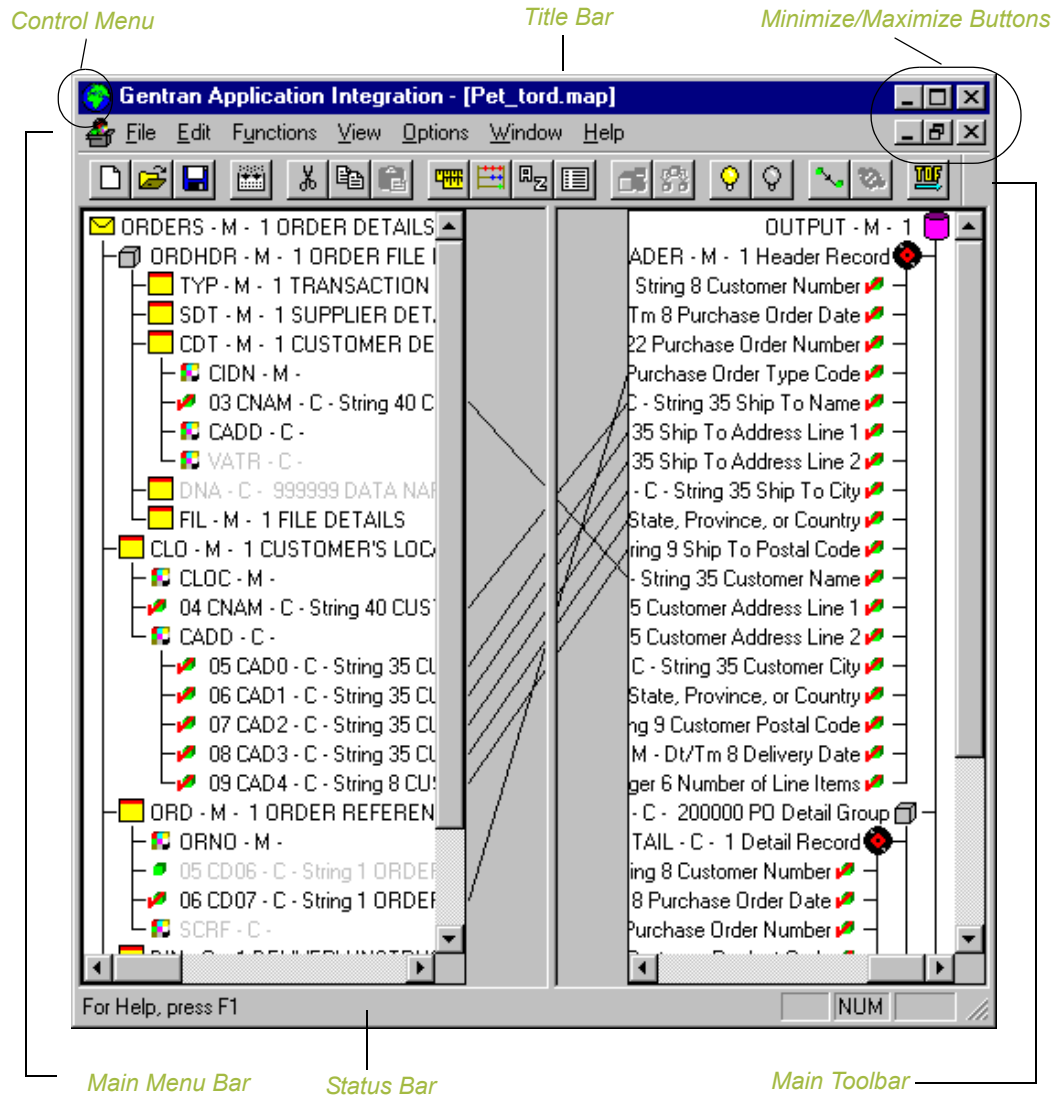
Note

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

Application Integration Window

Application integration window

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



Application integration window components

This table describes the Application Integration Window components.

Part	Function
Control Menu	Contains the following commands: <ul style="list-style-type: none"> ▶ 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).

Customising Global Display Options

Overview

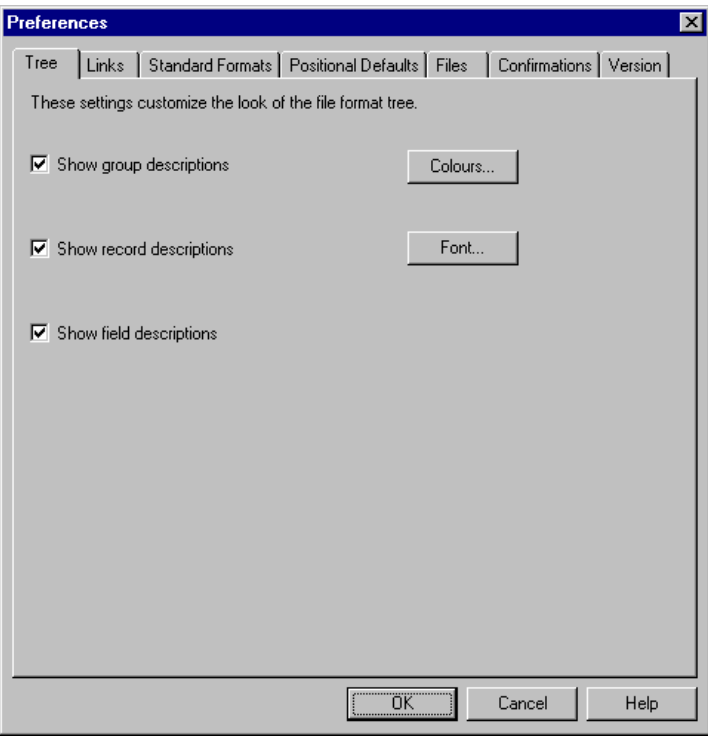
The Gentran:Server for Windows Application Integration subsystem allows you to customise the display of maps in several different ways. The global display options that are established apply to all maps. This section explains how to customise the global display of the following:

- ▶ Customising global display options
- ▶ Displaying links (global option)

The global map display options can be set or changed at any time.

How to customise global display options

Use this procedure to customize global display options.

Step	Action
1	<p>From the Options menu, select Preferences to display the Preferences dialog box (the Tab tree is displayed by default). The Preferences dialog box is a property sheet that enables you to set global defaults for Gentran:Server for Windows.</p>  <p style="text-align: right; color: green;">(Continued on next page)</p>


Step	Action
2	<p>To turn on the default display of group, record (segment), and field (element) descriptions, click the check box in front of each in the List section of the dialog box (so that a tick is displayed in the check box).</p> <p>To turn off the default display of group, record (segment), and field (element) descriptions, click the appropriate check box again to remove the tick.</p> <p>Typically, you will want to have all the descriptions displayed for reference. However, for a variety of reasons, you may not want the descriptions displayed. Depending on the size of your monitor, it may be easier to see the entire map if the descriptions are not displayed.</p> <p>Note You may want to experiment with shrinking the size of the font for the map before you turn off the display of descriptions.</p> <p>Reference See <i>Customizing Global Fonts</i> in the <i>Gentran:Server for Windows Application Integration User's Guide</i> for more information.</p>
3	Click OK to save the changes and exit the Preferences dialog box.

How to Customise global display of links

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

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

Step	Application
1	<p>From the Options menu, select Preferences to display the Preferences dialog box. The Preferences dialog box is a property sheet that enables you to set global defaults for Gentran:Server for Windows.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

Step	Application
2	<p>Click the Links tab to display the linking options. The Links tab enables you to set the default display of mapping links.</p> 
3	<p>On the Links section of the Options dialog box, select from these options:</p> <ul style="list-style-type: none"> ▶ Show no links: do not visually display mapping links ▶ Show links to or from the currently selected element: display only the mapping link(s) 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. <p>Select the option for the link display option you want to be the default for all maps.</p>
4	Click OK to save the changes and exit the Preferences dialog box.

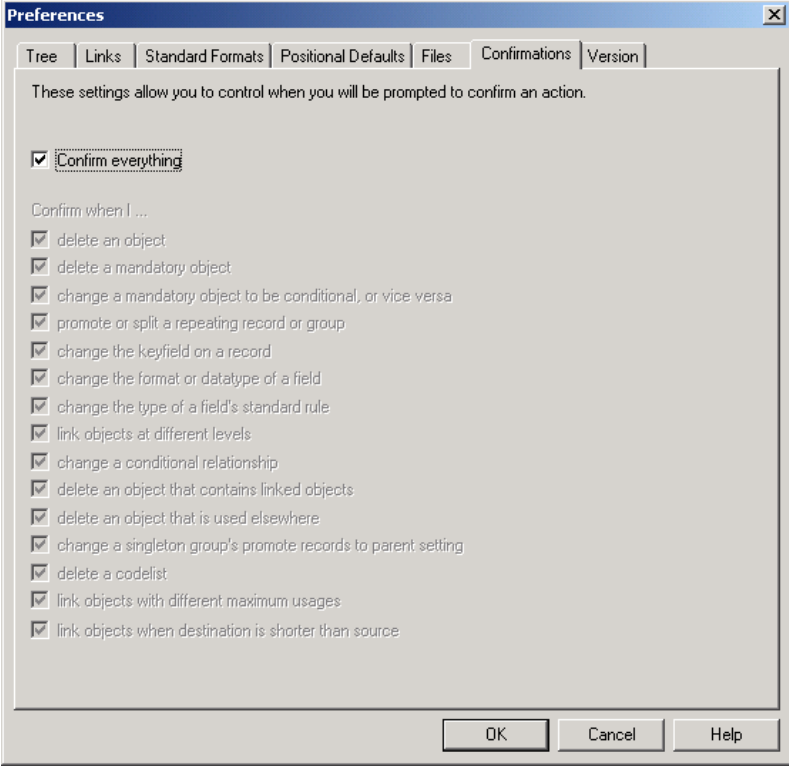
How to customise global confirmation options

The Confirmations tab on the Preferences dialogue 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	<p>From the Options menu, select Preferences.</p> <p>System response The system displays the Preferences dialogue box (Tree tab displayed by default).</p>
2	<p>Select the Confirmation tab.</p> <p>System response The system displays the confirmation options.</p>  <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
3	Set the global confirmation options by either selecting the “Confirm everything” check box (displays all confirmation messages) or by selecting individual confirmation messages by action performed.
4	Click OK to save the confirmation options. System response The system sets the confirmation options you selected.

Building a Map with Application Integration

Overview This section defines how to proceed with building a map using the Gentran:Server Application Integration subsystem.

Mapping process This table provides an overview of the mapping process.

Stage	Description
1	<p>Preparation and Analysis</p> <p>Obtain a layout of your application file and determine how it corresponds with the EDI standard you will be using. Determine how you will move data to or from each application field.</p>
2	<p>Set Global Defaults (first time only)</p> <p>The first time you use the Gentran:Server for Windows Application Integration subsystem, you should establish the default date format that the system will use. Please see the Gentran:Server for Windows <i>User's Guide</i> for more information on setting the global date default.</p>
3	<p>Create, Save, and Name a New Map</p> <p>Please see <i>How to Create the Export Map</i> on page 2 - 16 for more information on creating, saving, and naming a map.</p>
4	<p>Activate the Appropriate EDI Groups, Segments, and Elements</p> <p>Please see <i>Activating EDI Map Components</i> on page 2 - 23 for more information on activation.</p>
5	<p>Define your Application File Format</p> <p>Please see <i>Defining Your Application File Format</i> on page 2 - 29 for more information on defining your application file format.</p>
6	<p>Map the Appropriate Data for Each Application Field</p> <p>Please see <i>Mapping Your Data</i> on page 2 - 54 for more information on mapping your data.</p>
7	<p>Compile the Translation Object</p> <p>Please see <i>Compiling the Map</i> on page 2 - 97 for more information on compiling the translation object and translation object naming conventions.</p>

(Continued on next page)

(Contd) Stage	Description
8	<p>Print the Mapping Report</p> <p>Validate and review the map, and make modifications as needed. Please see <i>Printing the Report</i> on page 2 - 99 for more information on printing the mapping report.</p>
9	<p>Register the Translation Object with Gentran:Server for Windows</p> <p>Please see the Gentran:Server for Windows <i>User's Guide</i> for more information on registering translation objects.</p>
10	<p>Create the Appropriate Trading Relationship</p> <p>Establish the appropriate trading relationship in Gentran:Server for Windows for your trading partners. Please see the Gentran:Server for Windows <i>User's Guide</i> for more information on creating trading relationships. The trading relationship that you will use in this tutorial (PETZONE3) is provided on the tutorial diskette(s). You need to import the partner profile (PETZONE3.PAR) into Gentran:Server before you begin this tutorial. Please see the Gentran:Server for Windows <i>User's Guide</i> for more information on importing trading relationships.</p>
11	<p>Test the Translation Object</p> <p>Obtain test data from your partner(s) and process the data. Verify acknowledgement processing (if applicable). Verify communications with your network.</p>

Using the Tutorial

Tutorial basics

For these tutorials, we provide you with test data for the translation objects you create.

For the inbound (export) translation object, the test data is PETTSTT1.INT. The data file is located in the TUTORIAL subfolder under the folder where Gentran:Server is installed. The default full path for the data file is the following:

```
C:\GENSRVNT\TUTORIAL\PETTSTT1.INT
```

For the outbound (system import and import) translation objects, the test data is PET_TINV.TXT. The data file is located in the TUTORIAL subfolder under the folder where Gentran:Server is installed. The default full path for the data file is the following:

```
C:\GENSRVNT\TUTORIAL\PET_TINV.TXT
```

Inbound Purchase Order (ORDERS)

Contents	Overview	2 - 3
	▶ Introduction.....	2 - 3
	▶ Getting Started.....	2 - 4
	▶ Analysing Your Application File Format.....	2 - 5
	▶ Analysing the EDI File.....	2 - 9
	▶ Reconciling Your Application File and the EDI File.....	2 - 12
	The Export Map	2 - 16
	▶ How to Create the Export Map.....	2 - 16
	▶ How to Load the ORDHDR Message.....	2 - 21
	▶ Activating EDI Map Components.....	2 - 23
	▶ How to Verify EDI Delimiters.....	2 - 27
	Application File Format	2 - 29
	▶ Defining Your Application File Format.....	2 - 29
	▶ How to Create the POHEADER Record.....	2 - 30
	▶ How to Create the PODetailGroup.....	2 - 33
	▶ Creating the PODetailGroup Subordinate Records.....	2 - 35
	▶ How to Create the POSUMMARY Record.....	2 - 39
	Creating the Application Fields	2 - 41
	▶ Overview.....	2 - 41
	▶ How to Create the POHEADER Record Application Fields.....	2 - 44
	▶ How to Create the PODETAIL Record Application Fields.....	2 - 47
	▶ How to Create the PODESC Record Application Fields.....	2 - 49
	▶ How to Create the POSUMMARY Record Application Fields.....	2 - 51
	Mapping Your Data	2 - 54
	▶ Overview.....	2 - 54
	Mapping the POHEADER Record	2 - 55
	▶ Overview.....	2 - 55
	▶ CUSTNUMHDR Field.....	2 - 56
	▶ PODATEHDR Field.....	2 - 58

Overview

Introduction

In this chapter

This tutorial exercise defines the step-by-step tasks that are necessary for you to create a map that will be used to translate TRADACOMS purchase orders (ORDERS) that you receive from your trading partner.

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

- Mapping analysis
- Creating the new map
- Activating EDI map components
- Verifying EDI Delimiters
- Defining your application file format
- Mapping your data
- Finalising the map

Note

Press **F1** from any area of the system to access the online Help for that function. The online Help system contains field definitions and other detailed information.

Getting Started

Overview

This tutorial exercise is meant to be completed in its entirety. We recommend that you complete each task sequentially, without referring to the completed map that is provided with the installation program (although you may want to verify your completed map with the model).

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

Inbound mapping assumptions and scenario

Make these assumptions in this tutorial exercise:

- Your company is MWT Manufacturing Co., which manufactures a wide variety of supplies for pet stores.
- MWT Manufacturing Co. sells 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. Once 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 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 the analysis of the mapping requirements. This is the most important step in creating a successful map. If the analysis you perform is complete, you will have all the information you need to create the map in an efficient and logical manner. If you omit this critical step and proceed directly to creating the map, it is likely that creating the map will be a much longer and arduous task, and the map may be invalid due to oversights and omissions.

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

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

After you complete these steps, you can begin creating the map.

Analysing Your Application File Format

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

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

Application file layout This table provides 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 (DD/MM/YY)	8	8
PONUMHDR	X	Purchase Order Number	String	1	22
POTYPECODE		Purchase Order Type Code	String	0	2
SHIPTONAME		Ship To Name	String	0	35
SHIPTOADD1		Ship To Address Line 1	String	0	35
SHIPTOADD2		Ship To Address Line 2	String	0	35
SHIPTOCITY		Ship To City	String	0	35
SHIPTOSTATE		Ship To State, Province, or Country	String	0	35
SHIPTOPOSTALCODE		Ship To Postal Code	String	0	9
CUSTNAME		Customer Name	String	0	35
CUSTADD1		Customer Address Line 1	String	0	35
CUSTADD2		Customer Address Line 2	String	0	35
<i>(Continued on next page)</i>					

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
CUSTCITY		Customer City	String	0	35
CUSTSTATE		Customer State, Province, or Country	String	0	35
CUSTPOSTALCODE		Customer Postal Code	String	0	9
DELDATE	X	Delivery Date	Dt/Tm (DD/MM/YY)	8	8
NUMLINEITEMS	X	Number of Line Items	Number (N0)	1	6
PO Detail Group (PODetailGroup) up to 999,999 per message					
Detail Record (PODETAIL) 1 per message					
CUSTNUMDET	X	Customer Number	String	1	8
PODATEDET	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMDET	X	Purchase Order Number	String	1	22
CUSTPROCEDURE	X	Customer Product Code	String	1	30
UPCCODE	X	UPC Code	String	1	30
CONSUMERUNITS		Consumer Units in Trading Unit	Number (N0)	0	15
UOM		Unit of Measure	String	0	7
UNITPR		Unit Price	Number (R2)	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 (DD/MM/YY)	8	8
PONUMDESC	X	Purchase Order Number	String	1	22
DESC		Description	String	0	81
<i>(Continued on next page)</i>					

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Summary Record (POSUMMARY) 1 per message					
CUSTNUMSUM	X	Customer Number	String	1	8
PODATESUM	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
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 the Application Integration subsystem (name, mandatory status, description, data type, and field length). Each record contains a group of logically-related application fields. These fields will contain the information from the EDI file that your system needs to process the purchase order. The group contains logically-related application records.

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

- your partner's customer number
- the purchase order date
- the 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 999,999 times per order translated. This equals a maximum order of 999,999 items. The information within the PODetailGroup is further segregated by record.

(Continued on next page)

Record/Group	Definition
PODETAIL (detail)	Contains line item code number, unit of measure, price, and quantity ordered. The PODETAIL record occurs once per PODetailGroup.
PODESC (description)	<p>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.</p> <p>Example If the line item is a “dog collar,” the description fields may be “red,” “large,” and “leather.”</p>
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 analysed your application file format, the next step is to determine the requirements of the purchase order that you will receive from your partner(s).

Analysing the EDI File

Overview

To analyse the EDI file, you must first determine what your trading partner(s) will be sending you. You and your partner(s) need to agree on which standard, version, and message you will use. In this tutorial, we are using the TRADACOMS standard, version ANA001, message ORDERS (purchase order message), release 9. It is important to know the information that your partner(s) will send you within the purchase order, and what data will be contained in each element used.

It is very helpful to review the EDI standard(s) that you will use before analysing the EDI file. Obtain an EDI standard manual for each standard and version that you will use. Standard manuals are available from EDI standard agencies.

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

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

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.

Map components

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

Group	Segment	Comp.	Element	
ORDHDR (already active)	SDT (already active)	SIDN	SID0	
		CDT (already active)	CIDN	CID0
				CNAM
			CADD	CAD0 CAD1 CAD2 CAD3 CAD4
	CLO (already active)	CLOC	CLO0	
			CNAM	
		CADD	CAD0 CAD1 CAD2 CAD3 CAD4	
	ORD (already active)	ORNO (already active)	ORN0 ORN2	
			CD07	
	DIN		LDAT	
OLD (already active)	OLD:2	SPRO	SPR1	
		CPRO	CPR1	
		UNOR	UNO0 CD04	
		OQTY	OQT0	
		OUCT	OUC0	
		TDES	TDE0 TDE1	

(Continued on next page)

**Map components
(contd)****Note**

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

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

Recommendation

Also, 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. The Document Name allows you to differentiate between messages in the document browsers in Gentran:Server.

To set up a Document Name, you must select an element or field in the map that contains data that will distinguish the messages translated by this map. Then, the Name field of the document browsers that contain this message will contain the data from the selected element or field. You can choose any element or field as the Document Name, but we strongly recommend that you select an element or field that occurs only once in the message and has a data type of “String”. For a purchase order, the document name is typically the Customer’s Order Number.

Next step

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

Reconciling Your Application File and the EDI File

Overview To reconcile your application file format with the EDI file, for each application field you must determine where the information with which you need to populate the field is located in the EDI file, and how you can map that information to the application. To map information to a field, you can use linking, standard rules, extended rules, or a combination of the 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 (hereafter referred to as fields) is visually represented with a line connecting the two fields.

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

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

Recommendation

We recommend that you determine which mapping operations will be 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 ORD ORNO ORN2
PONUMHDR	Purchase Order Number	Link to ORD ORNO ORN0
POTYPECODE	Purchase Order Type Code	Link to ORD CD07
SHIPTONAME	Ship To Name	Link to CLO CNAM
SHIPTOADD1	Ship To Address Line 1	Link to CLO CADD CAD0
SHIPTOADD2	Ship To Address Line 2	Link to CLO CADD CAD1
SHIPTOCITY	Ship To City	Link to CLO CADD CAD2
<i>(Continued on next page)</i>		

(Contd) Field Name	Description	Mapping Operations
SHIPTOSTATE	Ship To State, Province, or Country	Link to CLO CADD CAD3
SHIPTOPOSTALCODE	Ship To Postal Code	Link to CLO CADD CAD4
CUSTNAME	Customer Name	Link to CDT CNAM
CUSTADD1	Customer Address Line 1	Link to CDT CADD CAD0
CUSTADD2	Customer Address Line 2	Link to CDT CADD CAD1
CUSTCITY	Customer City	Link to CDT CADD CAD2
CUSTSTATE	Customer State, Province, or Country	Link to CDT CADD CAD3
CUSTPOSTALCODE	Customer Postal Code	Link to CDT CADD CAD4
DELDATE	Delivery Date	Link to DIN LDAT
NUMLINEITEMS	Number of Line Items	Link to OTR LORD
PO Detail Group (PODetailGroup) up to 999,999 per message		
Detail Record (PODETAIL) 1 per message		
CUSTNUMDET	Customer Number	Extract Application ID from Partner record
PODATEDET	Purchase Order Date	Link to ORD ORNO ORN2 using extended rule
PONUMDET	Purchase Order Number	Link to ORD ORNO ORN0 using extended rule
CUSTPROCEDURE	Customer Product Code	Link to OLD:2 CPRO CPR1
UPCCODE	UPC Code	Extract Partner xref by partner item from subtable "PRODCODE"
CONSUMERUNITS	Consumer Units in Trading Unit	Link to OLD:2 UNOR UNO0
UOM	Unit of Measure	Link to OLD:2 UNOR CD04
UNITPR	Unit Price	Link to OLD:2 OUCT OUC0
QTYORD	Quantity Ordered	Link to OLD:2 OQTY OQT0
<i>(Continued on next page)</i>		
Description Record (PODESC) up to 5 per message		

(Contd) Field Name	Description	Mapping Operations
CUSTNUMDESC	Customer Number	Extract Application ID from Partner record
PODATEDESC	Purchase Order Date	Link to ORD ORNO ORN2 using extended rule
PONUMDESC	Purchase Order Number	Link to ORD ORNO ORN0 using extended rule
DESC	Description	Link to OLD:2 TDES TDE0 and OLD:2 TDES TDE1 after concatenating them with an extended rule
Summary Record (POSUMMARY) 1 per message		
CUSTNUMSUM	Customer Number	Extract Application ID from Partner record
PODATESUM	Purchase Order Date	Link to ORD ORNO ORN2
PONUMSUM	Purchase Order Number	Link to ORD ORNO ORN0
NUMLINEITEMSUM	Number of Line Items	Link to OTR LORD

Cross-reference tables

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

Reference

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

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	

Note

You do not need to create this cross-reference table; it is included with the PETZONE3 partner (PETZONE3.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

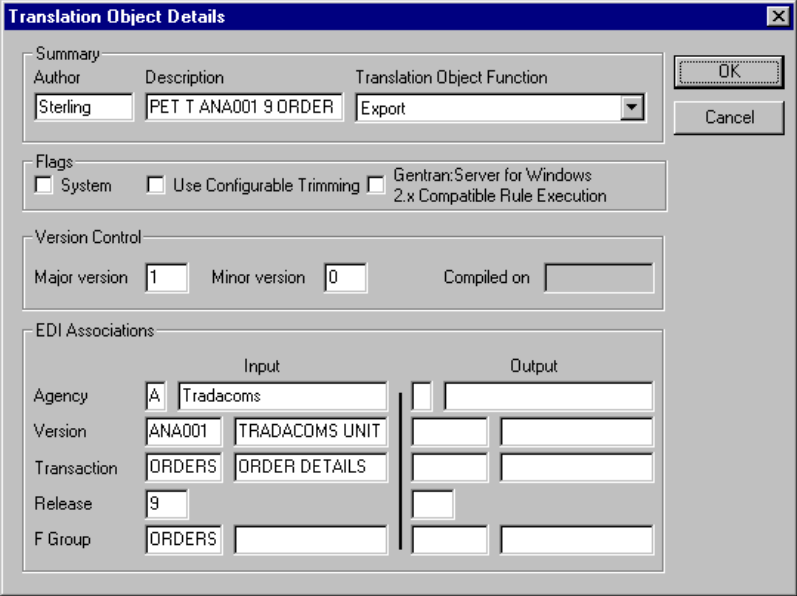
How to Create the Export Map

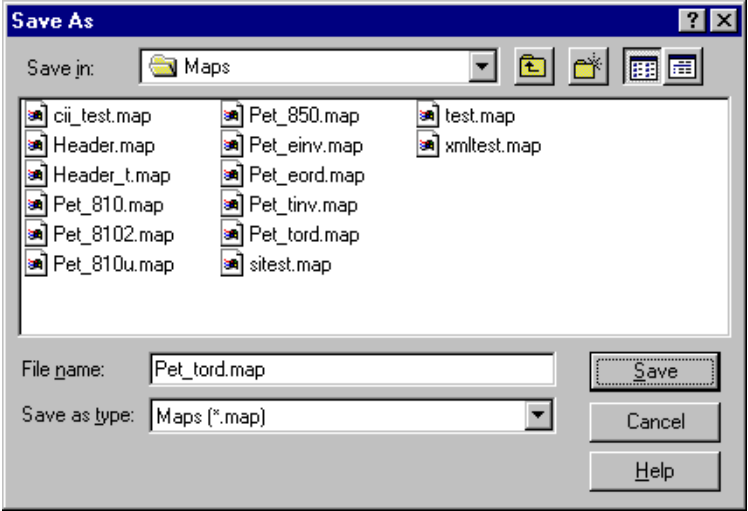
Overview The first task in creating a new map is to define the format that will be used for the input and output sides of the map. In this example, you are creating an export map so Gentran:Server for Windows can translate a purchase order that your partner sends you to your application file. Therefore, the input side of the map needs to reflect the TRADACOMS ORDERS purchase order that you will receive. Instead of defining the ORDERS message to the system yourself, you want the Application Integration subsystem to use the standards database on your system to build the purchase order according to the standard. The output side of the map needs to reflect your application file, which you will define to the Application Integration subsystem.

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

Step	Action
1	<p>From the File menu, select New.</p> <p>System response The system displays the New Map Wizard.</p>
2	<p>Answer the following questions and then click Next.</p> <ul style="list-style-type: none"> ▶ <i>What kind of map are you creating?</i> Select Export. ▶ <i>What is the name of the map?</i> Type PET_TORD as the unique name of the map. The system adds the .MAP extension. ▶ <i>What is your name?</i> Type your name if it differs from the user name prompted by the system. <p>System response The system displays the New Map Wizard - Input Format dialogue box.</p> <p>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.</p>
3	<p>From the Create a new data format using this syntax list, select Delimited EDI and click Customize.</p> <p>System response The system displays the New Delimited EDI Wizard.</p> <p style="text-align: right;">(Continued on next page)</p>

(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: <ol style="list-style-type: none"> a. Select the [A] Tradacoms standards agency, b. [ANA001 TRADACOMS UNITED KINGDOM] version, c. [ORDERS] ORDER DETAILS transaction set, and d. 9 release (for TRADACOMS only). e. Click Next.
6	Click Finish . System response The system loads the selected transaction set and returns to the New Map Wizard - Input Format dialogue box.
7	Click Next . System response The system displays the New Map Wizard - Output Format dialogue box.
8	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.
9	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. <div style="text-align: right; color: green;">(Continued on next page)</div>

(Contd) Step	Action
10	<p>From the Edit menu, select Details.</p> <p>System response The system displays the Translation Object Details dialogue box.</p> 
11	In the Description field, type PET T ANA001 9 ORDERS Export .
12	<p>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.</p> <p>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).</p>
13	<p>Click OK to save the translation object details.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
14	<p>From the File menu, select Save.</p> <p>System response The system displays the Save As dialogue box.</p>  <p style="text-align: right; color: green;">(Continued on next page)</p>



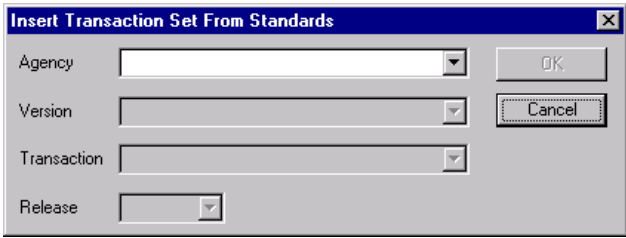
(Contd) Step	Action
15	<p>Verify that the File name box contains pet_tord.map.</p> <p>Note You allocate a field name to a map when you save the map for the first time or use the Save As function to save a copy of the map under a different name. The name of the map must be one to eight unique characters 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:</p> <ul style="list-style-type: none"> • 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, <i>but</i> with the file extension .TPL (e.g., HEADER_T.TPL). <p>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.</p> <p>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.</p>
16	<p>Click Save to save the map.</p> <p>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.</p>

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

How to Load the ORDHDR Message

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






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

Step	Action
1	Right-click the ORDERS ORDER DETAILS EDI file icon and select Create Sub from the submenu.
2	From the Create Sub submenu, select Transaction set from standards to display the Select Data Source dialogue box.
3	Select the Machine Data Sources tab.
4	Select Gentran Standards (if you defined your EDI standards data source with another name, select that name) and click OK . System response The system displays the Insert Transaction Set from Standards dialogue box.
	
5	From the Agency list, select [A] Tradacoms as the EDI standard you will use for the inserted message.
6	From the Version list, select [ANA001] TRADACOMS UNITED KINGDOM as the version of the standard you will use. The versions that are available depend on which standard you selected.
7	From the Transaction list, select [ORDHDR] ORDER FILE HEADER as the transaction set (message) you will use. The transactions that are available depend on which standard you selected.
	<i>(Continued on next page)</i>





(Contd) Step	Action
8	From the Release list, select 9 as the message version you will use. The message versions that are available depend on which message (transaction) you selected.
9	Click OK to load the ORDHDR message into the ORDERS EDI file as a subordinate group.

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

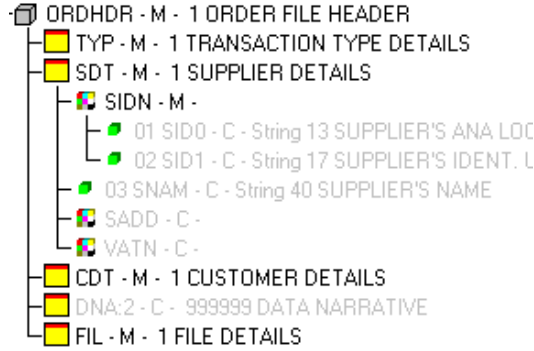
The specific EDI map components that you will use depends on the type of map you create, the standard, version, and transaction set (message) selected, and which segments and elements your company requires. We recommend that you determine which segments and elements you will use before starting this 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 standard 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 grey font (dimmed).

(Continued on next page)

Overview (contd)

In the example below, the ORDHDR group, TYP segment, SDT segment, and the SIDN composite are mandatory, so they are already activated by the system. You also need to use the SDT SIDN SID0 element (a conditional element), so you 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 by the standard, but that you have determined that you need to use in mapping.

Activation

This table lists the groups, segments and elements you must activate. The composite column is provided for your information so you know where to locate the appropriate elements.

Group	Segment	Comp.	Element
ORDHDR (already active)	SDT (already active)	SIDN	SID0
			CNAM
	CDT (already active)	CIDN	CID0
		CADD	CAD0 CAD1 CAD2 CAD3 CAD4
(Continued on next page)			
	CLO (already active)	CLOC	CLO0
			CNAM
		CADD	CAD0 CAD1 CAD2 CAD3 CAD4

Group	Segment	Comp.	Element
	ORD (already active)	ORNO (already active)	ORN0 ORN2
			CD07
	DIN		LDAT
OLD (already active)	OLD:2	SPRO	SPR1
		CPRO	CPR1
		UNOR	UNO0 CD04
		OQTY	OQT0
		OUCT	OUC0
		TDES	TDE0 TDE1

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

Step	Action
1	<p>Double-click the groups, segments, and composites listed in the above 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.</p> <p>Note When you activate a conditional group or segment, the mandatory subordinate segments, composites, and elements are automatically activated.</p>
2	<p>Click the Activate button on the Main Toolbar.</p> <p style="text-align: right;">(Continued on next page)</p>
3	<p>Click each inactive segment that you need to use. This activates the segments.</p> <p>Note If you accidentally click a segment that you did not mean to activate, click the segment with the right mouse button to access the shortcut menu. Select Deactivate from the shortcut menu.</p>

Step	Action
4	<p>Click each inactive element that you need to use. This activates the elements.</p> <p>Note If you accidentally click an element that you did not mean to activate, click the element with the right mouse button to access the shortcut menu. Select Deactivate from the shortcut menu.</p>
5	<p>Once you have activated all the necessary groups, segments, and elements, click the Activate button on the Main Toolbar to turn activation mode off.</p>

How to Verify EDI Delimiters

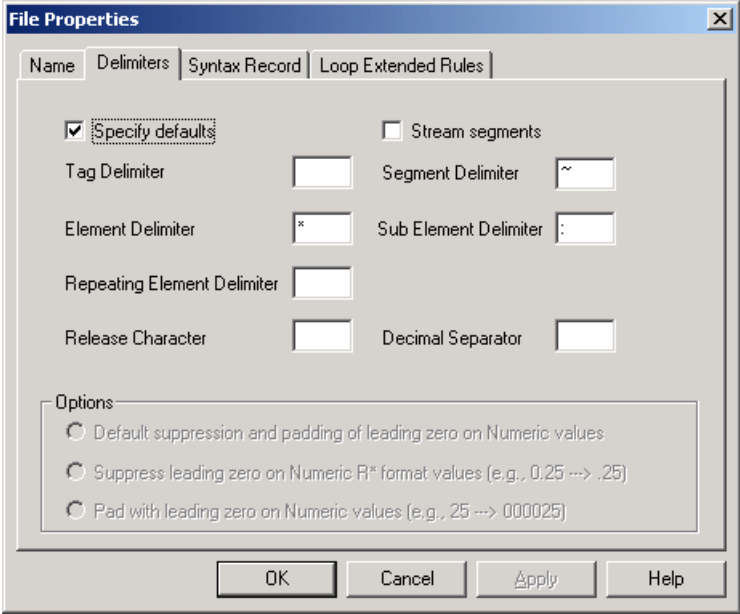
Overview If you are using an EDI standard that contains composite elements or sub-elements, you must verify that Gentran:Server for Windows is specifying the correct delimiters. Delimiters are flags that you define to the system as separating specific EDI components. Delimiters are necessary for all 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	<p>Right-click the ORDERS ORDER DETAILS EDI file icon and from the shortcut menu, select Properties.</p> <p>System response The system displays the File Properties dialogue box.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action																					
2	<p>Select the Delimiters tab.</p> <p>System response The system displays the delimiter settings.</p> 																					
3	Verify that the Specify defaults check box is selected.																					
4	<p>Verify the required delimiters for the TRADACOMS standard. If the delimiters differ from the defaults in this table, type either the character or hexadecimal value in the correct box.</p> <table border="1" data-bbox="638 1289 1398 1766"> <thead> <tr> <th>Field</th> <th>TRADACOMS Default Character</th> <th>TRADACOMS Default Hex Value</th> </tr> </thead> <tbody> <tr> <td>Tag Delimiter</td> <td>= (equal sign)</td> <td>0x3D</td> </tr> <tr> <td>Element Delimiter</td> <td>+ (plus sign)</td> <td>0x2B</td> </tr> <tr> <td>Segment Delimiter</td> <td>' (single quote)</td> <td>0x27</td> </tr> <tr> <td>Sub Element Delimiter</td> <td>: (colon)</td> <td>0x3A</td> </tr> <tr> <td>Release Character</td> <td>? (question mark)</td> <td>0x3F</td> </tr> <tr> <td>Decimal Separator</td> <td>. (full stop)</td> <td>0x2E</td> </tr> </tbody> </table>	Field	TRADACOMS Default Character	TRADACOMS Default Hex Value	Tag Delimiter	= (equal sign)	0x3D	Element Delimiter	+ (plus sign)	0x2B	Segment Delimiter	' (single quote)	0x27	Sub Element Delimiter	: (colon)	0x3A	Release Character	? (question mark)	0x3F	Decimal Separator	. (full stop)	0x2E
Field	TRADACOMS Default Character	TRADACOMS Default Hex Value																				
Tag Delimiter	= (equal sign)	0x3D																				
Element Delimiter	+ (plus sign)	0x2B																				
Segment Delimiter	' (single quote)	0x27																				
Sub Element Delimiter	: (colon)	0x3A																				
Release Character	? (question mark)	0x3F																				
Decimal Separator	. (full stop)	0x2E																				
5	Click OK to exit the File Properties dialogue 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 process the data accurately.

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

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

How to Create the POHEADER Record

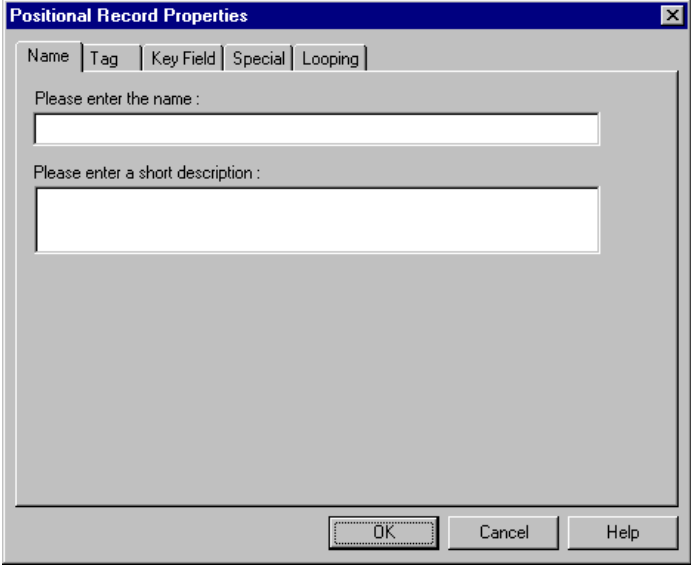
Overview The first record in your application file is usually the header record. The first record will be subordinate to the icon in the application side (output) of the map.

The POHEADER (header) record contains the purchase order type code, shipping and customer information, delivery 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.

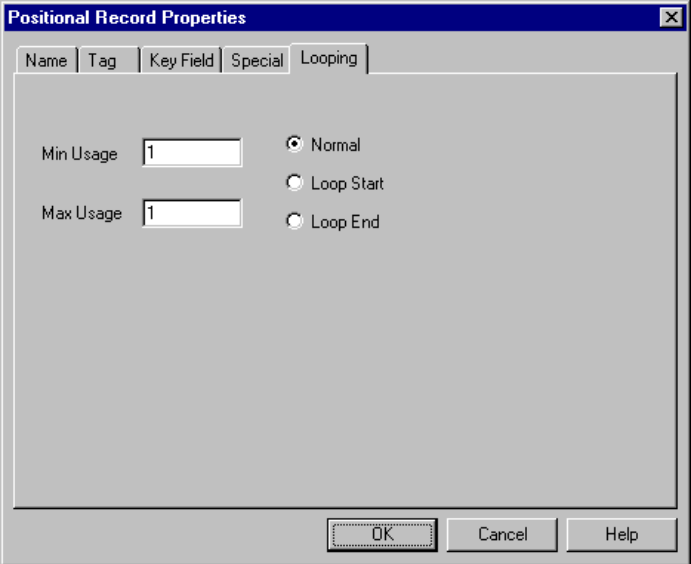
Recommendation

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

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

Step	Action
1	<p>Right-click the Positional File icon to highlight it (Output for an inbound map) and select Create Sub\Record from the submenu.</p> <p>System response The system displays the Positional Record Properties dialogue box.</p> 
2	<p>In the Name box, type “POHEADER” as the record identification code.</p> <p>Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

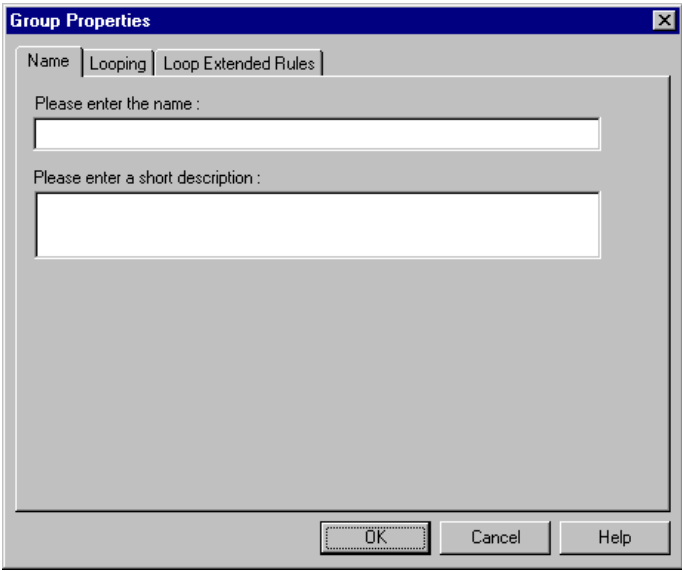
(Contd) Step	Action
3	In the Desc box, type “Header Record” as the description of the record. This box is used to provide a brief explanation of the record, which will allow you to differentiate it from similar records.
4	Select the Tag tab to access tag information.
5	<p>In the Tag box, type “HDR” as the record identification code <TAG>. For example, a record is recognised by the system as the following:</p> <pre><TAG>[Field_1][Field_2]</pre> <p>The record TAG for each record enables the system to recognise that record and then determine the mapping requirements.</p> <p>Recommendation We recommend that you define the record tag on the Positional Record Properties dialogue box instead of defining fields with the purpose of explicitly containing the record tag.</p>
6	In the Position box, accept the default of “1” as the starting column position of the tag in the data record.
7	<p>Select the Looping tab.</p> <p style="text-align: right;">(Continued on next page)</p>

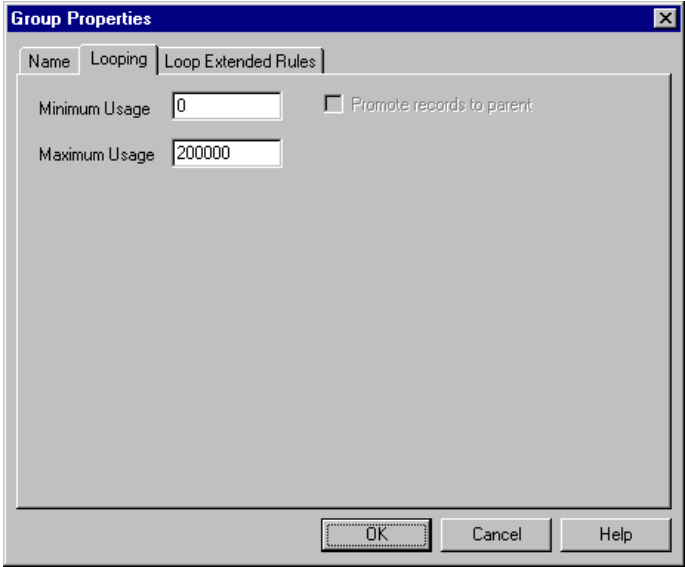
(Contd) Step	Action
8	<p>In the Min Usage box, type "1" to make the header record mandatory. This is the minimum number of times the record must repeat.</p> <p>Note If the Min Usage box contains "0" (zero), the record is conditional. If the Min Usage box contains a "1" or greater, the record is mandatory.</p> <p>If you want the record to repeat (loop), type the maximum number of times it can repeat in the Max Usage box. However, for the header record, we recommend that you always accept the default of "1" so it remains a record that only occurs once (not a looping structure.)</p> <p>The Positional Record Properties dialogue box should now look like this:</p> 
9	Click OK to create the POHEADER record.

How to Create the PODetailGroup

Overview The PODetailGroup (detail information) contains all the line item information for the order, including the item, unit of measure, price, quantity ordered, description, and the allowance and charge information. Each PODetailGroup can occur up to 999,999 times per order translated. This equals a maximum order of 999,999 items. The information within the PODetailGroup is further segregated by related records that repeat in sequence until either the data ends or the maximum number of times that the loop is allowed to repeat is exhausted.

Procedure Use this procedure to create the PODetailGroup at the same level as the POHEADER record.

Step	Action
1	<p>Click the POHEADER record to highlight it. From the Edit menu, select Insert. From the submenu, select Group. The Group Properties dialogue box is displayed.</p> 
2	<p>In the Name box, type “PODetailGroup” to indicate that this group contains all of the detail records. This is the segment loop identification code.</p> <p>Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.</p>
3	<p>In the Description box, type “PO Detail Group” as the loop description. This box is used to provide a brief explanation of the loop.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
4	Select the Looping tab to access looping options.
5	<p>Accept the default of “0” 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.</p> <p>In the Maximum Usage box, type “999999”. This indicates that this group of records can repeat up to 999,999 times. The Group Properties dialogue should now look like this:</p>  <p>The screenshot shows a dialog box titled "Group Properties" with three tabs: "Name", "Looping", and "Loop Extended Rules". The "Looping" tab is selected. It contains two input fields: "Minimum Usage" with the value "0" and "Maximum Usage" with the value "200000". There is also a checkbox labeled "Promote records to parent" which is unchecked. At the bottom of the dialog are three buttons: "OK", "Cancel", and "Help".</p>
6	Click OK to create the PODetailGroup.

Creating the PODetailGroup Subordinate Records

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

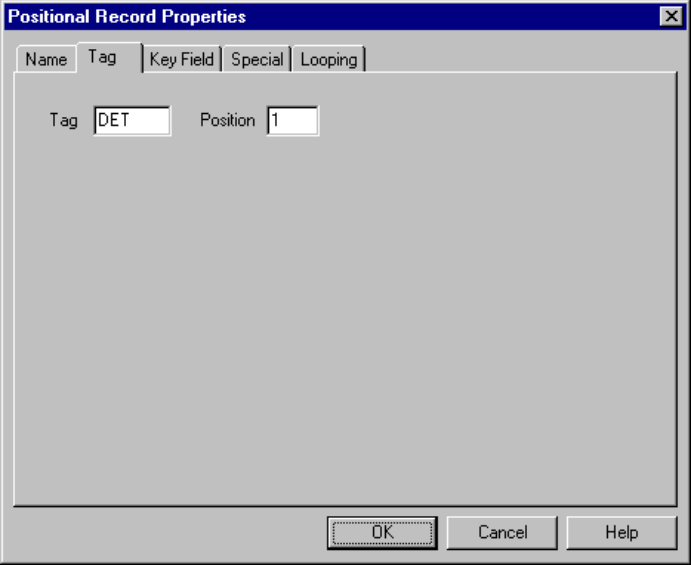
- PODETAIL
- PODESC

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

How to create the PODETAIL record

Use this procedure to create the PODETAIL record.

Step	Action
1	Click the PODetailGroup to highlight it. From the Edit menu, select Create Sub. From the submenu, select Record. The Positional Record Properties dialogue box is displayed.
2	In the Name box, type “PODETAIL” as the record identification code. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.
3	In the Desc box, type “Detail Record” as the description of the record. This box is used to provide a brief explanation of the record, which will allow 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 recognised by the system as the following: <TAG>[Field_1][Field_2] The record TAG for each record enables the system to recognise that record and then determine the mapping requirements. Recommendation We recommend that you define the record tag on the Positional Record Properties dialogue box instead of defining the fields with the purpose of explicitly containing the record tag. (Continued on next page)

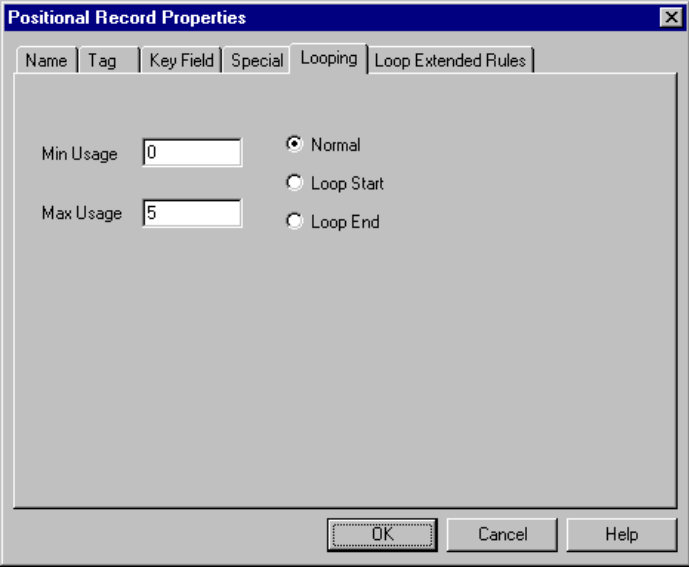
(Contd) Step	Action
6	<p>In the Position box, accept the default of “1” as the starting column position of the tag in the data record. The Positional Record Properties dialogue box should now look like this:</p> 
7	Click OK to create the PODETAIL record.

How to create the PODESC record

Use this procedure to create the PODESC record.

Step	Action
1	Click the PODETAIL to highlight it. From the Edit menu, select Insert. From the submenu, select Record. The Positional Record Properties dialogue box is displayed.
2	<p>In the Name box, type “PODESC” as the record identification code.</p> <p>Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.</p>
3	In the Desc box, type “Description Record” as the description of the record. This box is used to provide a brief explanation of the record, which will allow you to differentiate it from similar records.
4	<p>Select the Tag tab to access tag information.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
5	<p>In the Tag box, type “DES” as the record identification code <TAG>. For example, a record is recognised by the system as the following:</p> <pre><TAG>[Field_1][Field_2]</pre> <p>The record TAG for each record enables the system to recognise that record and then determine the mapping requirements.</p> <p>Recommendation We recommend that you define the record tag on the Positional Record Properties dialogue box instead of defining the fields with the purpose of explicitly containing the record tag.</p>
6	<p>In the Position box, accept the default of “1” as the starting column position of the tag in the data record.</p> <p>The Looping section of the Positional Record Properties dialogue box enables you to specify the minimum and maximum number of times this record can repeat, and whether a record is mandatory or conditional. Alter this section <i>only</i> if the record is a looping structure. Otherwise, accept the defaults.</p>
7	<p>Select the Looping tab.</p> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
8	<p>Accept the default of “0” in the Min Usage box. This is the minimum number of times the record must repeat.</p> <p>Note If the Min Usage box contains “0” (zero), the record is conditional. If the Min Usage box contains “1” or greater, the record is mandatory.</p> <p>In the Max Usage box, type “5” as the maximum number of times the record can repeat. The Positional Record Properties dialogue box should look like this:</p> 
9	Click OK to create the PODESC record.

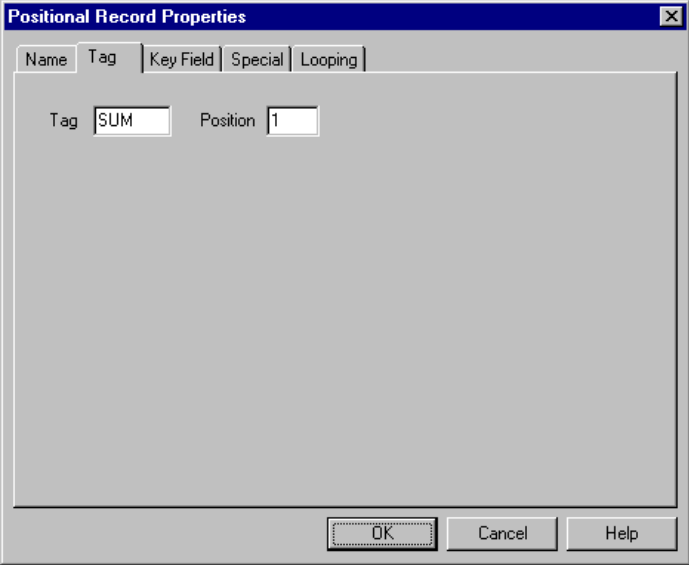
How to Create the POSUMMARY Record

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

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

Procedure Use this procedure to create the POSUMMARY record.

Step	Action
1	<p>Click the PODetailGroup to highlight it. From the Edit menu, select Insert. From the submenu, select Record to display the Positional Record Properties dialogue box.</p> <p>Note Remember to select “Insert” from the Edit menu when you want to create equal records/groups and select “Create Sub” from the Edit menu to create subordinate groups/records/fields.</p>
2	<p>In the Name box, type “POSUMMARY” as the record identification code.</p> <p>Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore () to separate words.</p>
3	<p>In the Desc box, type “Summary Record” as the description of the record. This box is used to provide a brief explanation of the record, which will allow you to differentiate it from similar records.</p>
4	<p>Select the Tag tab to access tag information.</p>
5	<p>In the Tag box, type “SUM” as the record identification code <TAG>. For example, a record is recognised by the system as the following:</p> <pre data-bbox="630 1465 959 1493"><TAG>[Field_1][Field_2]</pre> <p>The record TAG for each record enables the system to recognise that record and then determine the mapping requirements.</p> <p>Recommendation We recommend that you define the record tag on the Positional Record Properties dialogue box instead of defining fields with the purpose of explicitly containing the record tag.</p>
6	<p>In the Position box, accept the default of “1” as the starting column position of the tag in the data record.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
7	Select the Looping tab.
8	<p>Accept the default of “0” in the Min Usage box. This is the minimum number of times the record must repeat.</p> <p>Note If the Min Usage box contains “0” (zero), the record is conditional. If the Min Usage box contains “1” or greater, the record is mandator.</p> <p>Also, accept the default of “1” in the Max Usage box so the summary record only occurs once (not a looping structure). The Positional Record Properties dialogue box should now look like this:</p> 
9	<p>Click OK to create the POSUMMARY record. The application side of the map (output) should now look like this:</p> <pre> OUTPUT - M - 1 POHEADER - M - 1 Header Record PODetailGroup - C - 999999 PO Detail Group PODETAIL - C - 1 Detail Record PODESC - C - 1 Description Record POSUMMARY - C - 1 Summary Record </pre>

Creating the Application Fields

Overview

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

The easiest way to add application fields to a record is to use the Positional Field Editor. You will 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 (DD/MM/YY)	8	8
PONUMHDR	X	Purchase Order Number	String	1	22
POTYPECODE		Purchase Order Type Code	String	0	2
SHIPTONAME		Ship To Name	String	0	35
SHIPTOADD1		Ship To Address Line 1	String	0	35
SHIPTOADD2		Ship To Address Line 2	String	0	35
SHIPTOCITY		Ship To City	String	0	35
SHIPTOSTATE		Ship To State, Province, or Country	String	0	35
SHIPTOPOSTAL CODE		Ship To Postal Code	String	0	9
CUSTNAME		Customer Name	String	0	35
CUSTADD1		Customer Address Line 1	String	0	35
CUSTADD2		Customer Address Line 2	String	0	35
(Continued on next page)					

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
CUSTCITY		Customer City	String	0	35
CUSTSTATE		Customer State, Province, or Country	String	0	35
CUSTPOSTALCODE		Customer Postal Code	String	0	9
DELDATE	X	Delivery Date	Dt/Tm (DD/MM/YY)	8	8
NUMLINEITEMS	X	Number of Line Items	Number (N0)	1	6
PO Detail Group (PODetailGroup) up to 999,999 per message					
Detail Record (PODETAIL) 1 per message					
CUSTNUMDET	X	Customer Number	String	1	8
PODATEDET	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMDET	X	Purchase Order Number	String	1	22
CUSTPROCEDURE	X	Customer Product Code	String	1	30
UPCCODE	X	UPC Code	String	1	30
CONSUMERUNITS		Consumer Units in Trading Unit	Number (N0)	0	15
UOM		Unit of Measure	String	0	7
UNITPR		Unit Price	Number (R2)	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 (DD/MM/YY)	8	8
PONUMDESC	X	Purchase Order Number	String	1	22
DESC		Description	String	0	81
<i>(Continued on next page)</i>					

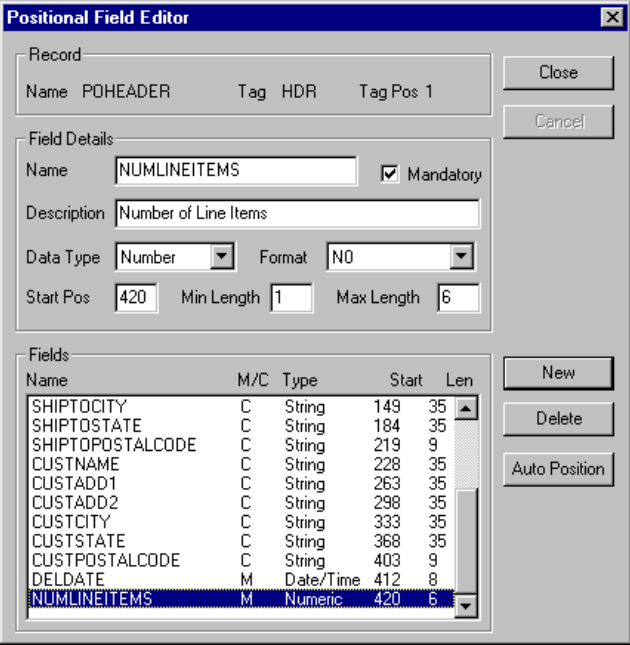
(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Summary Record (POSUMMARY) 1 per message					
CUSTNUMSUM	X	Customer Number	String	1	8
PODATESUM	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMSUM	X	Purchase Order Number	String	1	22
NUMLINEITEM SUM	X	Number of Line Items	Number (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	<p>From the shortcut menu, select Edit Fields to access the Positional Field Editor dialogue box.</p> 
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	<p>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 each field that occurs in multiple records with a suffix that identifies the record that contains it. For example, use the tag “HDR” for the Header record.</p> <p>Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore () to separate words.</p> <p style="text-align: right;">(Continued on next page)</p>

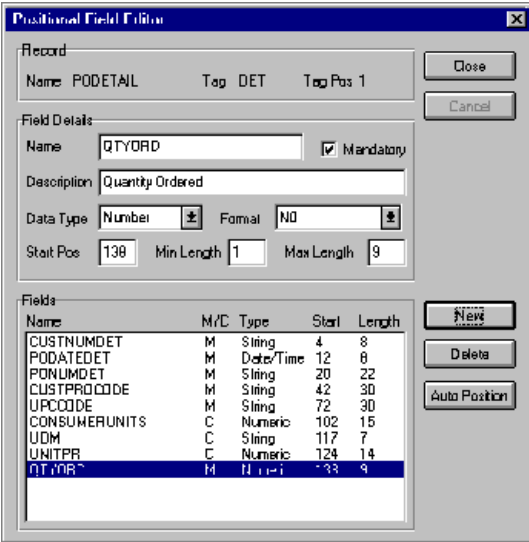
(Contd) Step	Action
5	Select the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system will not be able to identify who sent the purchase order.
6	In the Description box, type “Customer Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7	<p>From the Data Type list, select “String” as the type of field and specify that this field must contain alphanumeric data.</p> <p>Note If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You will want to specify field start positions, for example, if you will only be using a few fields but you want them positioned exactly in the record.</p> <p>In this tutorial, you will use the alternative to specifying the start position of each field, which is to add field sequentially in the record and then use the Auto Position function. Please see Step 13 below for more information about the Auto Position function.</p>
8	If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format list.
9	<p>In the Min Length box, type “1” as the minimum field length.</p> <p>Note If the minimum length of the field is “0”, then accept that default in the Min Length field.</p>
10	In the Max Length box, type “8” as the maximum length of the field.
11	To add the field and create the next field positioned after it, press Enter. A new field with blank values ready for you to identify is positioned below the field.
12	<p>Create the rest of the fields according to the record layout table starting on 2 - 41.</p> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
13	<p>After you add the last field, click Auto Position to automatically position the fields in the record. The Auto Position function will automatically calculate 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 will be sequenced in order.</p> <p>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 dialogue box and if you define every field in the record in the sequence that each field occurs (as you are doing in this tutorial).</p> <p>The Positional Field Editor dialogue box should now look like the following (only the last eleven fields in the record are visible in the Fields list box):</p> 
14	When you finish 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.

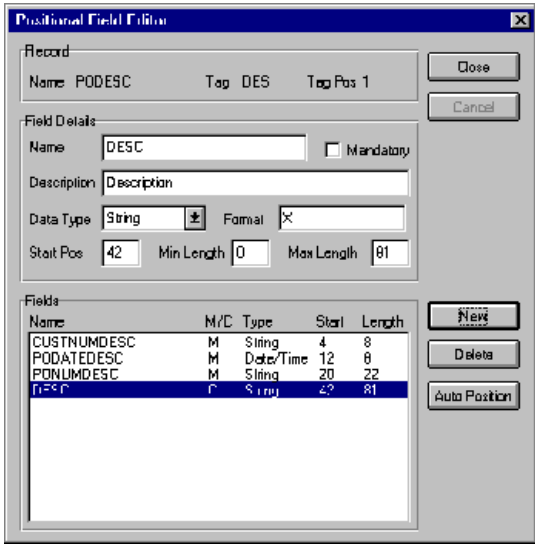
Step	Action
1	Click the PODETAIL application record with the right mouse button to access the shortcut menu.
2	From the shortcut menu, select Edit Fields to access the Positional Field Editor dialogue box.
3	If the field you are creating is the first field in the record, click New . A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the name box, type “CUSTNUMDET”, which is the name of the first field in the PODETAIL record. Each application field must have a unique name. It is useful to tag the end of each field that occurs in multiple records with a suffix that identifies the record that contains it. For example, use the tag “DET” for the Detail record. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.
5	Click the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system will not be able to identify who sent the purchase order.
6	In the Description box, type “Customer Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7	From the Data Type list, select “String” as the type of field and specify that this field must contain alphanumeric data. Note If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You will want to specify field start positions, for example, if you will only be using a few fields but you want them positioned exactly in the record. In this tutorial, you will use the alternative to specifying the start position of each field, which is to add field sequentially in the record and then use the Auto Position function. Please see Step 13 below for more information about the Auto Position function.
8	If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format list. <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
9	In the Min Length box, type “1” as the minimum field length. If the minimum length of the field is “0”, then accept that default in the Min Length box.
10	In the Max Length box, type “8” as the maximum length of the field.
11	To add the field and create the next field positioned after it, press Enter. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to the record layout table starting on 2 - 41.
13	<p>After you add the last field, click Auto Position to automatically position the fields in the record. The Auto Position function will automatically calculate 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 will be sequenced in order.</p> <p>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 dialogue box and if you define every field in the record in the sequence that each field occurs (as you are doing in this tutorial).</p> <p>The Positional Field Editor dialogue box should now look like the following:</p> 
14	When you finish 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 PODETAIL record.

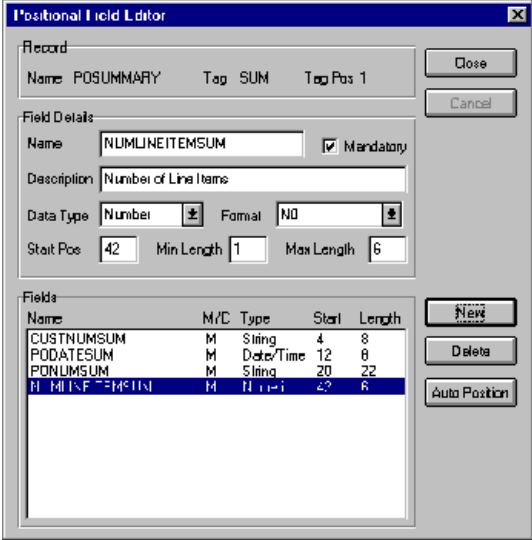
Step	Action
1	Click the PODESC application record with the right mouse button to access the shortcut menu.
2	From the shortcut menu, select Edit Fields to access the Positional Field Editor dialogue box.
3	If the field you are creating is the first field in the record, click New . A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the name box, type “CUSTNUMDESC”, which is the name of the first field in the PODESC record. Each application field must have a unique name. It is useful to tag the end of each field that occurs in multiple records with a suffix that identifies the record that contains it. For example, use the tag “DESC” for the Description record. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.
5	Click the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system will not be able to identify who sent the purchase order.
6	In the Description box, type “Customer Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7	From the Data Type list, select “String” as the type of field and specify that this field must contain alphanumeric data. Note If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You will want to specify field start positions, for example, if you will only be using a few fields but you want them positioned exactly in the record. In this tutorial, you will use the alternative to specifying the start position of each field, which is to add field sequentially in the record and then use the Auto Position function. Please see Step 13 below for more information about the Auto Position function.
8	If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format list. <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
9	In the Min Length box, type “1” as the minimum field length. If the minimum length of the field is “0”, then accept that default in the Min Length box.
10	In the Max Length box, type “8” as the maximum length of the field.
11	To add the field and create the next field positioned after it, press Enter. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to the record layout table starting on 2 - 41.
13	<p>After you add the last field, click Auto Position to automatically position the fields in the record. The Auto Position function will automatically calculate 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 will be sequenced in order.</p> <p>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 dialogue box and if you define every field in the record in the sequence that each field occurs (as you are doing in this tutorial).</p> <p>The Positional Field Editor dialogue box should now look like the following:</p> 
14	When you finish adding fields to the PODESC 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 shortcut menu.
2	From the shortcut menu, select Edit Fields to access the Positional Field Editor dialogue box.
3	If the field you are creating is the first field in the record, click New . A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the name box, type “CUSTNUMSUM”, which is the name of the first field in the POSUMMARY record. Each application field must have a unique name. It is useful to tag the end of each field that occurs in multiple records with a suffix that identifies the record that contains it. For example, use the tag “SUM” for the Summary record. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.
5	Click the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system will not be able to identify who sent the purchase order.
6	In the Description box, type “Customer Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7	From the Data Type list, select “String” as the type of field and specify that this field must contain alphanumeric data. Note If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You will want to specify field start positions, for example, if you will only be using a few fields but you want them positioned exactly in the record. In this tutorial, you will use the alternative to specifying the start position of each field, which is to add field sequentially in the record and then use the Auto Position function. Please see Step 13 below for more information about the Auto Position function.
8	If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format list. <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
9	In the Min Length box, type “1” as the minimum field length. If the minimum length of the field is “0”, then accept that default in the Min Length box.
10	In the Max Length box, type “8” as the maximum length of the field.
11	To add the field and create the next field positioned after it, press Enter. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to the record layout table starting on 2 - 41.
13	<p>After you add the last field, click Auto Position to automatically position the fields in the record. The Auto Position function will automatically calculate 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 will be sequenced in order.</p> <p>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 dialogue box and if you define every field in the record in the sequence that each field occurs (as you are doing in this tutorial).</p> <p>The Positional Field Editor dialogue box should now look like the following:</p> 
14	When you finish adding fields to the POSUMMARY record, click Close .

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

Mapping Your Data

Overview

Introduction

Each record contains a group of logically-related application fields. These fields will contain the information from the EDI file that your system needs to process the purchase order. The group contains logically-related application records.

Each record contains a three-field record key that ensures that the same identifying information is located at the beginning of each record. Your application file format contains a record key so that each record can be identified individually. The record key for this purchase order consists of your partner's customer number, the purchase order date, and the purchase order number.

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

Mapping the POHEADER Record

Overview

Introduction

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

- the purchase order type code
- shipping and customer information
- delivery date information
- the number of line items in the purchase order.

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

CUSTNUMHDR Field

Overview

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

For this map, you need to extract 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 four times in the application file format, once in each record (CUSTNUMHDR, CUSTNUMDET, CUSTNUMDESC, and CUSTNUMSUM).

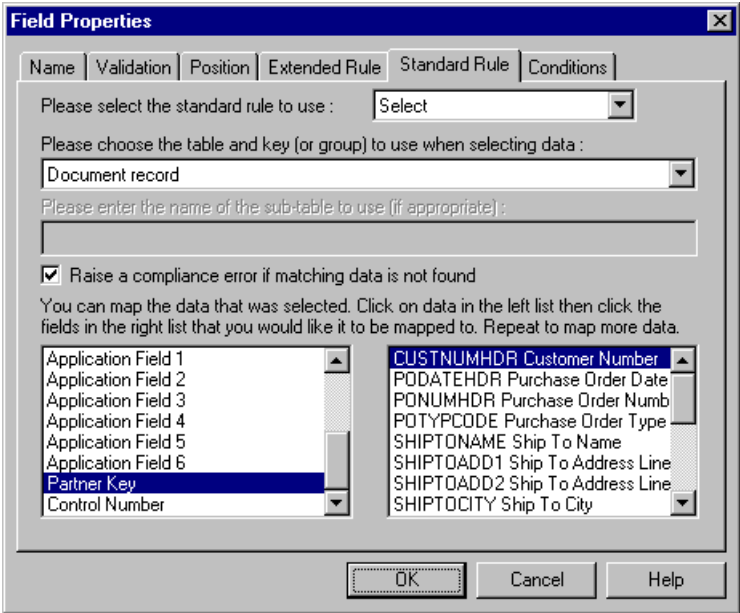
Your partner's customer number is part of the record key for each record in the application file format. The customer number identifies which trading partner sent the purchase order. This customer number is already part of the partner definition for this partner, so you can map it from the partner definition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

How to map the customer number

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 dialogue box.
2	Select the Standard Rule tab to access 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	Select the compliance error check box to indicate that a compliance error is to be generated if the Select function does not find a valid entry.
6	From the Map From list on the left, select "Partner Key." This is the field that contains the customer number for this partner.

(Continued on next page)

(Contd) Step	Action
7	<p>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 dialogue box should now look like this:</p> 
8	Click OK to add the standard rule to the CUSTNUMHDR field.

PODATEHDR Field

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

Procedure Use this procedure to map the date order placed by customer.

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 ORD ORNO ORN2 element (Date Order Placed by Customer). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field PODATEHDR in the POHEADER record. A line is displayed, visually illustrating the link between the ORD ORNO ORN2 element and the PODATEHDR field.

PONUMHDR Field

Overview

You can map the PONUMHDR field directly from the ORD ORNO ORN0 element (Customer's Order Number), by using the Link function. The ORD ORNO ORN0 element can be directly mapped (linked) to the PONUMHDR field in the POHEADER application record.

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. The Document Name 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 will distinguish the messages translated by this map. Then, the Name field of the document browsers that contain this message will contain the data from the selected element or field.

Recommendation

You can choose any element or field as the Document Name, but we strongly recommend that you select an element or field that occurs only once in the message and has a data type of "String". For a purchase order, the document name is typically the customer's order number.

How to map the customer's order number

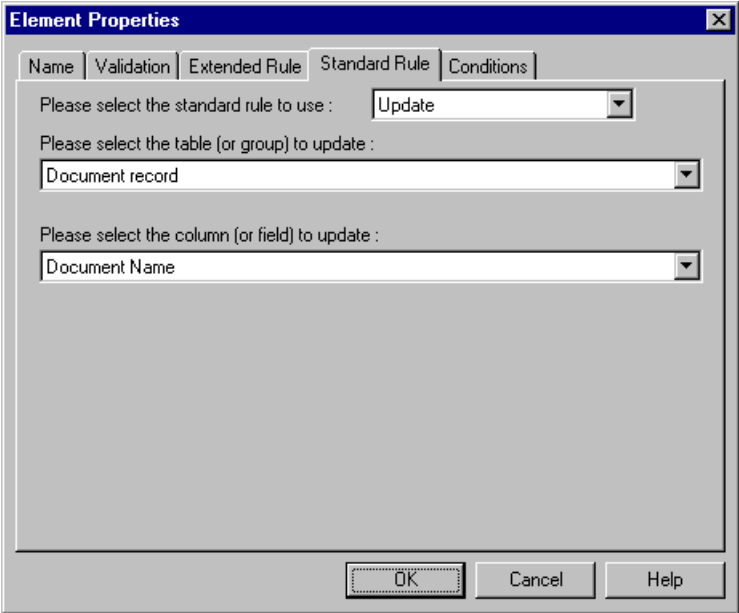
Use this procedure to map the customer's order number.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the ORD ORNO ORN0 element (Customer's Order Number). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field PONUMHDR in the POHEADER record. A line is displayed, visually illustrating the link between the ORD ORNO ORN0 element and the PONUMHDR field.

How to set up the document name

Use this procedure to set up the document name for this purchase order.

Step	Action
1	Double-click the ORD ORNO ORN0 element (Customer's Order Number) to display the Element Properties dialog box.
2	Select the Standard Rule tab to access the standard rule options. (Continued on next page)

Step	Action
3	From the standard rule list, select “Update.”
4	From the table to update list, choose “Document record.” This indicates that you are updating the document record in Gentran:Server for Windows.
5	<p>From the column list, choose “Document Name.” This indicates that you are updating the Name field in the document browsers with the contents of this element.</p> <p>The Standard Rule tab of the dialogue box should now look like this:</p> 
6	Click OK to set up the document name.

POTYPCODE Field

Procedure You can map the POTYPCODE field directly from the ORD CD07 element (Order Code), by using the Link function.

Use this procedure to map the coded message function.

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 ORD CD07 element (Order Code), When you click the element, the cursor changes to a “Link Arrow.”
3	Click the application field POTYPCODE in the POHEADER record. A line is displayed, visually illustrating the link between the ORD CD07 element and the POTYPCODE field.

Mapping the Shipping Information

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

How to map the SHIPTONAME field

Use this procedure to map the ship to name.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the CLO CNAM element (Customer's Name), When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field SHIPTONAME in the POHEADER record. A line is displayed, visually illustrating the link between the CLO CNAM element and the SHIPTONAME application field.

How to map the SHIPTOADD1 field

Use this procedure to map the ship to address line 1.

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 CLO CADD CAD0 element (Customer's Address Line 1), When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field SHIPTOADD1 in the POHEADER record. A line is displayed, visually illustrating the link between the CLO CADD CAD0 element and the SHIPTOADD1 application field.

How to map the SHIPTOADD2 field

Use this procedure to map the ship to address line 2.

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 CLO CADD CAD1 element (Customer's Address Line 2), When you click the element, the cursor changes to a "Link Arrow." (Continued on next page)

Step	Action
3	Click the application field SHIPTOADD2 in the POHEADER record. A line is displayed, visually illustrating the link between the CLO CADD CAD1 element and the SHIPTOADD2 application field.

How to map the SHIPTOCITY field

Use this procedure to map the ship to city.

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 CLO CADD CAD2 element (Customer's Address Line 3), When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field SHIPTOCITY in the POHEADER record. A line is displayed, visually illustrating the link between the CLO CADD CAD2 element and the SHIPTOCITY application field.

How to map the SHIPTOSTATE field

Use this procedure to map the ship to state.

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 CLO CADD CAD3 element (Customer's Address Line 4), When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field SHIPTOSTATE in the POHEADER record. A line is displayed, visually illustrating the link between the CLO CADD CAD3 element and the SHIPTOSTATE application field.

How to map the SHIPTOPOSTAL CODE field

Use this procedure to map the ship to postal code.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the CLO CADD CAD4 element (Customer's Post Code), When you click the element, the cursor changes to a "Link Arrow." (Continued on next page)

Step	Action
3	Click the application field SHIPTOPOSTALCODE in the POHEADER record. A line is displayed, visually illustrating the link between the CLO CADD CAD4 element and the SHIPTOPOSTALCODE application field.

Mapping Customer Information

Overview

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

How to map the CUSTNAME field

Use this procedure to map the customer name.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the CDT CNAM element (Customer's Name). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field CUSTNAME in the POHEADER record. A line is displayed, visually illustrating the link between the CDT CNAM element and the CUSTNAME application field.

How to map the CUSTADD1 field

Use this procedure to map the customer address line 1.

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 CDT CADD CAD0 element (Customer's Address Line 1). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field CUSTADD1 in the POHEADER record. A line is displayed, visually illustrating the link between the CDT CADD CAD0 element and the CUSTADD1 application field.

How to map the CUSTADD2 field

Use this procedure to map the customer address line 2.

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 CDT CADD CAD1 element (Customer's Address Line 2). When you click the element, the cursor changes to a "Link Arrow." (Continued on next page)

Step	Action
3	Click the application field CUSTADD2 in the POHEADER record. A line is displayed, visually illustrating the link between the CDT CADD CAD1 element and the CUSTADD2 application field.

How to map the CUSTCITY field

Use this procedure to map the customer city.

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 CDT CADD CAD2 element (Customer's Address Line 3). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field CUSTCITY in the POHEADER record. A line is displayed, visually illustrating the link between the CDT CADD CAD2 element and the CUSTCITY application field.

How to map the CUSTSTATE field

Use this procedure to map the customer state.

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 CDT CADD CAD3 element (Customer's Address Line 4). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field CUSTSTATE in the POHEADER record. A line is displayed, visually illustrating the link between the CDT CADD CAD3 element and the CUSTSTATE application field.

How to map the CUSTPOSTAL CODE field

Use this procedure to map the customer postal code.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the CDT CADD CAD4 element (Customer's Post Code). When you click the element, the cursor changes to a "Link Arrow." (Continued on next page)

Step	Action
3	Click the application field CUSTPOSTALCODE in the POHEADER record. A line is displayed, visually illustrating the link between the CDT CADD CAD4 element and the CUSTPOSTALCODE application field.

DELDATE Field

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

Procedure Use this procedure to map the delivery date.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the DIN LDAT element (Latest Delivery Date). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field DELDATE in the POHEADER record. A line is displayed, visually illustrating the link between the DIN LDAT element and the DELDATE application field.

NUMLINEITEMS Field

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

Procedure Use this procedure to map the number of line items.

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 OTR LORD element (Lines Ordered). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field NUMLINEITEMS in the POHEADER record. A line is displayed, visually illustrating the link between the OTR LORD element and the NUMLINEITEMS field.

Mapping the PODetailGroup Records: PODETAIL Record

Overview

Introduction

The PODetailGroup (detail information) contains all the line item information for the order, including the item, unit of measure, price, quantity ordered, and the description information. Each PODetailGroup can occur up to 999,999 times per order translated. This equals a maximum order of 999,999 items. The information within the PODetailGroup is further segregated by record.

This section describes the actions necessary for you to map to each field in the PODETAIL (detail) record. The PODETAIL (detail) record contains line item code number, consumer units in trading unit, unit of measure, price, and quantity ordered. The PODETAIL record occurs once per PODetailGroup.

CUSTNUMDET Field

Overview

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

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

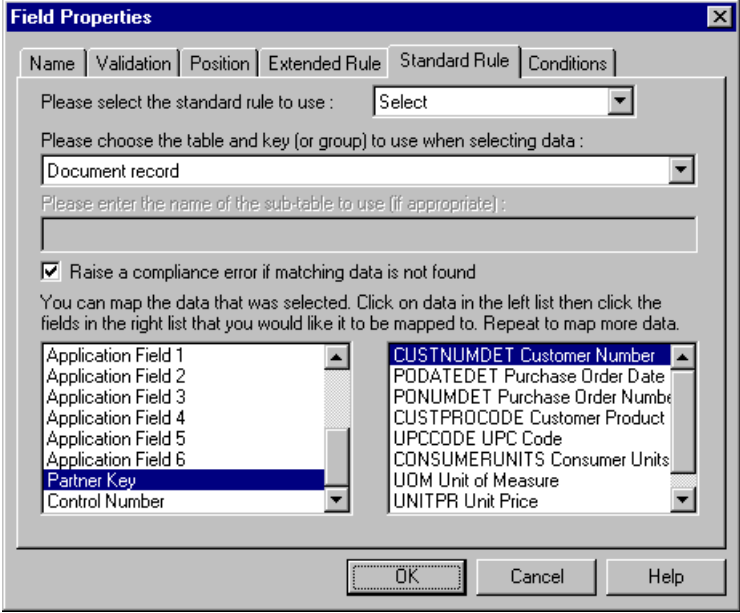
Your partner's customer number is part of the record key for each record in the application. The customer number identifies which trading partner sent the purchase order. This customer number is already part of the partner definition for this partner, so you can map it from the partner definition to 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 dialogue box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, choose "Select."
4	From the Table list, select "Document record." This indicates that the system updates this field with selected information from the document record.
5	Select the compliance error check box to indicate that a compliance error is to be generated if the Select function does not find a valid entry.
6	From the Map From list on the left, choose "Partner Key." This is the field that contains the customer number for this partner.

(Continued on next page)

(Contd) Step	Action
7	<p>From the Map To list on the right, choose “CUSTNUMDET.” This is the field to which the information from the Partner Editor is mapped. The Standard Rule tab of the Field Properties dialogue box should now look like this:</p> 
8	Click OK to add the standard rule to the CUSTNUMDET field.

PODATEDET Field

Overview

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

You can only map fields and elements that are at the same level. Therefore, you cannot map to a group from a segment that does not repeat. For example, you cannot map the date order placed by customer in the ORD segment (which repeats once) directly to your application PODATEDET field (which is included in the PODetailGroup that repeats 999,999 times).

Therefore, to map the purchase order date to this application field, you need to use an extended rule to extract the purchase order date from the ORD ORNO ORD2 element and move it to the PODATEDET field.

How to create an extended rule

Use this procedure to create the extended rule.

Step	Action
1	Double-click the PODATEDET field to access the Field Properties dialogue box.
2	Select the Extended Rule tab to display the extended rule options.
3	In the extended rule list, type the following: #PODATEDET = \$ORDERS.#ORN2; The "\$ORDERS.#ORN2" is the addressing for the ORD ORNO ORD2 element. The system executes this rule when it processes the PODATEDET field.
4	Click Compile to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually so that you can verify the accuracy of the rule after you create it. Note The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process are displayed in the Errors list.
5	Click OK to add the extended rule to the PODATEDET field.

PONUMDET Field

Overview

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

You can only map fields and elements that are at the same level. Therefore, you cannot map to a group from a segment that does not repeat. For example, you cannot map the customer's order number in the ORD segment directly to your application PONUMDET field.

Therefore, to map the customer's order number to this application field, you need to use an extended rule to extract the number from the ORD ORNO ORN0 element and move it to the PONUMDET field.

How to create an extended rule

Use this procedure to create the extended rule.

Step	Action
1	Double-click the PONUMDET field to access the Field Properties dialogue box.
2	Select the Extended Rule tab to display the extended rule options.
3	In the Rule list box, type the following: <pre>#PONUMDET = \$ORDERS.#ORN0;</pre> <p>The "\$ORDERS.#ORN0" is the addressing for the ORD ORNO ORN0 element. The system executes this rule when it processes the PODATEDET field.</p>
4	Click Compile to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually so that you can verify the accuracy of the rule after you create it. Note The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process are displayed in the Errors list.
5	Click OK to add the extended rule to the PONUMDET field.

CUSTPROCEDURE and UPCCODE Fields

Overview

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

For this map, you need to map information from a cross-reference table for the PETZONE3 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 (CUSTPROCEDURE) for each item that is meaningful to Pet Zone. However, your company uses a UPC code (UPCCODE) for each item. After you and your partner determine what the equivalent customer product code is for each UPC code, you can create a cross-reference table in the Partner Editor and use that information in your map.

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

In this map, you have one code (UPC code) for each item ordered. Your partner refers to the same item by a different code (customer product code). You need to map the customer's item code from the OLD:2 CPRO CPR1 element (Customer's Item Code) to the customer product code field (CUSTPROCEDURE) on the application side of the map. Therefore, you must use a cross-reference table to translate the customer product code to your UPC code. You and your partner determined what the equivalent customer product code is for each UPC code. In this tutorial, you the cross-reference table is included with the partner (PETZONE3.PAR) you will import later in this tutorial into Partner Editor.

Cross-Reference table

This table contains the information in the cross-reference table.

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	

Note

You do not need to create this cross-reference table – it is included with the PETZONE3 partner (PETZONE3.PAR) that is provided with the tutorial data when you install Gentran:Server for Windows.

How to map the customer's item code

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

Use this procedure to map the customer's item code.

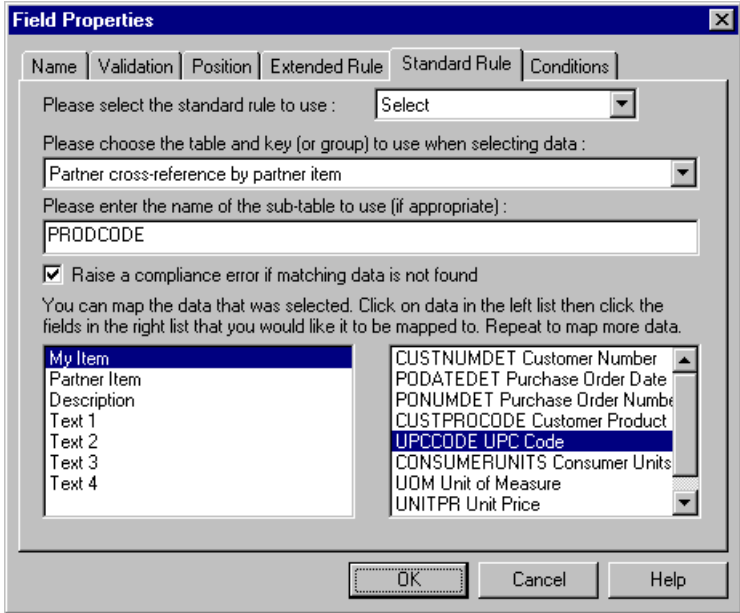
Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the OLD:2 CPRO CPR1 element (Customer's Item Code). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field CUSTPROCEDURE in the PODETAIL record. A line is displayed, visually illustrating the link between the OLD:2 CPRO CPR1 element and the CUSTPROCEDURE field.

How to translate the CUSTPROCEDURE to the 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 CUSTPROCEDURE field (in the PODETAIL record). The Field Properties dialogue box is displayed.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, choose "Select."
4	From the table and key list, select "Partner xref by partner item." This indicates that you will be using a cross-reference table and translating by your partner's item.
5	In the sub-table box, type "PROCEDURE". This is the name of the cross-reference table.
6	Click the compliance error check box to signal the system that if this translation fails, you want the system to generate an error.
7	From the Map From list on the left, select "My Item." This is the field from which the contents will be mapped.

(Continued on next page)

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

CONSUMERUNITS Field

Overview

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

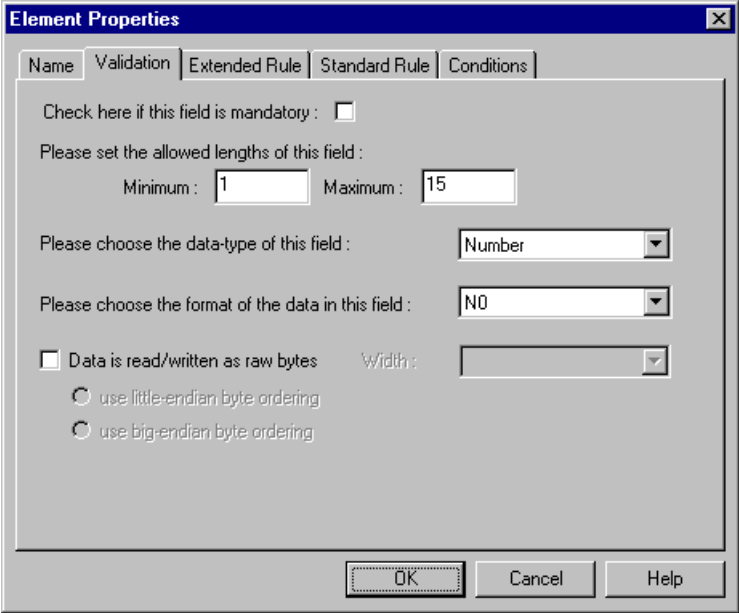
Therefore, you will need to change the properties of the OLD:2 UNOR UNO0 element to reflect the type and the format that you have determined that your partner will send you. Then, you can map the consumer units information directly from the OLD:2 UNOR UNO0 element to the CONSUMERUNITS application field.

How to change the properties of the OLD:2 UNOR UNO0 element

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

Step	Action
1	Double-click the OLD:2 UNOR UNO0 element (Consumer Units in Traded Units) to display the Element Properties dialogue box.
2	Select the Validation tab.
3	From the data-type list, select “Number” to indicate that this is a number element. The system will prompt you with a message that changing the field’s data type invalidates its format. Click Yes to acknowledge this message and proceed.

(Continued on next page)

(Contd) Step	Action
4	<p>From the Format list, select “N0” to indicate that this element will be formatted with an implicit decimal point and no decimal place. The Field Properties dialogue box should now look like this:</p> 
5	Click OK to exit the Element Properties dialogue box.

How to map the consumer units

Use this procedure to map the consumer units.

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 OLD:2 UNOR UNO0 element (Consumer Units in Traded Units). When you click the element, the cursor changes to a “Link Arrow.”
3	<p>Click the application field CONSUMERUNITS in the PODETAIL record.</p> <p>A line is displayed, visually illustrating the link between the OLD:2 UNOR UNO0 element and the CONSUMERUNITS field.</p>

UOM Field

Overview

On the EDI side of the map, the measure indicator information is contained in the OLD:2 UNOR CD04 element (Measure Indicator). You can map the unit of measurement directly from the OLD:2 UNOR CD04 element to the UOM application field.

How to map the UOM field

Use this procedure to map the measure indicator.

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 OLD:2 UNOR CD04 element (Measure Indicator). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field UOM in the PODETAIL record. A line is displayed, visually illustrating the link between the OLD:2 UNOR UNO4 element and the UOM field.

UNITPR Field

Overview

On the EDI side of the map, the cost price information is contained in the OLD:2 OUCT OUC0 element (Cost Price). You can map the unit of measurement directly from the OLD:2 OUCT OUC0 element to the UNITPR application field.

How to map the UNITPR field

Use this procedure to map the unit cost price.

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 OLD:2 OUCT OUC0 element (Cost Price). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field UNITPR in the PODETAIL record. A line is displayed, visually illustrating the link between the OLD:2 OUCT OUC0 element and the UNITPR field.

QTYORD Field

Overview

On the EDI side of the map, the quantity ordered information is contained in the OLD:2 OQTY OQT0 element (Number of Traded Units Ordered). The OLD:2 OQTY OQT0 element is defined as a string-type element in the standard. However, you have determined that your partner will send you a number-type element in N0 format (formatted with an implicit decimal point and no decimal places).

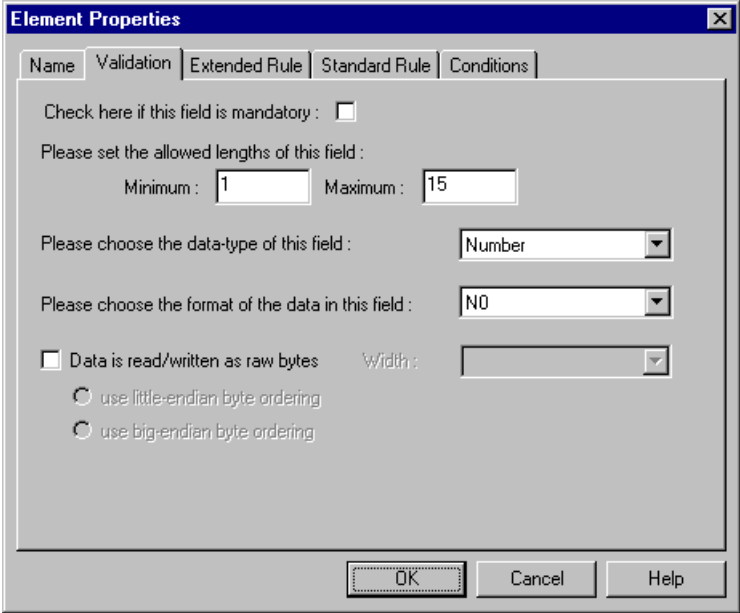
Therefore, you will need to change the properties of the OLD:2 OQTY OQT0 element to reflect the type and the format that you have determined that your partner will send you. Then, you can map the quantity ordered directly from the OLD:2 OQTY OQT0 element to the QTYORD application field.

How to change the properties of the OLD:2 OQTY OQT0 element

Use this procedure to change the type and format of the OLD:2 OQTY OQT0 element.

Step	Action
1	Double-click the OLD:2 OQTY OQT0 element (Number of Traded Units Ordered) to display the Element Properties dialogue box.
2	Select the Validation tab.
3	From the data-type list, select “Number” to indicate that this is a number element. The system will prompt you with a message that changing the field’s data type will invalidate its format. Click Yes to acknowledge this message and proceed.

(Continued on next page)

(Contd) Step	Action
4	<p>From the Format list, select “N0” to indicate that this element will be formatted with an implicit decimal point and no decimal place. The Field Properties dialogue box should now look like this:</p> 
5	Click OK to exit the Element Properties dialogue box.

How to map the quantity ordered

Use this procedure to map the quantity ordered.

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 OLD:2 OQTY OQT0 element (Number of Traded Units Ordered). When you click the element, the cursor changes to a “Link Arrow.”
3	<p>Click the application field QTYORD in the PODETAIL record.</p> <p>A line is displayed, visually illustrating the link between the OLD:2 OQTY OQT0 element and the QTYORD field.</p>

PODESC Record

Overview

Introduction

This section describes the actions necessary for you to map to each field in the PODESC (description) record. The PODESC (description) record contains descriptive information about the line item ordered. The PODESC record can occur up to five times per PODetailGroup, so multiple descriptive terms can be used for each item. For example, if the line item is a “dog collar,” the description fields may be “red,” “large,” and “leather.”

CUSTNUMDESC Field

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

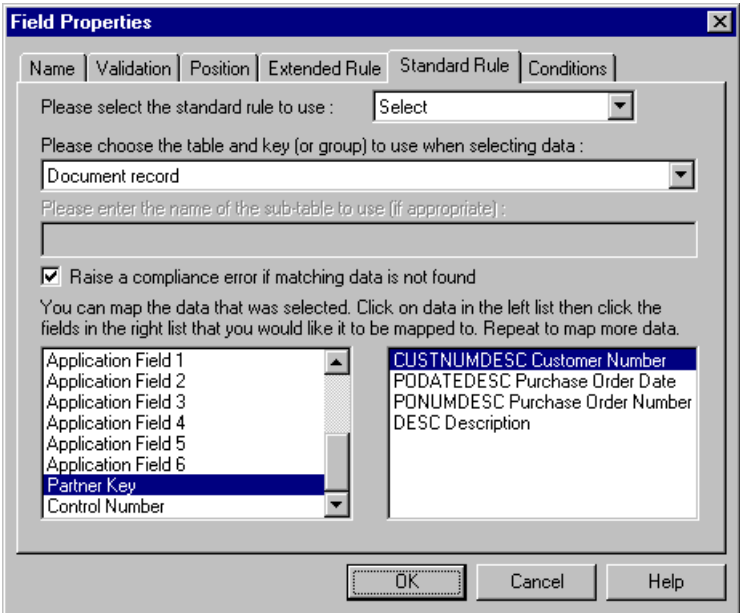
For this map, you need to extract information from the partner definition to populate the Customer Number fields on the application side of the map. Since the Customer Number is a part of the record key for each record, it appears four times in the application file format, once in each record (CUSTNUMHDR,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

Your partner's customer number is part of the record key for each record in the application file format. The customer number 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 dialogue box.
2	Select the Standard Rule tab to access 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	Click the compliance error check box to indicate that a compliance error should be generated if the Select function does not find a valid entry.
6	From the Map From list on the left, select "Partner Key." This is the field that contains the customer number for this partner.

(Continued on next page)

(Contd) Step	Action
7	<p>From the Map To list on the right, select “CUSTNUMDESC.” This is the field to which the information from the Partner Editor is mapped. The Standard Rule tab of the Field Properties dialogue box should now look like this:</p> 
8	Click OK to add the standard rule to the CUSTNUMDESC field.

PODATEDESC Field

Overview

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

You can only map fields and elements that are at the same level. Therefore, you cannot map to a group from a segment that does not repeat. For example, you cannot map the date order placed by customer in the ORD segment (which repeats once) directly to your application PODATEDESC field (which is included in the PODetailGroup that repeats 999,999 times).

Therefore, to map the purchase order date to this application field, you need to use an extended rule to extract the purchase order date from the ORD ORNO ORD2 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 dialogue box.
2	Select the Extended Rule tab.
3	In the Rule box, type the following: <code>#PODATEDESC = \$ORDERS.#ORN2;</code> The “\$ORDERS.#ORN2” is the addressing for the ORD ORNO ORD2 element. The system executes this rule when it processes the PODATEDESC field.
4	Click Compile to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually so that you can verify the accuracy of the rule after you create it. Note The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Errors list.
5	Click OK to add the extended rule to the PODATEDESC field.

PONUMDESC Field

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

You can only map fields and elements that are at the same level. Therefore, you cannot map to a group from a segment that does not repeat. For example, you cannot map the customer's order number in the ORD segment directly to your application PONUMDESC field.

Therefore, to map the customer's order number to this application field, you need to use an extended rule to extract the number from the ORD ORNO ORN0 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 dialogue box.
2	Select the Extended Rule tab.
3	In the Rule box, type the following: <pre>#PONUMDESC = \$ORDERS.#ORN0;</pre> <p>The "\$ORDERS.#ORN0" is the addressing for the ORD ORNO ORN0 element. The system executes this rule when it processes the PONUMDESC field.</p>
4	Click Compile to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually so that you can verify the accuracy of the rule after you create it. Note The Compile function will compile the rule interactively and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process are displayed in the Errors list.
5	Click OK to add the extended rule to the PONUMDESC field.

DESC Field

Overview

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

How to create an extended rule

Use this procedure to create the extended rule.

Step	Action
1	Double-click the DESC field to access the Field Properties dialogue box.
2	Select the Extended Rule tab.
3	<p>In the Rule list, type the following:</p> <pre>#DESC = \$OLD.#TDE0+""+\$OLD.#TDE1;</pre> <p>The "\$OLD.#TDE0" is the addressing for the OLD:2 TDES TDE0 element. The "+" concatenates the character strings. The "" is a blank space to distinguish the two concatenated lines of text. The "\$OLD.#TDE1" is the addressing for the OLD:2 TDES TDE1 element. The system executes this rule when it processes the PONUMDET field.</p>
4	<p>Click Compile to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually so that you can verify the accuracy of the rule after you create it.</p> <p>Note The Compile function will compile the rule interactively and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process are displayed in the Errors list.</p>
5	Click OK to add the extended rule to the PONUMDESC field.

POSUMMARY Record

Overview

Introduction

This section describes the actions necessary for you to map to each field in the POSUMMARY (summary) record. 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.

CUSTNUMSUM Field

Overview

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

For this map, you need to extract 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 four times in the application file format, once in each record (CUSTNUMHDR, CUSTNUMDET, CUSTNUMDESC, and CUSTNUMSUM).

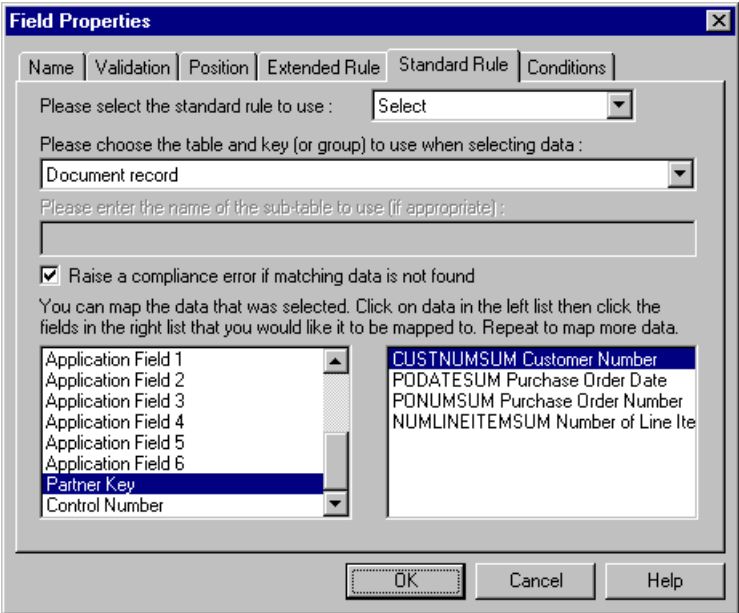
Your partner's customer number is part of the record key for each record in the application file format. The customer number identifies which trading partner sent the purchase order. This customer number is already part of the partner definition for this partner, so you can map it from the partner definition 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 dialogue box.
2	Select the Standard Rule tab to access standard rule options.
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	Click the compliance error check box to indicate that a compliance error should be generated if the Select function does not find a valid entry.
6	From the Map From list on the left, select "Partner Key." This is the field that contains the customer number for this partner.

(Continued on next page)

(Contd) Step	Action
7	<p>From the Map To list on the right, select “CUSTNUMSUM.” This is the field to which the information from the Partner Editor is mapped. The Standard Rule tab of the Field Properties dialogue box should now look like this:</p> 
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 ORD segment. The ORD segment contains order reference information for the header part of the map, and repeats multiple times. You need to map the date order placed by customer (ORD ORNO ORN2) to the PODATE field in each of the application records (PODATEHDR, PODATEDET, PODATEDESC, and PODATESUM). The ORD ORNO ORN2 element is already directly linked to the PODATEHDR field in the POHEADER record. You will also map the ORD ORNO ORN2 element directly to the PODATESUM application field.

Procedure Use this procedure to map the date order placed by customer.

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 ORD ORNO ORN2 element (Date Order Placed By Customer). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field PODATESUM in the POSUMMARY record. A line is displayed, visually illustrating the link between the ORD ORNO ORN2 element and the PODATESUM field.

PONUMSUM Field

Overview You can map the PONUMSUM field directly from the ORD ORNO ORN0 element (Customer's Order Number), by using the Link function. The ORD ORNO ORN0 element is already mapped (linked) to the PONUMHDR field in the POHEADER application record, so you will add an additional link from this element to the PONUMSUM field in the POSUMMARY record.

Procedure Use this procedure to map the customer's order number.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the ORD ORNO ORN0 element (Customer's Order Number). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field PONUMSUM in the POSUMMARY record. A line is displayed, visually illustrating the link between the ORD ORNO ORN0 element and the PONUMSUM field.

NUMLINEITEMSUM Field

Overview You can map the NUMLINEITEMSUM field directly from the OTR LORD element (Lines Ordered), by using the Link function. The OTR LORD element is already mapped (linked) to the NUMLINEITEMS field in the POHEADER application record, so you will be adding an additional link from this element to the NUMLINEITEMSUM field in the POSUMMARY record.

Procedure Use this procedure to map the number of line items.

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 OTR LORD element (Lines Ordered). When you click the element, the cursor changes to a "Link Arrow."
3	Click the application field NUMLINEITEMSUM in the POSUMMARY record. A line is displayed, visually illustrating the link between the OTR LORD element and the NUMLINEITEMSUM field.

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 tick (except the UPCCODE field in the PODETAIL application record). The red tick indicates that a mapping operation has been performed against the field.

Finalising the Map

Overview

Introduction

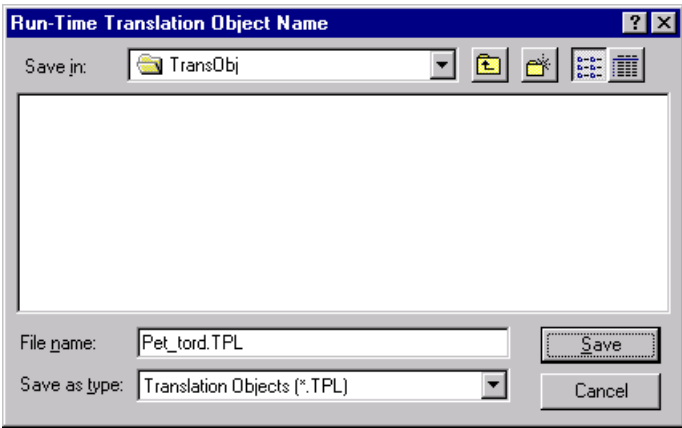
After you create the map, define your application file format, and perform the appropriate mapping operations, you need to finalise the map. To complete the mapping process, you need to save the map, compile the translation object, print and review the Gentran:Server for Windows report, and test the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

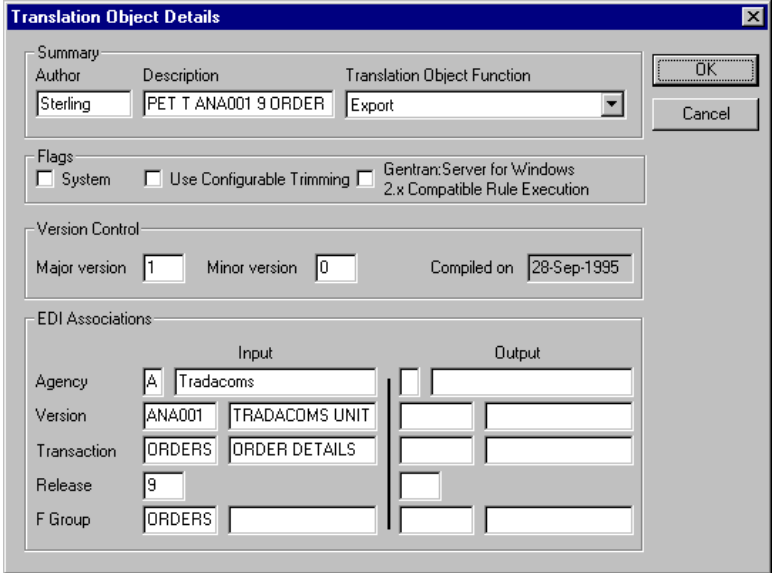
Compiling the Map

Overview The Compile function compiles the map and generates a translation object. The map that you created using Gentran:Server for Windows (PET_TORD.MAP) 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 for Windows system before you can use it. You will use the Compile function after the map is completed and saved.

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

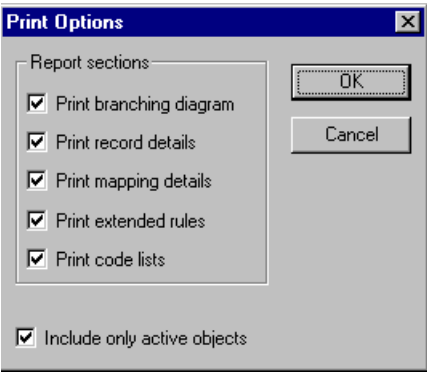
Step	Action
1	From the File menu, select Save to save the source map prior to using the Compile function.
2	<p>From the File menu, select Compile to display the Run-Time Translation Object Name dialogue box.</p> 
3	<p>The File name box will automatically contain “pet_tord.tpl”. This is the name of the translation object, using the default .TPL file extension and our recommendation that you name the translation object (.TPL file) the same file name (one to eight characters long) as you named the map (.MAP file). Preserving the same file name (with different file extensions) means that the relationship between the source map and the compiled translation object remains evident.</p> <p>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.</p> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
4	<p>Change the Drive and Folder where the compiled translation object is stored, if necessary.</p> <p>Caution Do <i>not</i> 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.</p>
5	<p>Click Save and the system will compile the map and generate a translation object. The Compile Errors dialogue box is displayed. Verify that no errors occurred and click OK to exit the dialogue box. The date on which the translation object was compiled will be automatically loaded into the Compiled on box on the Translation Object Details dialogue box.</p>  <p>The screenshot shows the 'Translation Object Details' dialog box. The 'Summary' section includes: Author: Sterling, Description: PET T ANA001 9 ORDER, Translation Object Function: Export. The 'Flags' section has three unchecked checkboxes: System, Use Configurable Trimming, and Gentran:Server for Windows 2.x Compatible Rule Execution. The 'Version Control' section shows Major version: 1, Minor version: 0, and Compiled on: 28-Sep-1995. The 'EDI Associations' section has a table with Input and Output columns. The Input column has fields for Agency (A), Tradacoms, Version (ANA001), TRADACOMS UNIT, Transaction (ORDERS), ORDER DETAILS, Release (9), and F Group (ORDERS). The Output column has empty fields.</p>
6	<p>From the File menu, select Save to save the source map with the Compiled on date.</p>
7	<p>You must register this translation object with the Gentran:Server for Windows system before you can use it.</p> <p>Reference See the <i>Gentran:Server for Windows User's Guide</i> for more information on registering a translation object.</p>

Printing the Report

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

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

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

Testing the Translation Object

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

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

Step	Action
1	Register the PET_TORD.TPL translation object with Gentran:Server for Windows.
2	Import the PETZONE3.PAR partner relationship into Gentran:Server for Windows. The default path for the partner file is: C:\GENSRVTN\TUTORIAL\PETZONE3.PAR
3	Verify (in Partner Editor) that the PET T ANA001 9 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 (PETTSTT1.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:\GENSRVTN\TUTORIAL\PETTSTT1.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)

Contents	Overview.....3 - 4
	<ul style="list-style-type: none"> ▶ Introduction..... 3 - 4 ▶ Getting Started..... 3 - 5 ▶ Analysing Your Application File Format..... 3 - 7 ▶ Analysing the EDI File..... 3 - 12 ▶ Correlating Your Application File and the EDI File..... 3 - 14
	The System Import Map.....3 - 19
	<ul style="list-style-type: none"> ▶ Creating the System Import Map..... 3 - 19 ▶ Creating the HEADER Record..... 3 - 23 ▶ Creating the HEADER Fields..... 3 - 25 ▶ Defining the Key..... 3 - 29 ▶ Compiling the Map..... 3 - 33 ▶ Printing the Report..... 3 - 35 ▶ Testing the Translation Object..... 3 - 36
	The Import Map.....3 - 37
	<ul style="list-style-type: none"> ▶ Creating the Import Map..... 3 - 37 ▶ Activating EDI Map Components..... 3 - 42 ▶ Verifying EDI Delimiters..... 3 - 46
	Defining the Rest of Your Application File Format.....3 - 48
	<ul style="list-style-type: none"> ▶ Overview..... 3 - 48 ▶ Copying the HEADER Record..... 3 - 49 ▶ Creating the INVBEGIN Record..... 3 - 51 ▶ Creating the INVCOMMENT Group..... 3 - 53 ▶ Creating the INVCOMMENT Record..... 3 - 55 ▶ Creating the INVDETAIL Record..... 3 - 57 ▶ Creating the TEMPVATSTL Record..... 3 - 59 ▶ Creating the INVSUMMARY Record..... 3 - 61
	Creating the Application Fields.....3 - 63
	<ul style="list-style-type: none"> ▶ Overview..... 3 - 63

		How to Define the INVBEGIN Record Application Fields	3 - 66
		How to Define the INVCOMMENT Record Application Fields	3 - 69
		How to Define the INVDETAIL Record Application Fields	3 - 71
		How to Define the TEMPVATSTL Record Application Fields	3 - 73
		How to Define the INVSUMMARY Record Application Fields	3 - 75
		Mapping Your Data	3 - 78
		Overview	3 - 78
		Mapping the INVNUMBEG Field	3 - 79
		How to Map the INVDATE Field	3 - 81
		Mapping the PONUM Field	3 - 82
		How to Map the PODATE Field	3 - 86
		Mapping the Shipping Information	3 - 87
		How to Map the ShipTo Fields	3 - 87
		The INVCommentGroup: INVCOMMENT Record	3 - 92
		Overview	3 - 92
		Mapping the MESSAGETEXT Information	3 - 93
		The INVCommentGroup: INVDETAIL Record	3 - 95
		Overview	3 - 95
		How to Map the ITEMNUM Field	3 - 96
		Mapping the QTYSHIP Field	3 - 97
		How to Map the UNITPRICE Field	3 - 98
		How to Map the UOM Field	3 - 99
		How to Map the CUSTPROCEDURE Field	3 - 100
		How to Map the UPCCODE Field	3 - 101
		How to Map the ITEMDESC Field	3 - 102
		Mapping the VATCODE Field	3 - 103
		How to Map the TEMPVATRATE Field	3 - 105
		Mapping the TEMPLEXC Field	3 - 106
		The INVCommentGroup: TEMPVATSTL Record	3 - 112
		Overview	3 - 112
		How to Map the SEQA Field	3 - 113
		How to Map the VATRATE Field	3 - 114
		How to Map the VATRATECODE Field	3 - 115
		How to Map the NUMLINES Field	3 - 116
		How to Map the LVLA Field	3 - 117
		How to Map the VATA Field	3 - 118
		The INVCommentGroup: INVSUMMARY Record	3 - 119
		Overview	3 - 119
		How to Map the TOTALAMOUNT Field	3 - 120
		Producing the VAT Summary	3 - 121
		Producing the Invoice Trailer	3 - 127

Finalising the Map.3 - 130

- ▶ Overview 3 - 130
- ▶ How to Compile the Map 3 - 131
- ▶ How to Print the Report 3 - 133
- ▶ Testing the Translation Object 3 - 134
- ▶ How to Populate the SDT and CDT Segments 3 - 135



Overview

Introduction

In this chapter

This tutorial defines the tasks necessary to create the map used to translate your application file to TRADACOMS 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
- Activating EDI map components
- Verifying EDI delimiters
- Defining the application side of the map
- Mapping your data
- Finalising the map

Note

Press **F1** from any area of the system to access the online Help for that function. The online Help system contains field definitions and other detailed information.

Getting Started

Overview

This tutorial exercise is meant to be completed in its entirety. We recommend that you complete each task sequentially, without referring to the completed map that is provided with the installation program (although you may want to verify your completed map with the model).

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

For outbound translation, you will need to set up a system import map and an import map. Generally, you will create one system import map for each message type, and set up an import map for each partner to whom you are sending 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 will enable your company to translate your application file to the invoices that you will send to Pet Zone. After you ship an order to Pet Zone and generate the corresponding invoice, and once your application file is translated into an EDI invoice, your company can send the invoice to Pet Zone. You will invoice only *one* order per message. You will not be using settlement discounts.

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

Mapping analysis

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

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

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

Analysing Your Application File Format

Overview The first step of mapping analysis is analysing your application file format, because this is probably the component that is the most familiar to you. For an outbound map, your application file contains the data that you need to convert to an EDI file so your system can generate the 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. This enables your company to bill your partner correctly for the goods you shipped to them.

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

System import map application file layout

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

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per message					
PARTNER_ID	X	Partner Identifier	String	1	9
APPL_ID		Application ID	String	0	6
APPL_ALIAS_VALUE		Application Alias Value	String	0	6

**Import map
application file
layout**

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

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per message					
PARTNER_ID	X	Partner Identifier	String	1	9
APPL_ID		Application ID	String	0	6
APPL_ALIAS_VALUE		Application Alias Value	String	0	6
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
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	35
SHIPTOPOSTALCODE		Ship To Postal Code	String	0	8
SHIPTOLOC		Ship To Store Location	String	0	8
BOLNUM		Bill of Lading Number	String	0	12
SHIPDATE		Ship Date	Dt/Tm (YYMMDD)	6	6
TERMSNETDUE		Terms New Due Days	Number (R0)	0	3
TERMSTEXT		Payment Terms Text	String	0	40
<i>(Continued on next page)</i>					

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Comment Group (INVCommentGroup) 10 per message					
Invoice Comment Record (INVCOMMENT) 1 per message					
BILLTOACCTCMT	X	Bill To Acct Number	String	1	9
DOCTYPCMT	X	Document Type	String	1	6
INVNUMCMT	X	Invoice Number	String	1	10
MESSAGETEXT1		Message Text Line 1	String	0	40
MESSAGETEXT2		Message Text Line 2	String	0	40
MESSAGETEXT3		Message Text Line 3	String	0	40
MESSAGETEXT4		Message Text Line 4	String	0	40
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)	Number (R0)	1	6
QTYSHIP	X	Quantity Shipped	Number (R0)	1	5
UNITPRICE	X	Unit Price	Number (R2)	1	7
UOM	X	Unit of Measure	String	1	4
CUSTPROCEDURE	X	Customer Product Code	String	1	10
UPCCODE	X	UPC Code	String	1	12
ITEMDESC		Item Description	String	0	30
VATCODE	X	VAT Rate Code	String	1	2
TEMPVATRATE		Temporary Storage VAT Percentage Rate	String	0	10
TEMPLEXC		Temporary Line Extended Cost	Number (R4)	0	14
<i>(Continued on next page)</i>					

(Contd) Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Temporary Storage VAT Record (TEMPVATSTL) up to 10 per message					
BILLTOACCTTS	X	Bill To Acct Number	String	1	9
DOCTYPTS	X	Document Type	String	1	6
INVNUMTS	X	Invoice Number	String	1	10
SEQA		Sequence Number of STL	Number (R0)	0	10
VATRATE		VAT Rate	Number (R3)	0	6
VATRATECODE		VAT Code	String	0	1
NUMLINES		Number of Lines for VAT Code	Number (R0)	0	10
LVLA		Line Subtotal Amount (Before VAT)	Number (R2)	0	12
VATA		VAT Amount Payable	Number (R2)	0	12
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 (R2)	1	10

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 will contain the information that your system needs to generate the invoice.

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

- your partner's bill to account number
- the document type
- the invoice number

Record/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.
INVCOMMENT (comment record)	Contains the message text. The first 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, product code, and VAT calculation information. The detail record is mandatory and occurs up to 200,000 times per invoice translated (once for each line item invoiced).
TEMPVATSTL (temporary storage for VAT subtotals)	Contains temporary storage fields to enable VAT subtotal calculations. The temporary storage record is conditional and occurs up to 20 times per invoice translated (once for each possible VAT code subtotal).
INVSUMMARY (summary)	Contains the total invoice amount. The summary record is mandatory and occurs once per invoice translated.

Next step

Now that you have analysed your application file, the next step is to determine the requirements of the invoice that you will send to your partner.

Analysing the EDI File

Overview

To analyse the EDI file, you must first determine what you need to send your trading partner and what your partner expects to receive. You and your partner will need to agree on which standard, version, and message you will be using. In this tutorial, we are using a TRADACOMS version ANA001, message INVOIC (invoice), release 9. It is important to know the information that your partner expects you to send within the invoice, and what data will be contained in each element used.

It is very helpful to review the EDI standard that you will be using before analysing the EDI file. Obtain an EDI standard manual for the standard and version that you will use. Standard manuals are available from EDI standard agencies.

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

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.

Map components

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

Group	Segment	Group	Segment	Comp.	Element
	CLO (already active)			CLOC	CLO1
					CNAM
				CADD	CAD0 CAD1 CAD2 CAD3 CAD4
	PYT				PAYT
				PAYY	PAY2
(Continued on next page)					

Group	Segment	Group	Segment	Comp.	Element	
	DNA			GNAR	GNA0 GNA1 GNA2 GNA3	
ODD (already active)	ODD:2			ORNO	ORN0 ORN2	
				DELN	DEL0 DEL1	
			ILD (already active)	ILD:2	SPRO	SPR1
					CPRO	CPR1
					QTYI	QTY0 CD04:2
					AUCT	CD04:3
				TDES	TDE0	

Note

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

You will be invoicing one order *only* per INVOIC message. Therefore, you will need to change the maximum usage for the IRF segment (Invoice References), PYT segment (Settlement Terms), and ODD group (Order and Delivery Terms and Invoice Line Details) to reflect the *one* order that you are invoicing.

Document name and reference

Also, 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. The Document Name 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 that contains data that will distinguish the messages translated by this map. Then, the Name field of the document browsers that contain this message will contain the data from the selected element or field. You can choose any element or field as the Document Name, but we strongly recommend that you select an element or field that occurs only once in the message and has a data type of "String". For an INVOIC, the document name is typically the Invoice Number. Therefore, the document name for the import map in this tutorial will be the invoice number.

Next step

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

Correlating Your Application File and the EDI File

Overview

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

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

- the six-field key
- three-field key

You will have to decide whether you will be using a six-field key or a three-field key for the header record for a system import map. In this tutorial, you will use the three-field key.

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

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

(Continued on next page)

Overview (contd)

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

Recommendation

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

System import map mapping operations

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

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)
APPL_ID	Application ID	Update standard rule to update the Application Field 1 on the document record
APPL_ALIAS_VALUE	Application Alias Value	Update standard rule to update the Application Field 2 on the document record

Import map mapping operations

This table 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
APPL_ID	Application ID	No mapping required
APPL_ALIAS_VALUE	Application Alias Value	No mapping required
<i>(Continued on next page)</i>		

(Contd) Field Name	Description	Mapping Operations
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 IRF INVN
INVDATE	Invoice Date	Link to IRF IVDT and IRF TXDT
PONUM	Purchase Order Number	Link to ODD:2 ORNO ORN0
PODATE	Purchase Order Date	Link to ODD:2 ORNO ORN2
SHIPTONAME	Ship To Name	Link to CLO CNAM
SHIPTOADD1	Ship To Address Line 1	Link to CLO CADD CAD0
SHIPTOADD2	Ship To Address Line2	Link to CLO CADD CAD1
SHIPTOCITY	Ship To City	Link to CLO CADD CAD2
SHIPTOSTATE	Ship To State	Link to CLO CADD CAD3
SHIPTOPOSTALCODE	Ship To Postal Code	Link to CLO CADD CAD4
SHIPTOLOC	Ship To Store Location	Link to CLO CLOC CLO1
BOLNUM	Bill of Lading Number	Link to ODD:2 DELN DEL0
SHIPDATE	Ship Date	Link to ODD:2 DELN DEL1
TERMSNETDUE	Terms New Due Days	Link to PYT PAYY PAY2
TERMSTEXT	Payment Terms Text	Link to PYT PAYT
Comment Group (INVCommentGroup) 10 per message		
Invoice Comment Record (INVCOMMENT) 1 per message		
BILLTOACCTCMT	Bill To Acct Number	No mapping required
DOCTYPCMT	Document Type	No mapping required
INVNUMCMT	Invoice Number	No mapping required
MESSAGETEXT1	Message Text Line 1	Link to DNA GNAR GNA0
MESSAGETEXT2	Message Text Line 2	Link to DNA GNAR GNA1
MESSAGETEXT3	Message Text Line 3	Link to DNA GNAR GNA2
MESSAGETEXT4	Message Text Line 4	Link to DNA GNAR GNA3

(Contd) Field Name	Description	Mapping Operations
Detail Record (INVDETAIL) up to 200,000 per message		
BILLTOACCTDET	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 ILD:2 SEQB
QTYSHIP	Quantity Shipped	Link to ILD:2 QTYI QTY0 and use accum standard rule to calculate extended price
UNITPRICE	Unit Price	Link to ILD:2 AUCT AUC0 and use accum standard rule to calculate extended price
UOM	Unit of Measure	Link to ILD:2 QTYI CD04:2 and ILD:2 AUCT CD04:3
CUSTPROCEDURE	Customer Product Code	Link to ILD:2 CPRO CPR1
UPCCODE	UPC Code	Link to ILD:2 SPRO SPR1
ITEMDESC	Item Description	Link to ILD:2 TDES TDE0
VATCODE	VAT Rate Code	Extract Division lookup from subtable "VATCODES" and Link to ILD:2 CD12
TEMPVATRATE	Temporary Storage VAT Percentage Rate	Standard rule to map Division lookup from subtable "VATCODES"
TEMPLEXC	Temporary Line Extended Cost	Use accum to extract extended price and Link to ILD:2 LEXC
Temporary Storage VAT Record (TEMPVATSTL) up to 10 per message		
BILLTOACCTTS	Bill To Acct Number	No mapping required
DOCTYPTS	Document Type	No mapping required
INVNUMTS	Invoice Number	No mapping required
SEQA	Sequence Number of STL	Link to STL SEQA:7
VATRATE	VAT Rate	Link to STL VATP:2
<i>(Continued on next page)</i>		

(Contd) Field Name	Description	Mapping Operations
VATRATECODE	VAT Code	Link to STL CD12:2
NUMLINES	Number of Lines for VAT Code	Link to STL NRIL
LVLA	Line Subtotal Amount (Before VAT)	Link to STL LVLA, STL EVLA, and STL ASDA
VATA	VAT Amount Payable	Link to STL VATA
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 TLR TPSI

Extended rules

To calculate the required VAT subtotals, summary, and invoice amount you will need to use extended rules on the INPUT group (application file) and on certain elements on the EDI side of the map. These extended rules will be explained in detail later in this tutorial. You will also use a combination of standard rules (accumulators) and direct mapping to complete the required calculations.

Code tables

You will need to use a code table to look up data for the map. If you need to use tables, you must decide whether to create a code table in Gentran:Server for Windows that can only be used with this map, or to add the table to either the partner relationship or the internal system partner in Partner Editor. You will set up the table you use in this tutorial under the internal system partner (Internal System User) so it can be used globally.

In this tutorial, you need to extract information from a lookup table to be inserted into the outbound data. A lookup table is used when you want to supplement data before it is sent to your partner. Please see the Gentran:Server for Windows *User's Guide* for more information on lookup tables.

Note

You do not need to create this lookup table (INVATCO.LKP). It is included with the tutorial data that is provided when you install Gentran:Server for Windows.

Next step

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

The System Import Map

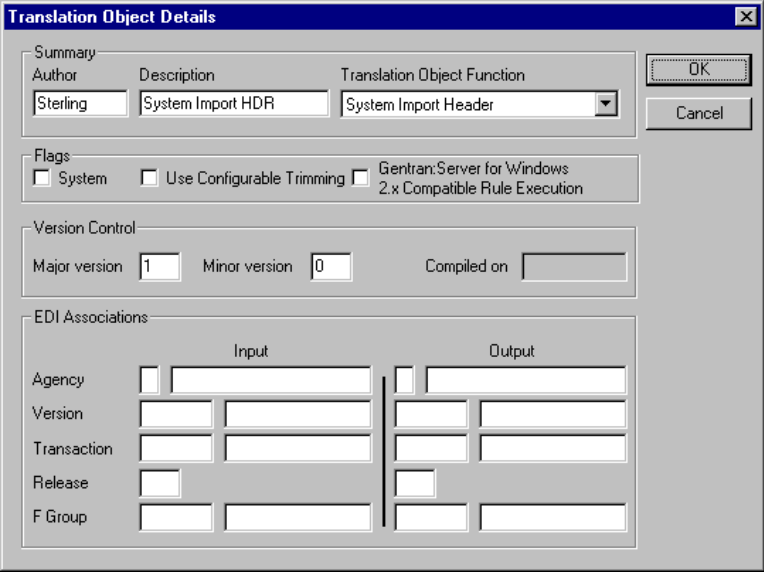
Creating the System Import Map

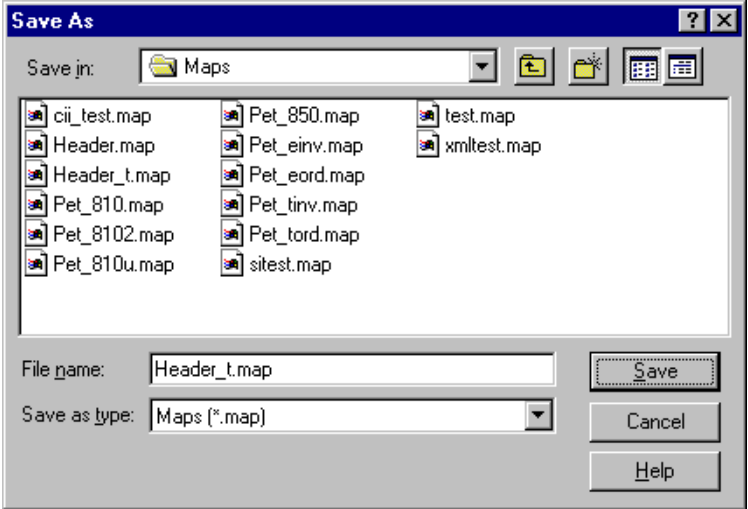
Overview For outbound translation, you will need to set up a system import map and an import map. Generally, you will create one system import map for each message type, and set up an import map for each partner to whom you are sending messages.

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

Procedure Use this procedure to create the system import map.

Step	Action
1	From the File menu, select New . System response The system displays the New Map Wizard.
2	Answer the following questions and then click Next . <ul style="list-style-type: none"> ▶ <i>What kind of map are you creating?</i> Select System Import Header. ▶ <i>What is the name of the map?</i> Type HEADER_T as the unique name of the map. The system adds the .MAP extension. ▶ <i>What is your name?</i> 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 dialogue 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 dialogue box. (Continued on next page)

(Contd) Step	Action
4	<p>From the Create a new data format using this syntax list, select Positional and click Next.</p> <p>System response The system displays the New Map Wizard.</p>
5	<p>Click Finish to create the new map (this may take a few seconds).</p> <p>System response The system displays the new map in the Application Integration Window.</p>
6	<p>From the Edit menu, select Details.</p> <p>System response The system displays the Translation Object Details dialogue box.</p> 
7	<p>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.</p> <p>Reference Please see the <i>Administration Guide</i> for more information about how your system administrator will need to modify the system configuration program.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
8	<p>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.</p> <p>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).</p>
9	Click OK to save the translation object details.
10	<p>From the File menu, select Save.</p> <p>System response The system displays the Save As dialogue box.</p>  <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
11	<p>Verify that the File name box contains Header_t.map.</p> <p>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:</p> <ul style="list-style-type: none"> • 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, <i>but</i> with the file extension .TPL (e.g., HEADER_T.TPL). <p>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.</p> <p>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.</p>
12	<p>Click Save to save the map.</p> <p>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.</p>

Next step

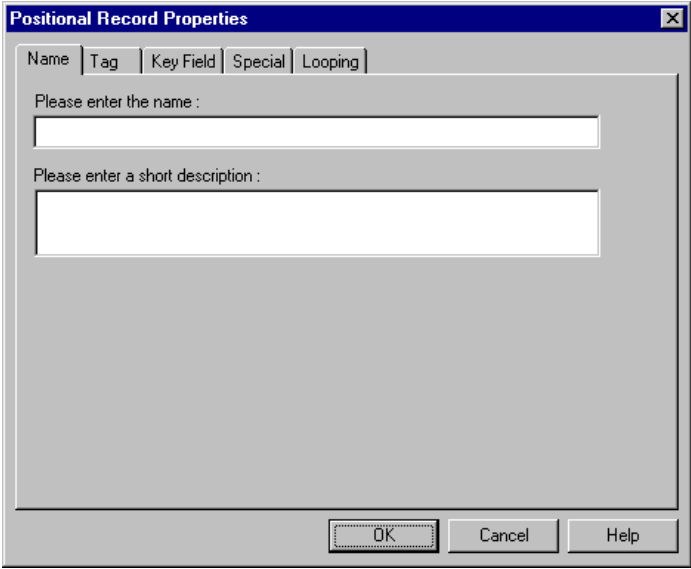
After creating the system import map, you need to define the header record for the system import map. You will use the three-field key in this tutorial.

Creating the HEADER Record

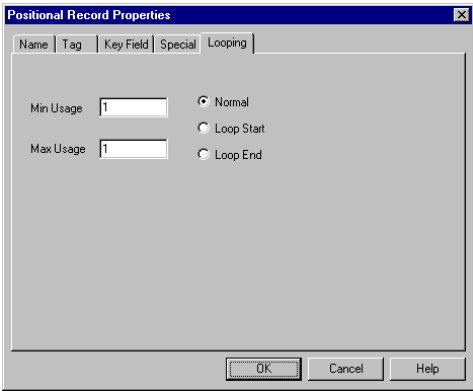
Overview The only record that you will define for the system import map is the header record. The header record will be subordinate to the icon on the input side of the map.

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

Procedure Use this procedure to create the header record (HEADER).

Step	Action
1	<p>Right-click the Positional File icon to highlight it (Output for an inbound map) and select Create Sub\Record from the submenu.</p> <p>System response The system displays the Positional Record Properties dialog box.</p> 
2	<p>In the Name box, type “HEADER” as the record identification code.</p> <p>Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.</p>
3	<p>In the Desc box, type “Header Record” as the description of the record. This box is used to provide a brief explanation of the record, which will allow you to differentiate it from similar records.</p>
4	<p>Select the Tag tab to access tag information.</p> <p style="text-align: right;">(Continued on next page)</p>



(Contd) Step	Action
5	<p>In the Tag box, type “HDR” as the record identification code <TAG>. For example, a record is recognised by the system as the following:</p> <pre><TAG>[Field_1][Field_2]</pre> <p>The record TAG for each record enables the system to recognise that record and then determine the mapping requirements.</p> <p>Recommendation We recommend that you define the record tag on the Positional Record Properties dialogue box instead of defining fields with the purpose of explicitly containing the record tag.</p>
6	<p>In the Position box, accept the default of “1” as the starting column position of the tag in the data record.</p>
7	<p>Select the Looping tab.</p>
8	<p>In the Min Usage box, type”1” to make the header record mandatory. This is the minimum number of times the record must repeat.</p> <p>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.</p> <p>If you want the record to repeat (loop), type the maximum number of times it can repeat in the Max Usage box. However, for the header record, we recommend that you always accept the default of “1” so it remains a record that only occurs once (not a looping structure.)</p> <p>The Positional Record Properties dialogue box should now look like this:</p> 
9	<p>Click OK to create the HEADER record.</p>

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

Creating the HEADER Fields

Overview To use the preferred method of building the system import key, you need to define three fields for the header record (partner key, application identifier, and application alias value). 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 can set up the mapping operations that will define the key that the translator uses to find the partner relationship.

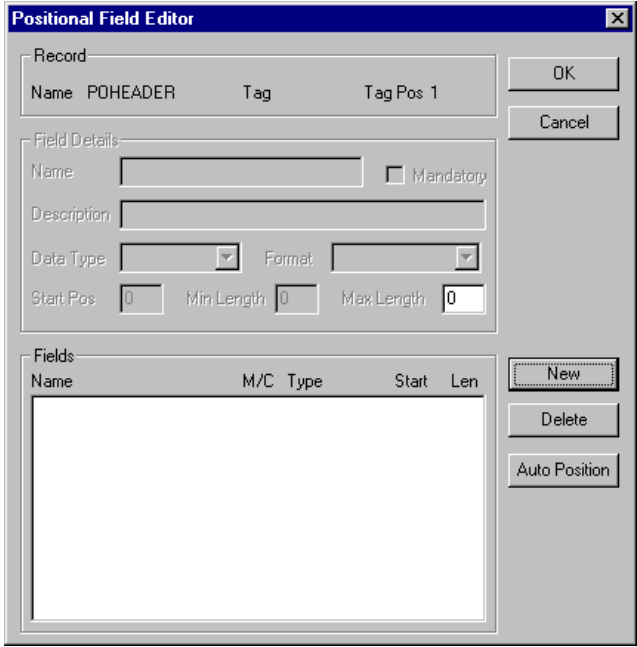
The easiest way to add fields to a record is to use the Positional Field Editor.

This table lists the fields you must create.

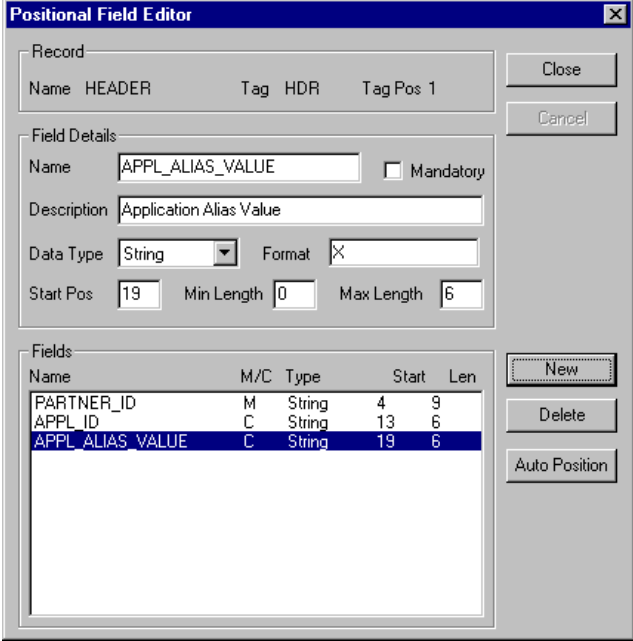
Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per message					
PARTNER_ID	X	Partner Identifier	String	1	9
APPL_ID		Application ID	String	0	6
APPL_ALIAS_VALUE		Application Alias Value	String	0	6

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

Step	Action
1	Right-click the HEADER record to access the shortcut menu. (Continued on next page)

(Contd) Step	Action
2	<p>From the shortcut menu, select Edit Fields to access the Positional Field Editor dialogue box.</p> 
3	<p>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.</p>
4	<p>In the Name box, type “PARTNER_ID”, which is the name of the first field in the HEADER record. Each field must have a unique name.</p> <p>Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore () to separate words.</p>
5	<p>Select the Mandatory check box to identify it as a required field. For the system import map, each field in the key is mandatory.</p>
6	<p>In the Description box, type “Partner Identifier” as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
7	<p>From the Data Type list, select “String” as the type of the field, to specify that this field must contain alphanumeric data.</p> <p>Note If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.</p> <p>In this tutorial, you will use the alternative to specifying the start position of each field; you will 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.</p>
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	<p>Create the rest of the fields according to the record layout table on 3 - 25.</p> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action																				
12	<p>After you add the last field, click Auto Position to automatically position the fields in the record. The Auto Position function will automatically calculate 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 will be sequenced in order.</p> <p>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 dialogue box, and if you define every field in the record in the sequence that each field occurs (as you are doing in this tutorial).</p> <p>The Positional Field Editor dialogue box should now look like this:</p>  <table border="1" data-bbox="659 1136 1122 1409"> <thead> <tr> <th>Name</th> <th>M/C</th> <th>Type</th> <th>Start</th> <th>Len</th> </tr> </thead> <tbody> <tr> <td>PARTNER_ID</td> <td>M</td> <td>String</td> <td>4</td> <td>9</td> </tr> <tr> <td>APPL_ID</td> <td>C</td> <td>String</td> <td>13</td> <td>6</td> </tr> <tr style="background-color: #e0e0ff;"> <td>APPL_ALIAS_VALUE</td> <td>C</td> <td>String</td> <td>19</td> <td>6</td> </tr> </tbody> </table>	Name	M/C	Type	Start	Len	PARTNER_ID	M	String	4	9	APPL_ID	C	String	13	6	APPL_ALIAS_VALUE	C	String	19	6
Name	M/C	Type	Start	Len																	
PARTNER_ID	M	String	4	9																	
APPL_ID	C	String	13	6																	
APPL_ALIAS_VALUE	C	String	19	6																	
13	When you are finished adding fields to the HEADER record, click Close .																				

Defining the Key

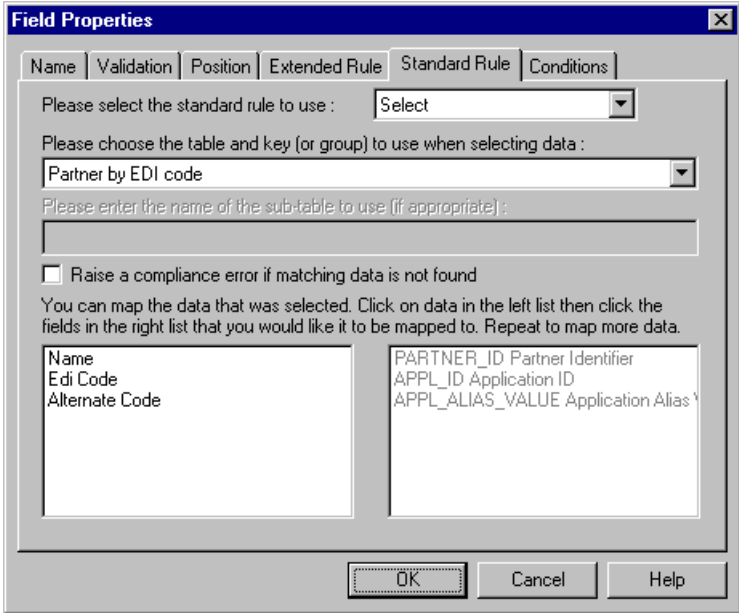
Overview After you define the header record and fields, you can set up the mapping operations that will define the key that the translator uses to find the partner relationship.

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 dialogue box. <div data-bbox="641 779 1377 1388" style="border: 1px solid black; padding: 5px; margin: 10px 0;"> </div>
3	Select the Standard Rule tab to access standard rule options.
4	From the standard rule list, choose “Select.”

(Continued on next page)

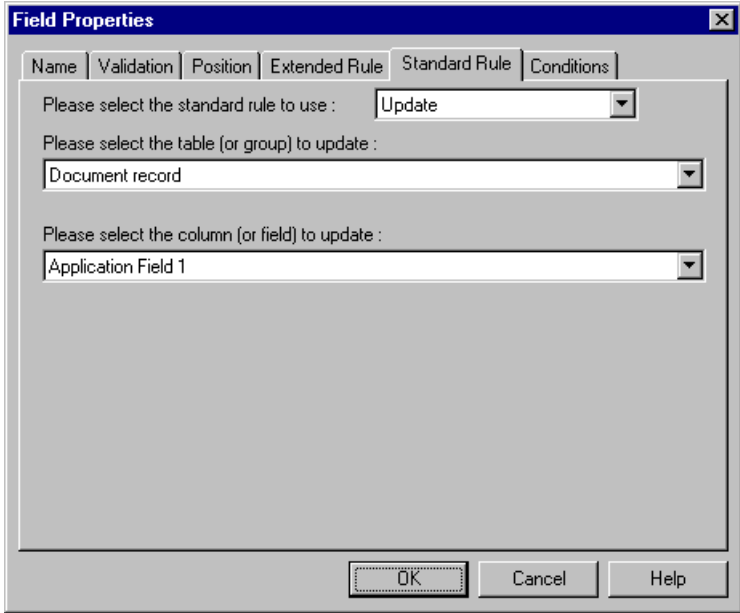


(Contd) Step	Action
5	<p>From the table and key list, select “Partner by EDI Code” (you could also select “Partner by Alternate code,” or “Partner by Partner Key”). This indicates that the system will look up the indicated partner key in the partner database and update this field.</p> <p>The Standard Rule tab of the Field Properties dialogue box should now look like this:</p> 
6	Click OK to add the standard rule to the partner key field.

How to define the application ID field

Use this procedure to define the application ID field.

Step	Action
1	Double-click the application ID field to display the Field Properties dialogue box.
2	Select the Standard Rule tab to access the standard rule options.
3	From the standard rule list, select “Update.”
4	<p>From the table to update list, choose “Document record.” This indicates that you are updating the document record in Gentran:Server for Windows.</p> <p style="text-align: right;">(Continued on next page)</p>

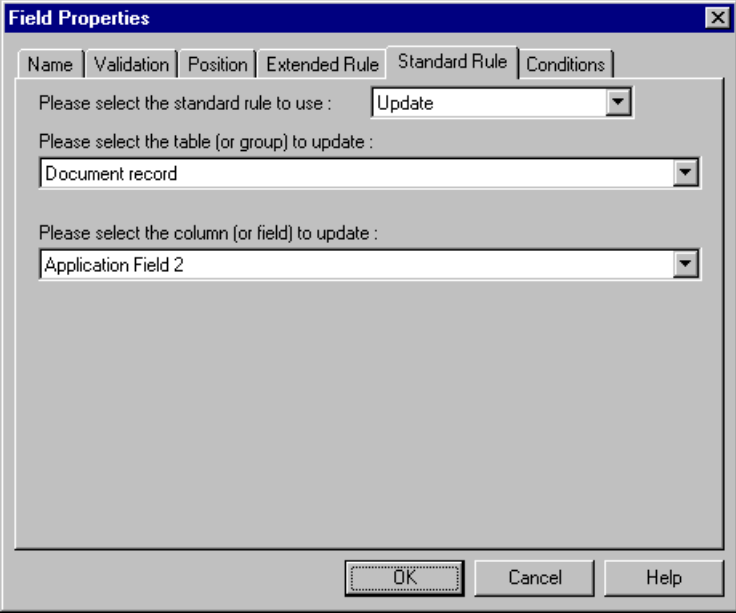
(Contd) Step	Action
5	<p>From the column list, choose “Application Field 1.” This indicates that you are updating the first application field in the system buffer.</p> <p>The Standard Rule tab of the Field Properties dialogue box should now look like this:</p> 
6	Click OK to add the standard rule to the application ID field.

How to define the application alias value field

Use this procedure to define the application alias value field.

Step	Action
1	Double-click the application alias value field to display the Field Properties dialogue box.
2	Select the Standard Rule tab to access the standard rule options.
3	From the standard rule list, select “Update.”
4	<p>From the table to update list, choose “Document record.” This indicates that you are updating the document record in Gentran:Server for Windows.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>



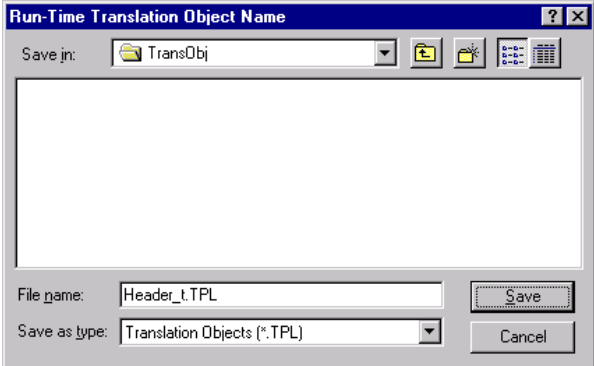
(Contd) Step	Action
5	<p>From the Field list, select “Application Field 2.” This indicates that you are updating the second application field in the system buffer.</p> <p>The Standard Rule tab of the Field Properties dialogue box should now look like this:</p> 
6	Click OK to add the standard rule to the application alias value field.

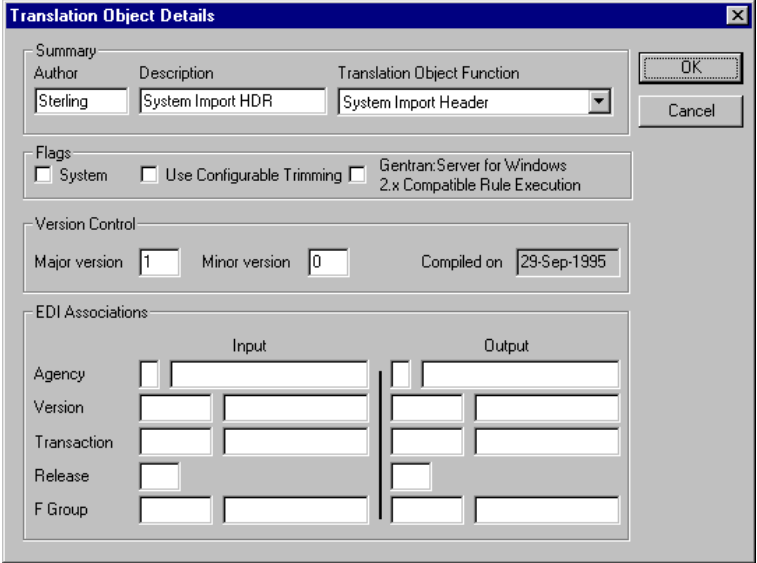
Compiling the Map

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

You must register this translation object with the Gentran:Server for Windows system before you can use it. You will use the Compile function after the map is completed and saved.

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

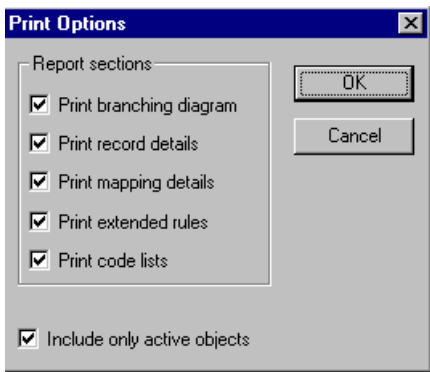
Step	Action
1	From the File menu, select Save to save the source map prior to using the Compile function.
2	From the File menu, select Compile to display the Run-Time Translation Object Name dialogue box. 
3	The File name box contains “Header_t.tpl”. This is the name of the translation object, using the default .TPL file extension. We recommend that you name the translation object (.TPL file) the same file name (one to eight characters long) as you named the map (.MAP file). Preserving the same file name (with different file extensions) means that the relationship between the source map and the compiled translation object remains evident. <p>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.</p>
4	Change the Drive and Folder where the compiled translation object is stored, if necessary. <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
5	<p>Click Save and the system compiles the map and generates a translation object. The Compile Errors dialogue box is displayed. Verify that no errors occurred and click OK to exit the dialogue box. The date on which the translation object as compiled will be automatically loaded into the Compiled on box on the Translation Object Details dialogue box.</p> 
6	<p>From the File menu, select Save to save the source map with the Compiled on date.</p>
7	<p>You must register this translation object with the Gentran:Server for Windows system before you can use it.</p> <p>Reference Please see the Gentran:Server for Windows documentation for information on registering a translation object.</p>

Printing the Report

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

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

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

Testing the Translation Object

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

Reference

Test the system import translation object (HEADER_T.TPL) when you test the import map (please see 3 - 134).

The Import Map

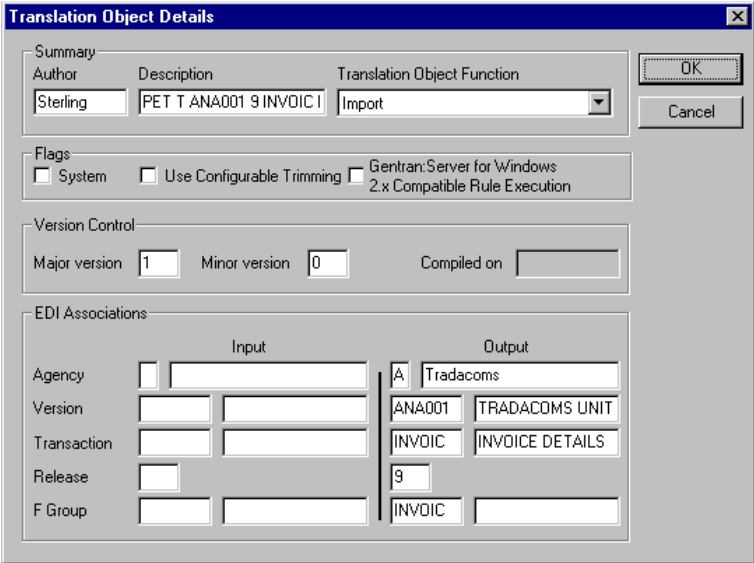
Creating the Import Map

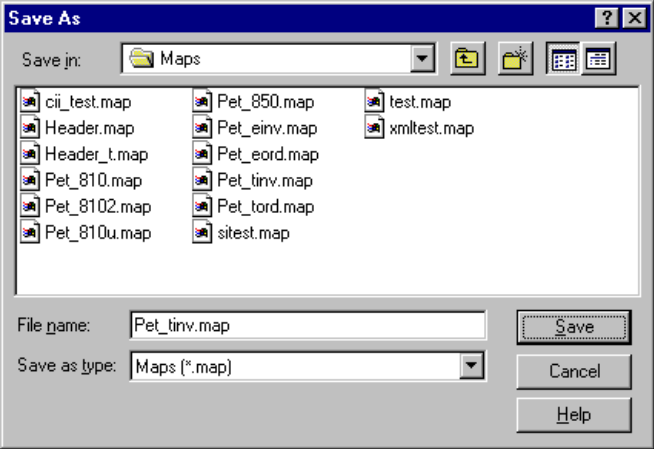
Overview The first task in creating a new map is to define the format that will be used for the input and output sides of the map. In this example, you are creating an import map so Gentran:Server for Windows can translate your application file to an invoice that you are sending to your partner. Therefore, the input side of the map needs to reflect your application file. The output side of the map needs to reflect the TRADACOMS ANA001 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 . <ul style="list-style-type: none"> ▶ <i>What kind of map are you creating?</i> Select Import. ▶ <i>What is the name of the map?</i> Type PET_TINV as the unique name of the map. The system adds the .MAP extension. ▶ <i>What is your name?</i> 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 dialogue 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 dialogue box. (Continued on next page)

(Contd) Step	Action
4	<p>From the Create a new data format using this syntax list, select Delimited EDI and click Customize.</p> <p>System response The system displays the New Delimited EDI Wizard.</p>
5	Click Next .
6	<p>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.</p>
7	<p>Select the following:</p> <ol style="list-style-type: none"> Select the [A] Tradacoms standards agency, [ANA001 TRADACOMS UNITED KINGDOM] version, [INVOIC] INVOICE DETAILS transaction set, and 9 release (for TRADACOMS only). Click Next.
7	<p>Click Finish.</p> <p>System response The system loads the selected transaction set and returns to the New Map Wizard - Output Format dialogue box.</p>
7	<p>Click Next.</p> <p>System response The system displays the New Map Wizard.</p>
6	<p>Click Finish to create the new map (this may take a few seconds).</p> <p>System response The system displays the new map in the Application Integration Window.</p> <p>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.</p>
7	<p>From the Edit menu, select Details.</p> <p>System response The system displays the Translation Object Details dialogue box.</p>
8	<p>In the Description field, type PET T ANA001 9 INVOICE Import.</p> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
9	<p>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.</p> <p>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).</p> <p>System response The Translation Object Details dialogue box should now look like this:</p> 
10	<p>Click OK to save the translation object details.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
11	<p>From the File menu, select Save.</p> <p>System response The system displays the Save As dialogue box.</p> 
12	<p>Verify that the File name box contains pet_tinv.map.</p> <p>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:</p> <ul style="list-style-type: none"> • The source map (map created in Gentran:Server for Windows) should be named using the file extension .MAP (e.g., PET_TINV.MAP). • The compiled translation object (translation object that is the result of compiling the source map) should be named using the same filename that was allocated to the source map, <i>but</i> with the file extension .TPL (e.g., PET_TINV.TPL). <p>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.</p> <p>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.</p> <p style="text-align: right;">(Continued on next page)</p>






(Contd) Step	Action
13	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 need to define the Input and Output sides of the map.

Activating EDI Map Components

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

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

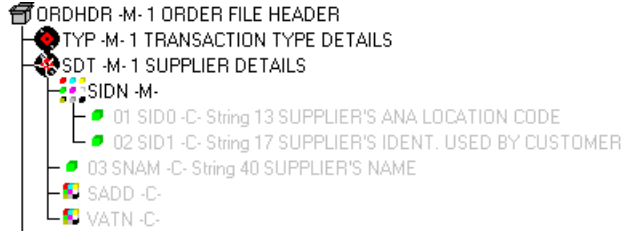
The specific EDI map components that you use depends on the type of map you are creating, the standard, version, and transaction set (message) selected, and which segments and elements your company requires. We recommend that you determine what segments and elements you will use before starting this 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 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 grey font (dimmed).

(Continued on next page)

Overview (contd)

In the example below, the ORDHDR group, TYP segment, SDT segment, and the SIDN composite are mandatory, so they are already activated by the system. You also need to use the SDT SIDN SID0 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 by the standard, but that you have determined that you need to use in mapping.

Required conditional segments and elements

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

Group	Segment	Group	Segment	Comp.	Element
	CLO (already active)			CLOC	CLO1
					CNAM
				CADD	CAD0 CAD1 CAD2 CAD3 CAD4
	PYT				PAYT
				PAYY	PAY2
	DNA			GNAR	GNA0
					GNA1
					GNA2
					GNA3
(Continued on next page)					

(Contd) Group	Segment	Group	Segment	Comp.	Element	
ODD (already active)	ODD:2			ORNO	ORN0 ORN2	
				DELN	DEL0 DEL1	
			ILD (already active)	ILD:2	SPRO	SPR1
					CPRO	CPR1
					QTYI	QTY0 CD04:2
					AUCT	CD04:3
					TDES	TDE0

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

Step	Action
1	<p>Double-click the groups, segments, and composites listed in the table above to open them. These are the segments and elements that you need to activate. The composite column is provided for your information so you will know where to locate the elements. 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.</p> <p>Note When you activate a group or segment, the mandatory subordinate segments and elements are automatically activated.</p>
2	Select the Activate button on the Main Toolbar.
3	<p>Click each inactive segment that you need to use. This activates the segments.</p> <p>Note If you accidentally click an element that you did not mean to activate, click the element with the right mouse button to access the shortcut menu. Select Deactivate from the shortcut menu.</p> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
4	Click each inactive element that you need to use. This activates the elements. Note If you accidentally click an element that you did not mean to activate, click the element with the right mouse button to access the shortcut menu. Select Deactivate from the shortcut menu.
5	After you activate all the necessary groups, segments, and elements, Select the Activate button on the Main Toolbar to turn the activation mode off.

Verifying EDI Delimiters

Overview

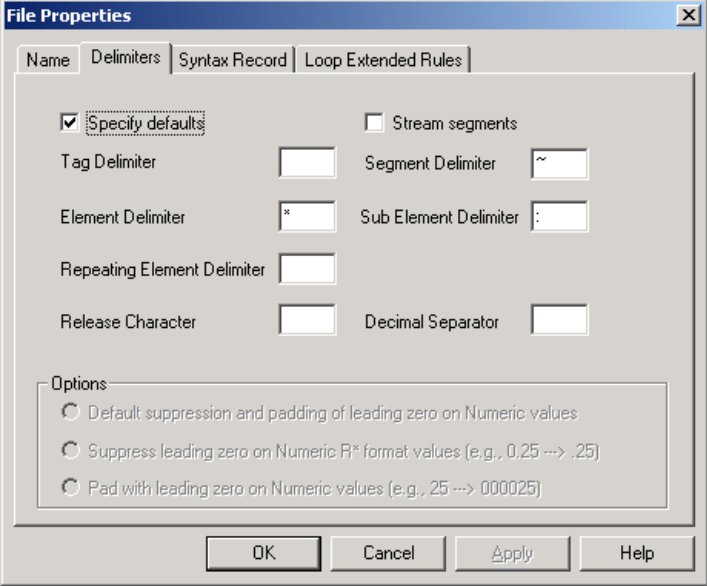
If you are using an EDI standard that contains composite elements or sub-elements, you must verify that Gentran:Server for Windows is specifying the correct delimiters. Delimiters are flags that you define to the system as separating specific EDI components. Delimiters are necessary for all variable 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.

Caution

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	<p>Right-click the INVOIC INVOICE DETAILS EDI file icon and from the shortcut menu, select Properties.</p> <p>System response The system displays the File Properties dialogue box.</p>
2	<p>Select the Delimiters tab.</p> <p>System response The system displays the delimiter settings.</p>  <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action																					
3	Verify that the Specify defaults check box is selected.																					
4	<p>Verify the required delimiters for the TRADACOMS standard. If the delimiters differ from the defaults in this table, type either the character or hexadecimal value in the correct box.</p> <table border="1" data-bbox="638 506 1398 947"> <thead> <tr> <th data-bbox="638 506 938 596">Field</th> <th data-bbox="938 506 1190 596">TRADACOMS Default Character</th> <th data-bbox="1190 506 1398 596">Default Hex Value</th> </tr> </thead> <tbody> <tr> <td data-bbox="638 596 938 655">Tag Delimiter</td> <td data-bbox="938 596 1190 655">= (equal sign)</td> <td data-bbox="1190 596 1398 655">0x3D</td> </tr> <tr> <td data-bbox="638 655 938 714">Element Delimiter</td> <td data-bbox="938 655 1190 714">+ (plus sign)</td> <td data-bbox="1190 655 1398 714">0x2B</td> </tr> <tr> <td data-bbox="638 714 938 772">Segment Delimiter</td> <td data-bbox="938 714 1190 772">' (single quote)</td> <td data-bbox="1190 714 1398 772">0x27</td> </tr> <tr> <td data-bbox="638 772 938 831">Sub Element Delimiter</td> <td data-bbox="938 772 1190 831">: (colon)</td> <td data-bbox="1190 772 1398 831">0x3A</td> </tr> <tr> <td data-bbox="638 831 938 890">Release Character</td> <td data-bbox="938 831 1190 890">? (question mark)</td> <td data-bbox="1190 831 1398 890">0x3F</td> </tr> <tr> <td data-bbox="638 890 938 947">Decimal Separator</td> <td data-bbox="938 890 1190 947">. (full stop)</td> <td data-bbox="1190 890 1398 947">0x2E</td> </tr> </tbody> </table>	Field	TRADACOMS Default Character	Default Hex Value	Tag Delimiter	= (equal sign)	0x3D	Element Delimiter	+ (plus sign)	0x2B	Segment Delimiter	' (single quote)	0x27	Sub Element Delimiter	: (colon)	0x3A	Release Character	? (question mark)	0x3F	Decimal Separator	. (full stop)	0x2E
Field	TRADACOMS Default Character	Default Hex Value																				
Tag Delimiter	= (equal sign)	0x3D																				
Element Delimiter	+ (plus sign)	0x2B																				
Segment Delimiter	' (single quote)	0x27																				
Sub Element Delimiter	: (colon)	0x3A																				
Release Character	? (question mark)	0x3F																				
Decimal Separator	. (full stop)	0x2E																				
5	Click OK to exit the File Properties dialogue box.																					

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

Defining the Rest of Your Application File Format

Overview

Introduction

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

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

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

Copying the HEADER Record

Overview

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

How to copy the HEADER record

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

Step	Action
1	Open the HEADER.MAP system import map if it is not already open.
2	Highlight the HEADER record.
3	Click Copy on the Main Toolbar.
4	Open the PET_TINV.MAP import map if it is not already open.
5	Highlight the INPUT positional file icon that you want the copied record pasted <i>after</i> .
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.

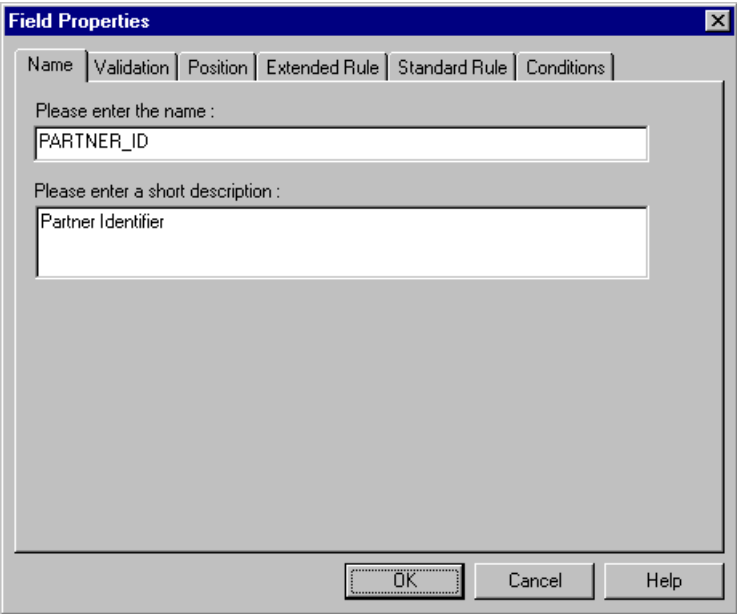
How to remove the HEADER standard rules

The HEADER record that you just copied from the system import map contains three fields. The first field (PARTNER_ID) has a Select standard rule applied to it in the system import map. The remaining two fields in the copied HEADER record in the import map (APPL_ID and APPL_ALIAS_VALUE) have Update standard rules applied to them in the system import map. These standard rules are not applicable to the import map, so you will need to remove them.

(Continued on next page)

How to remove the HEADER standard rules (contd)

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

Step	Action
1	Double-click the PARTNER_ID field in the HEADER record to display the Field Properties dialogue box. 
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select <none> to remove the standard rule.
4	The system prompts you to confirm the removal of the standard rule from the field. Click Yes to remove the standard rule.
5	Click OK to save the change to the field.
6	Repeat steps 1 - 5 for the APPL_ID and APPL_ALIAS_VALUE fields.

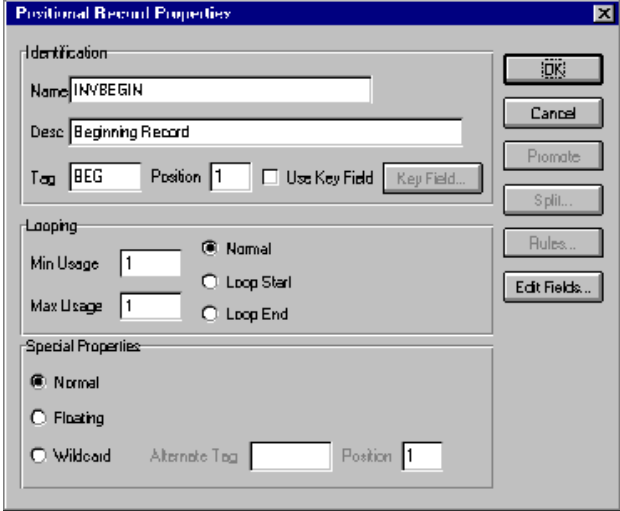
Creating the INVBEGIN Record

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

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

Procedure Use this procedure to create the INVBEGIN record.

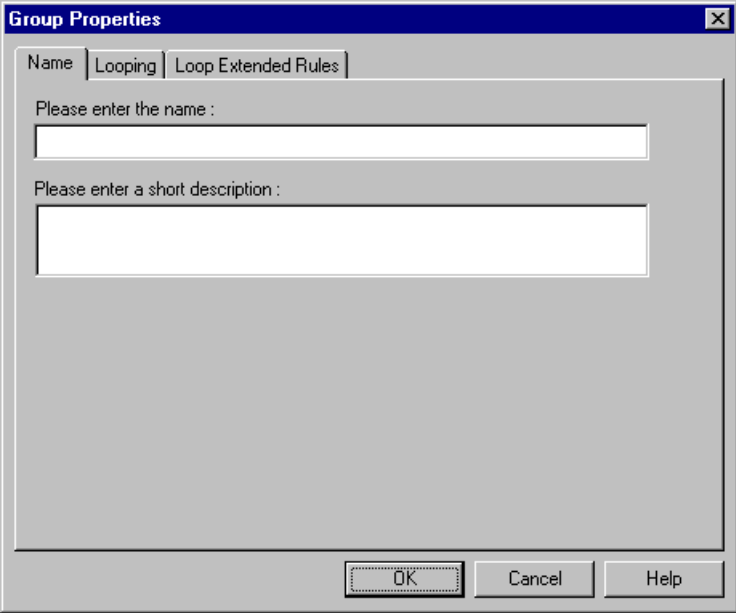
Step	Action
1	Highlight the HEADER record. From the Edit menu, select Insert. From the submenu, select Record to display the Positional Record Properties dialogue box. Note Select "Insert" from the Edit menu when you want to create equal records/groups and "Create Sub" from the Edit menu to create subordinate groups/records/fields.
2	In the Name box, type "INVBEGIN" as the record identification code. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.
3	In the Desc box, type "Beginning Record" as the description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the Tag tab to access tag information.
5	In the Tag box, type "BEG" as the record identification code <TAG>. For example, a record is recognised by the system as the following: <pre data-bbox="630 1465 959 1493"><TAG>[Field_1][Field_2]</pre> The record TAG for each record enables the system to recognise that record and then determine the mapping requirements. Note You must define the record tag on the Positional Record Properties dialogue 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. <div style="text-align: right; color: green;">(Continued on next page)</div>

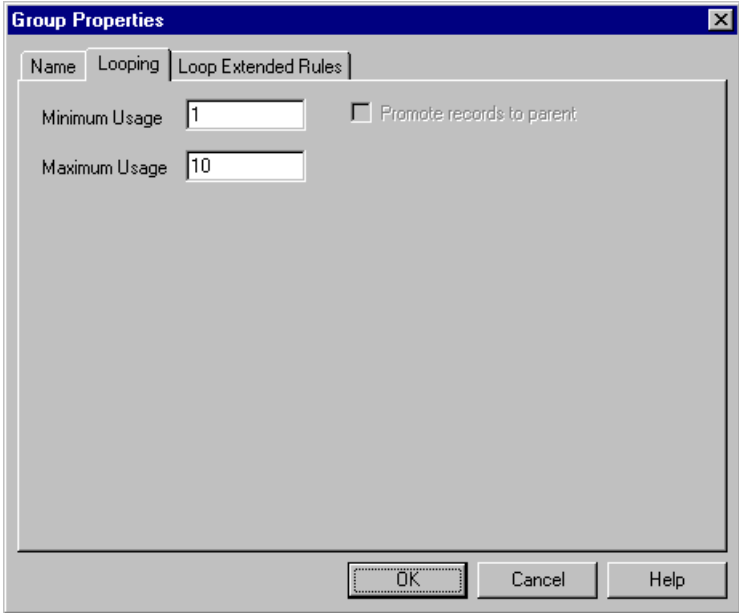
(Contd) Step	Action
7	Select the Looping tab.
8	<p>In the Min Usage box, type “1” to make the record mandatory. This box contains the minimum number of times a record must repeat.</p> <p>Note If the Min Usage box contains “0” (zero), the record is conditional. If the Min Usage box contains “1” or greater, the record is mandatory.</p> <p>Accept the default of “1” in the Max Usage box so the beginning record only occurs once (not a looping structure). The Positional Record Properties dialogue box should now look like this:</p> 
9	Click OK to create the INVBEGIN record.

Creating the INVCOMMENT Group

Overview The INVCommentGroup (comment information) groups all the comment information for the invoice, including the message text. The INVCommentGroup is mandatory and occurs up to ten times per invoice translated.

Procedure Use this procedure to create the INVCommentGroup at the same level as the INVBEGIN record.

Step	Action
1	<p>Highlight the INVBEGIN record. From the Edit menu, select Insert. From the submenu, select Group. The Group Properties dialogue box is displayed.</p> 
2	<p>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.</p> <p>Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.</p>
3	<p>In the Description box, type “Comment Group” as the loop description. This box is used to provide a brief explanation of the loop.</p>
4	<p>Select the Looping tab to access looping options.</p> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
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 always be "1" or greater.
6	<p>In the Maximum Usage box, type "10". This indicates that this group of records will occur up to ten times. The Group Properties dialogue box should now look like this:</p> 
7	Click OK to create the INVCommentGroup.

Creating the INVCOMMENT Record

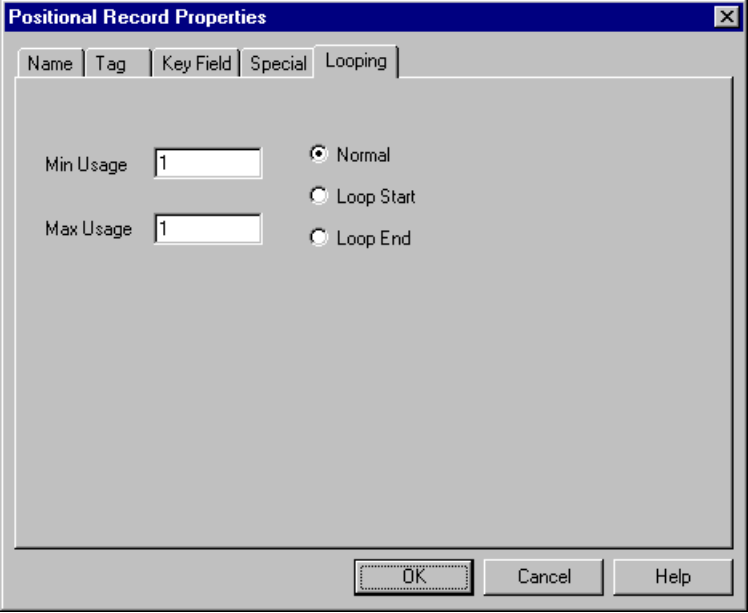
Overview In this tutorial, the INVCommentGroup contains one subordinate record (INVCOMMENT1).

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

Procedure Use this procedure to create the INVCOMMENT1 record.

Step	Action
1	Click the INVCommentGroup to highlight it and select Create Sub from the Edit menu. From the submenu, select Record. The Positional Record Properties dialogue box is displayed.
2	In the Name box, type “INVCOMMENT” as the record identification code. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore () to separate words.
3	In the Desc box, type “Invoice Comment Record 1” as the description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the Tag tab to access tag information.
5	In the Tag box, type “CMT” as the record identification code <TAG>. For example, a record is recognised by the system as the following: <TAG>[Field_1] [Field_2] The record TAG for each record enables the system to recognise that record and then determine the mapping requirements. Note You must define the record tag on the Positional Record Properties dialogue 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.
7	Select the Looping tab.

(Continued on next page)

(Contd) Step	Action
8	<p>In the Min Usage box, type “1” to make the record mandatory. This box contains the minimum number of times the record must repeat.</p> <p>Note If the Min Usage box contains “0” (zero), the record is conditional. If the Min Usage box contains “1” or greater, the record is mandatory.</p> <p>Accept the default of “1” in the Max Usage box so this record only occurs once (not a looping structure). The Positional Record Properties dialogue box should now look like this:</p> 
9	Click OK to create the INVCOMMENT record.

Creating the INVDETAIL Record

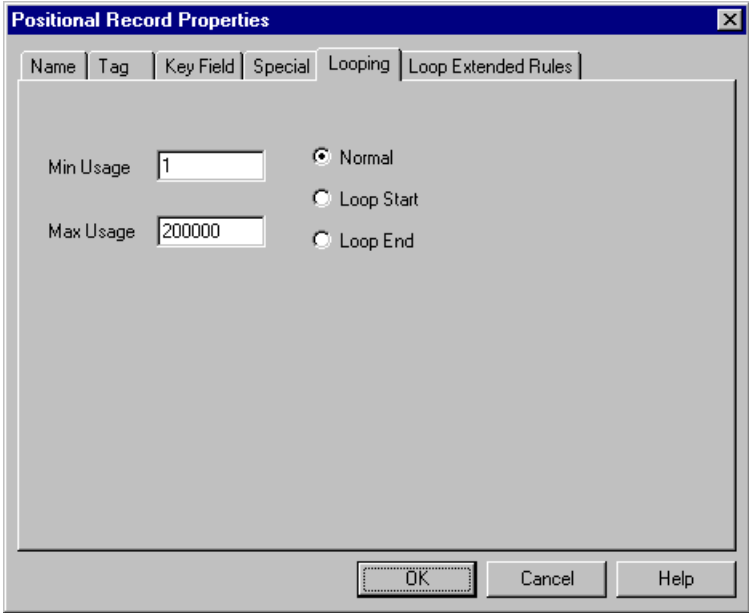
Overview The INVDETAIL (detail) record is at the same level as the HEADER and INVBEGIN records and the INVCommentGroup.

The INVDETAIL (detail) record contains line item number and description, quantity shipped, unit of measure, price, product code, and VAT calculation information. 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 dialogue box.
2	In the Name box, type “INVDETAIL” as the record identification code. Caution Do not use spaces or hyphens(-) in the Name box, You can use the underscore (_) to separate words.
3	In the Desc box, type “Detail Record” as a description of the record. This box is used to provide a brief description 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 recognised by the system as the following: <TAG>[Field_1] [Field_2] The record TAG for each record enables the system to recognise that record and then determine the mapping requirements. Note You must define the record tag on the Positional Record Properties dialogue 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” 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	<p>In the Min Usage box, type “1” to make the record mandatory. This box contains the minimum number of times the record must repeat.</p> <p>Note If the Min Usage box contains “0” (zero), the record is conditional. If the Min Usage box contains “1” or greater, the record is mandatory.</p>
9	<p>In the Max Usage box, type “200000” as the maximum number of times the record can repeat. The Positional Record Properties dialogue box should now look like this:</p> 
10	Click OK to create the INVDETAIL record.

Creating the TEMPVATSTL Record

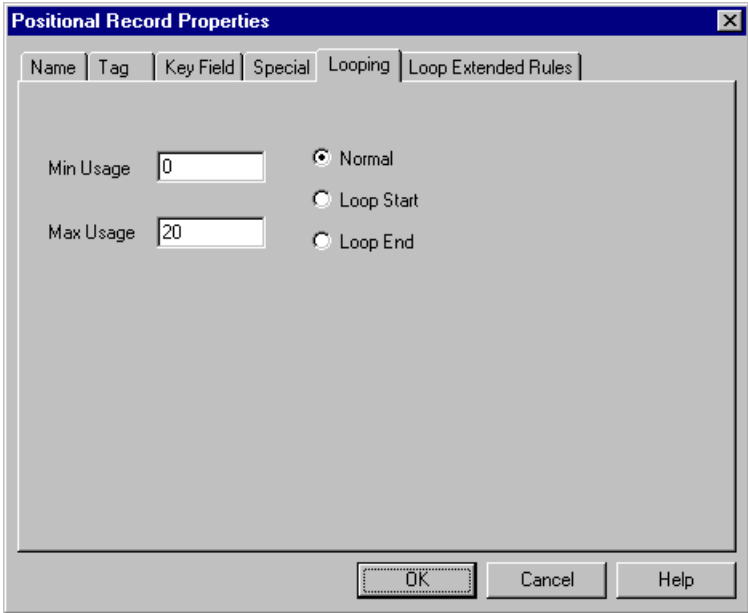
Overview The TEMPVATSTL (Temporary Storage VAT Subtotals) record is at the same level as the HEADER, INVBEGIN, and INVDETAIL records and the INVCommentGroup.

The TEMPVATSTL (temporary storage for VAT subtotals) record contains temporary storage fields to enable VAT subtotal calculations. The temporary storage record is conditional and occurs up to 20 times per invoice translated (once for each possible VAT code subtotal).

Procedure Use this procedure to create the TEMPVATSTL 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 dialogue box.
2	In the Name box, type “TEMPVATSTL” as the record identification code. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.
3	In the Desc box, type “Temporary Storage VAT Subtotals” 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 “TMP” as the record identification code <TAG>. <TAG>[Field_1][Field_2] The record TAG for each record enables the system to recognise that record and then determine the mapping requirements. Note You must define the record tag on the Positional Record Properties dialogue 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” 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	<p>In the Min Usage box, accept the default of “0” so the record is conditional. This box contains the minimum number of times the record must repeat.</p> <p>Note If the Min Usage box contains “0” (zero), the record is conditional. If the Min Usage box contains “1” or greater, the record is mandatory.</p>
9	<p>In the Max Usage box, type “20” as the maximum number of times the record can repeat. The Positional Record Properties dialogue box should now look like this:</p> 
10	Click OK to create the TEMPVATSTL record.

Creating the INVSUMMARY Record

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

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

Procedure Use this procedure to create the INVSUMMARY record.

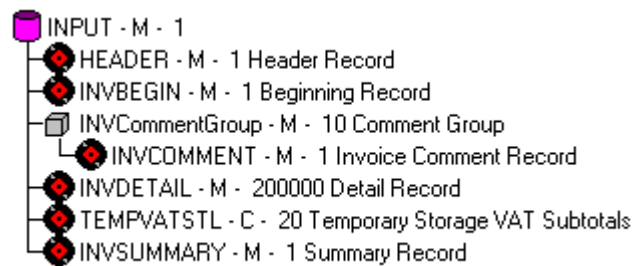
Step	Action
1	Highlight the TEMPVATSTL record. From the Edit menu, select Insert. From the submenu, select Record to display the Positional Record Properties dialogue box.
2	In the Name box, type “INVSUMMARY” as the record identification code. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.
3	In the Desc box, type “Summary Record” as a description of the record. This box is used to provide a brief explanation of the record that allows you to differentiate it from similar records.
4	Select the Tag tab to access tag information.
5	In the Tag box, type “SUM” as the record identification code <TAG>. <TAG>[Field_1] [Field_2] The record TAG for each record enables the system to recognise that record and then determine the mapping requirements. Note You must define the record tag on the Positional Record Properties dialogue 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” as the starting column position of the tag in the data record.
7	Select the Looping tab.

(Continued on next page)

Step	Action
8	In the Min Usage box, type “1” to make the record mandatory. This box contains the minimum number of times the record must repeat. Note If the Min Usage box contains “0” (zero), the record is conditional. If the Min Usage box contains “1” or greater, the record is mandatory.
9	Accept the default of “1” in the Max Usage box. This is the maximum number of times the record can repeat.
10	Click OK to create the INVSUMMARY record.

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:



Creating the Application Fields

Overview

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

The easiest way to add application fields to a record is to use the Positional Field Editor. You will 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
APPL_ID		Application ID	String	0	6
APPL_ALIAS_VALUE		Application Alias Value	String	0	6
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
SHIPTONAME		Ship To Name	String	0	35
SHIPTOADD1		Ship To Address Line 1	String	0	35
SHIPTOADD2		Ship To Address Line2	String	0	35
<i>(Continued on next page)</i>					

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
SHIPTOCITY		Ship To City	String	0	35
SHIPTOSTATE		Ship To State	String	0	35
SHIPTOPOSTAL CODE		Ship To Postal Code	String	0	8
SHIPTOLOC		Ship To Store Location	String	0	8
BOLNUM		Bill of Lading Number	String	0	12
SHIPDATE		Ship Date	Dt/Tm (YYMMDD)	6	6
TERMSNETDUE		Terms New Due Days	Number (R0)	0	3
TERMSTEXT		Payment Terms Text	String	0	40
Comment Group (INVCommentGroup) 10 per message					
Invoice Comment Record (INVCOMMENT) 1 per message					
BILLTOACCTCMT	X	Bill To Acct Number	String	1	9
DOCTYPCMT	X	Document Type	String	1	6
INVNUMCMT	X	Invoice Number	String	1	10
MESSAGETEXT1		Message Text Line 1	String	0	40
MESSAGETEXT2		Message Text Line 2	String	0	40
MESSAGETEXT3		Message Text Line 3	String	0	40
MESSAGETEXT4		Message Text Line 4	String	0	40
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)	Number (R0)	1	6
QTYSHIP	X	Quantity Shipped	Number (R0)	1	5
UNITPRICE	X	Unit Price	Number (R2)	1	7
UOM	X	Unit of Measure	String	1	4
<i>(Continued on next page)</i>					

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
CUSTPROCEDURE	X	Customer Product Code	String	1	10
UPCCODE	X	UPC Code	String	1	12
ITEMDESC		Item Description	String	0	30
VATCODE	X	VAT Rate Code	String	1	2
TEMPVATRATE		Temporary Storage VAT Percentage Rate	String	0	10
TEMPLEXC		Temporary Line Extended Cost	Number (R4)	0	14
Temporary Storage VAT Record (TEMPVATSTL) up to 10 per message					
BILLTOACCTTS	X	Bill To Acct Number	String	1	9
DOCTYPTS	X	Document Type	String	1	6
INVNUMTS	X	Invoice Number	String	1	10
SEQA		Sequence Number of STL	Number (R0)	0	10
VATRATE		VAT Rate	Number (R3)	0	6
VATRATECODE		VAT Code	String	0	1
NUMLINES		Number of Lines for VAT Code	Number (R0)	0	10
LVLA		Line Subtotal Amount (Before VAT)	Number (R2)	0	12
VATA		VAT Amount Payable	Number (R2)	0	12
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 (R2)	1	10

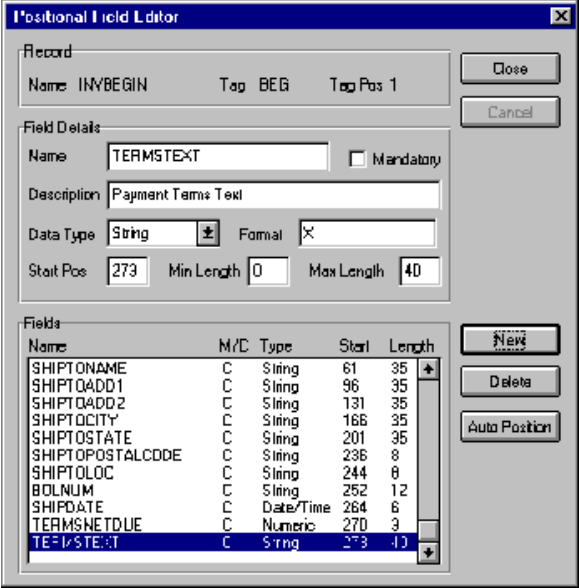
How to Define the INVBEGIN Record Application Fields

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

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

Step	Action
1	Click the INVBEGIN application record with the right mouse button to access the shortcut menu.
2	From the shortcut menu, select Edit Fields to access the Positional Field Editor dialogue box.
3	If the field you are create 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 values in the Field Details section.
4	In the Name box, type “BILLTOACCTBEG”, which is the name of the first field in the INVBEGIN record. Each application field must have a unique name. It is useful to tag the end of the fields that occur in multiple records with a suffix that identifies the record that contains it. For example, use “BEG” for the Beginning record. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.
5	Click the Mandatory check box to identify it as a required field.
6	In the Description box, type “Bill To Account Number” as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select “String” as the type of the field which means this field must contain alphanumeric data. Note If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You want to specify field start positions if, for example, you are only using a few fields but you want them positioned exactly in the record. In this tutorial, you 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. <p style="text-align: right;">(Continued on next page)</p>

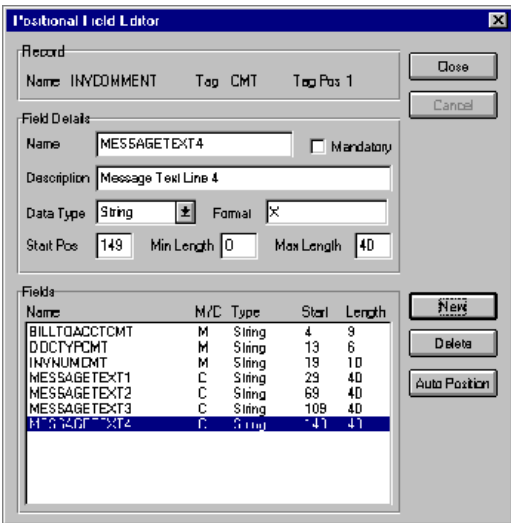
(Contd) Step	Action
8	If the Data Type is “Number” or “Dt/Tm,” select the appropriate format from the Format list.
9	In the Min Length box, type “1”. This is the minimum number of characters for this field. If, according to the table starting on 3 - 63, the minimum length of the field is “0”, then accept that default in the Min Length box.
10	In the Max Length box, type “9” as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press Enter. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to the record layout table starting on 3 - 63. <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
13	<p>After you add the last field, click Auto Position to automatically position the fields in the record. The Auto Position function 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 will be sequenced in order.</p> <p>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 dialogue box, and if you define every field in the record in the sequence that each field occurs, as you are doing in this tutorial.</p> <p>The Positional Field Editor dialogue box should now look like the following (only the last eleven fields in the record are currently visible in the Fields list):</p> 
14	<p>When you are completely finished adding fields to the INVBEGIN record, click Close.</p>

How to Define the INVCOMMENT Record Application Fields

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


Step	Action
1	Click the INVCOMMENT application record with the right mouse button to access the shortcut menu.
2	From the shortcut menu, select Edit Fields to access the Positional Field Editor dialogue 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	<p>In the Name box, type “BILLTOACCTCMT”, which is the name of the first field in the INVCOMMENT 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, use “CMT” for the first comment record.</p> <p>Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.</p>
5	Click the Mandatory check box to identify it is a required field.
6	In the Description box, type “Bill To Account Number” as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	<p>From the Data Type list, select “String” as the type of the field which means this data must contain alphanumeric data.</p> <p>Note If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.</p> <p>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 on the Auto Position function.</p>
8	<p>If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format list.</p> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
9	In the Min Length box, type “1”. This is the minimum number of characters for this field. If, according to the table starting on 3 - 63, the minimum length of the field is “0”, then accept that default in the Min Length box.
10	In the Max Length box, type “9” as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press Enter. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to the record layout table starting on 3 - 63.
13	<p>After adding the last field, click Auto Position to automatically position the fields in the record. The Auto Position function 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 will be sequenced in order.</p> <p>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 dialogue box, and if you define every field in the record in the sequence that each field occurs, as you are doing in this tutorial.</p> <p>The Positional Field Editor dialogue box should now look like this:</p> 
14	When you are completely finished adding fields to the INVCOMMENT record, click Close .

How to Define the INVDETAIL Record Application Fields

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

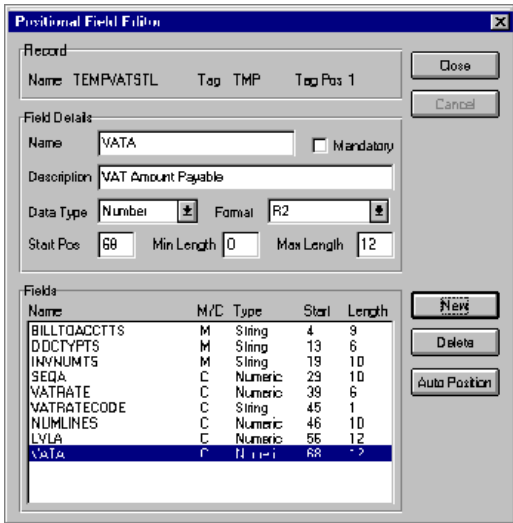
Step	Action
1	Click the INVDETAIL application record with the right mouse button to access the shortcut menu.
2	From the shortcut menu, select Edit Fields to access the Positional Field Editor dialogue box.
3	If the field you are creating is the first field in the record, click New . A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4	In the Name box, type “BILLTOACCTDET”, which is the name of the first field in the INVDETAIL record. Each application field must have a unique name. It is useful to tag the end of the fields that occur in multiple records with a suffix that identifies the record that contains it. For example, use “DET” for the detail record. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.
5	Click the Mandatory check box to identify it is a required field.
6	In the Description box, type “Bill To Account Number” as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select “String” as the type of the field which means this data must contain alphanumeric data. Note If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record. 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 on the Auto Position function.
8	If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format list. <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
9	In the Min Length box, type “1”. This is the minimum number of characters for this field. If, according to the table starting on 3 - 63, the minimum length of the field is “0”, then accept that default in the Min Length box.
10	In the Max Length box, type “9” as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press Enter. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to the record layout table starting on 3 - 63.
13	<p>After adding the last field, click Auto Position to automatically position the fields in the record. The Auto Position function 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 will be sequenced in order.</p> <p>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 dialogue box, and if you define every field in the record in the sequence that each field occurs, as you are doing in this tutorial.</p> <p>The Positional Field Editor dialogue box should now look like this:</p> 
14	When you are completely finished adding fields to the INVDETAIL record, click Close .

How to Define the TEMPVATSTL Record Application Fields

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

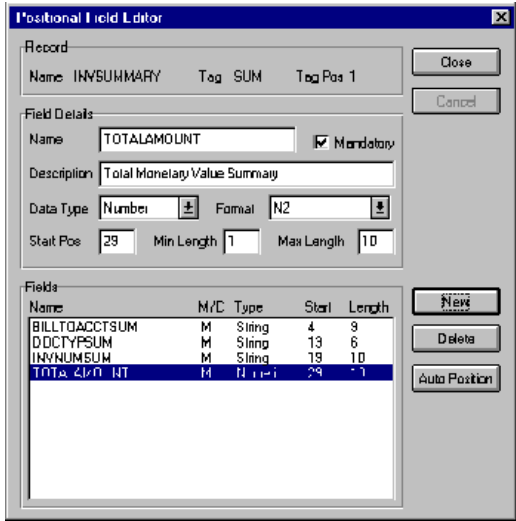
Step	Action
1	Click the TEMPVATSTL application record with the right mouse button to access the shortcut menu.
2	From the shortcut menu, select Edit Fields to access the Positional Field Editor dialogue box.
3	If the field you are creating is the first field in the record, click New . A highlight bar 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 “BILLTOACCTTS”, which is the name of the first field in the TEMPVATSTL 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, use “TS” for the temporary storage record. Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.
5	Click the Mandatory check box to identify it is a required field.
6	In the Description box, type “Bill To Account Number” as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	From the Data Type list, select “String” as the type of the field which means this data must contain alphanumeric data. Note If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record. 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 on the Auto Position function.
8	If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format list. <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
9	In the Min Length box, type “1”. This is the minimum number of characters for this field. If, according to the table starting on 3 - 63, the minimum length of the field is “0”, then accept that default in the Min Length box.
10	In the Max Length box, type “9” as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press Enter. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to the record layout table starting on 3 - 63.
13	<p>After adding the last field, click Auto Position to automatically position the fields in the record. The Auto Position function 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 will be sequenced in order.</p> <p>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 dialogue box, and if you define every field in the record in the sequence that each field occurs, as you are doing in this tutorial.</p> <p>The Positional Field Editor dialogue box should now look like this:</p> 
14	When you are completely finished adding fields to the TEMPVATSTL record, click Close .

How to Define the INVSUMMARY Record Application Fields

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

Step	Action
1	Click the INVSUMMARY application record with the right mouse button to access the shortcut menu.
2	From the shortcut menu, select Edit Fields to access the Positional Field Editor dialogue 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	<p>In the Name box, type “BILLTOACCTSUM”, which is the name of the first field in the INVSUMMARY record. Each application field must have a unique name. It is useful to tag the end of the fields that occur in multiple records with a suffix that identifies the record that contains it. For example, use “SUM” for the temporary storage record.</p> <p>Caution Do not use spaces or hyphens (-) in the Name box. You can use the underscore (_) to separate words.</p>
5	Click the Mandatory check box to identify it is a required field.
6	In the Description box, type “Bill To Account Number” as the description of the field. The description is used to provide a brief explanation of the field that allows you to differentiate it from similar fields.
7	<p>From the Data Type list, select “String” as the type of the field which means this data must contain alphanumeric data.</p> <p>Note If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos box. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.</p> <p>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 on the Auto Position function.</p>
8	<p>If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format list.</p> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
9	In the Min Length box, type “1”. This is the minimum number of characters for this field. If, according to the table starting on 3 - 63, the minimum length of the field is “0”, then accept that default in the Min Length box.
10	In the Max Length box, type “9” as the maximum length of the field.
11	To add the field and to create the next field positioned after it, press Enter. A new field with blank values ready for you to identify is positioned below the field.
12	Create the rest of the fields according to the record layout table starting on 3 - 63.
13	<p>After adding the last field, click Auto Position to automatically position the fields in the record. The Auto Position function 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 will be sequenced in order.</p> <p>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 dialogue box, and if you define every field in the record in the sequence that each field occurs, as you are doing in this tutorial.</p> <p>The Positional Field Editor dialogue box should now look like this:</p> 
14	When you are completely finished adding fields to the INVSUMMARY record, click Close .

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

Mapping Your Data

Overview

Introduction

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

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

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

HEADER record

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

INVBEGIN record

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

BILLTOACCTBE G field

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

Mapping the INVNUMBEG Field

Overview

The IRF segment contains invoice references. You can map the INVNUMBEG field directly to the IRF INVN element (Invoice Number), by using the Link function. The Link function allows you to map a field or element from the Input side of the map to a field or element on the Output side of the map. The link between the two fields is visually represented with a line connecting the two fields.

Also, we recommend that you set up a Document Name for each map, to make the identification of a message created by this map easier in Gentran:Server for Windows. The Document Name 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 will distinguish the messages translated by this map. Then, the Name field of the document browsers that contain this message will contain the data from the selected element or field.

Recommendation

You can choose any element or field as the Document Name, but we strongly recommend that you select an element or field that occurs only once in the message and has a data type of "String". For an Invoice, the document name is typically the Invoice Number.

How to map the invoice number

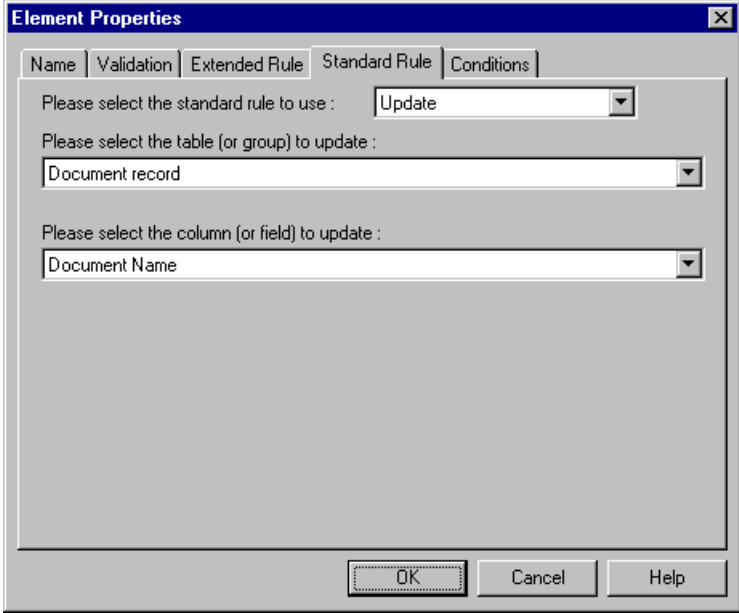
Use this procedure to map the invoice number.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the application field INVNUMBEG in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the IRF INVN element (Invoice Number). A line is displayed, visually illustrating the link between the INVNUMBEG field and the IRF INVN element.

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 IRF INVN element (Invoice Number) to display the Element Properties dialogue box.
2	Select the Standard Rule tab to access the standard rule options. (Continued on next page)

(Contd) Step	Action
3	From the standard rule list, select “Update.”
4	From the table to update list, choose “Document record.” This indicates that you are updating the document record in Gentran:Server for Windows.
5	<p>From the column list, choose “Document Name.” This indicates that you are updating the Name field in the document browsers with the contents of this element.</p> <p>The Standard Rule tab of the dialogue box should now look like this:</p> 
6	Click OK to set up the document name.

How to Map the INVDATE Field

Overview

You can map the INVDATE field directly to both the IRF IVDT (Date of Invoice) and IRF TXDT (Tax-Point Date) elements by using the Link function, because both of these dates are the same.

How to map the invoice and tax-point dates

Use this procedure to map the invoice and tax-point dates.

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 application field INVDATE in the INVBEGIN record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the IRF IVDT element (Date of Invoice). A line is displayed, visually illustrating the link between the INVDATE field and the IRF IVDT element.
4	Click the application field INVDATE in the INVBEGIN record again. When you click the field the cursor changes to a “Link Arrow.”
5	Click the IRF TXDT element (Tax-Point Date). A line is displayed, visually illustrating the link between the INVDATE field and the IRF TXDT element.

Mapping the PONUM Field

Overview

The ODD group contains order and delivery terms and invoice line details. This group repeats up to 999,999 times. You only need to send one ODD group to your partner because you are invoicing only one order for this message. Therefore, you will change the maximum usage of the ODD group to make it a nonrepeating group.

Since you know that you will only be invoicing one order per INVOIC message, you can use extended rules to hard code the first level sequence numbers to one. The first level sequence numbers (PYT SEQA, DNA SEQA:2, ODD:2 SEQA, and ILD:2 SEQA) each indicate the number of orders that this message is invoicing.

Then, you can map the PONUM field directly to the ODD:2 ORNO ORN0 element (Customer's Order Number), by using the Link function.

How to change the maximum usage of the ODD group

Use this procedure to make the ODD a nonrepeating group.

Step	Action
1	Click the ODD group with the right mouse button to access the shortcut menu.
2	From the shortcut menu, select Properties to access the Group Properties dialogue box.
3	In the Maximum Usage box, type "1" to make the ODD group a nonrepeating group. This box currently contains "999999".
4	Click the Promote records to parent check box to specify that when the group is compiled, the subordinate segments and group are extracted from the loop and located in the parent group.
5	Click OK to accept the change to the ODD group.

How to create an extended rule to specify the PYT SEQA

Use this procedure to create an extended rule to hard code the first level sequence numbers for the PYT segment (Settlement Terms) to one, indicating the number of orders that this message is invoicing.

Step	Action
1	Double-click the PYT SEQA element (First Level Sequence Number) to access the Element Properties dialogue box.
2	Select the Extended Rule tab.

(Continued on next page)

(Contd) Step	Action
3	<p>In the Rule list, type the following:</p> <pre>#SEQA = 1;</pre> <p>The “#SEQA = 1” sets the first level sequence number for the PYT segment to one. The system executes this rule when it processes the PYT SEQA element.</p>
4	<p>Click Compile to validate the syntax of the extended rule. Every rule in the map is compiled when you compile the translation object after you complete the map. However, the system allows you to compile each rule individually so that you can verify the accuracy of the rule after you create it.</p> <p>Note The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list.</p>
5	Click OK to add the extended rule to the PYT SEQA element.

How to create an extended rule to specify the DNA SEQA:2

Use this procedure to create an extended rule to hard code the first level sequence number for the DNA segment (Data Narrative) to one, indicating the number of orders that this message is invoicing.

Step	Action
1	Double-click the DNA SEQA:2 element (First Level Sequence Number) to access the Element Properties dialogue box.
2	Select the Extended Rule tab.
3	<p>In the Rule list, type the following:</p> <pre>#SEQA:2 = 1;</pre> <p>The “#SEQA:2 = 1” sets the first level sequence number for the DNA segment to one. The system executes this rule when it processes the DNA SEQA:2 element.</p> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
4	<p>Click Compile to validate the syntax of the extended rule. Every rule in the map is compiled when you compile the translation object after you complete the map. However, the system allows you to compile each rule individually so that you can verify the accuracy of the rule after you create it.</p> <p>Note The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list.</p>
5	Click OK to add the extended rule to the DNA SEQA:2 element.

How to create an extended rule to specify the ODD:2 SEQA

Use this procedure to create an extended rule to hard code the first level sequence numbers for the ODD:2 segment (Order and Delivery Terms) to one, indicating the number of orders that this message is invoicing.

Step	Action
1	Double-click the ODD:2 SEQA element (First Level Sequence Number) to access the Element Properties dialogue box.
2	Select the Extended Rule tab.
3	<p>In the Rule list, type the following:</p> <pre>#SEQA = 1;</pre> <p>The “#SEQA = 1” sets the first level sequence number for the ODD:2 segment to one. The system executes this rule when it processes the ODD:2 SEQA element.</p>
4	<p>Click Compile to validate the syntax of the extended rule. Every rule in the map is compiled when you compile the translation object after you complete the map. However, the system allows you to compile each rule individually so that you can verify the accuracy of the rule after you create it.</p> <p>Note The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list.</p>
5	Click OK to add the extended rule to the ODD:2 SEQA element.

How to create an extended rule to specify the ILD:2 SEQA

Use this procedure to create an extended rule to hard code the first level sequence numbers for the ILD:2 segment (Invoice Line Details) to one, including the number of orders that this message is invoicing.

Step	Action
1	Double-click the ILD:2 SEQA element (First Level Sequence Number) to access the Element Properties dialogue box.
2	Select the Extended Rule tab.
3	In the Rule list, type the following: #SEQA = 1; The “#SEQA = 1” sets the first level sequence number for the ILD:2 segment to one. The system executes this rule when it processes the ILD:2 SEQA element.
4	Click Compile to validate the syntax of the extended rule. Every rule in the map is compiled when you compile the translation object after you complete the map. However, the system allows you to compile each rule individually so that you can verify the accuracy of the rule after you create it. Note The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list.
5	Click OK to add the extended rule to the ILD:2 SEQA element.

How to map the order number

Use this procedure to map the customer’s order number.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the application field PONUM in the INVBEGIN record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the ODD:2 ORNO ORN0 element (Customer’s Order Number). A line is displayed, visually illustrating the link between the PONUM field and the ODD:2 ORNO ORN0 element.

How to Map the PODATE Field

Overview You can map the PODATE field directly to the ODD:2 ORNO ORN2 element (Date Order Placed by Customer) by using the Link function.

Procedure Use this procedure to map the date order placed by customer (Purchase Order Date).

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the application field PODATE in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the ODD:2 ORNO ORN2 element (Date Order Placed by Customer). A line is displayed, visually illustrating the link between the PODATE field and the ODD:2 ORNO ORN2 element.

Mapping the Shipping Information

How to Map the ShipTo Fields

Overview The shipping name, address, and location information can be directly mapped to the CLO segment (Customer's Location), by using the Link function.

How to map the SHIPTONAME field

Use this procedure to map the customer's name.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the application field SHIPTONAME in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the CLO CNAM element (Customer's Name). A line is displayed, visually illustrating the link between the SHIPTONAME field and the CLO CNAM element.

How to map the SHIPTOADD1 field

Use this procedure to map the customer's address line 1.

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 application field SHIPTOADD1 in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the CLO CADD CAD0 element (Customer's Address Line 1). A line is displayed, visually illustrating the link between the SHIPTOADD1 field and the CLO CADD CAD0 element.

How to map the SHIPTOADD2 field

Use this procedure to map the customer's address line 2.

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 application field SHIPTOADD2 in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the CLO CADD CAD1 element (Customer's Address Line 2). A line is displayed, visually illustrating the link between the SHIPTOADD2 field and the CLO CADD CAD1 element.

How to map the SHIPTOCITY field

Use this procedure to map the customer's address line 3.

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 application field SHIPTOCITY in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the CLO CADD CAD2 element (Customer's Address Line 3). A line is displayed, visually illustrating the link between the SHIPTOCITY field and the CLO CADD CAD2 element.

How to map the SHIPTOSTATE field

Use this procedure to map the customer's address line 4.

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 application field SHIPTOSTATE in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the CLO CADD CAD3 element (Customer's Address Line 4). A line is displayed, visually illustrating the link between the SHIPTOSTATE field and the CLO CADD CAD3 element.

How to map the SHIPTOPOSTALCODE field

Use this procedure to map the customer's post code.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the application field SHIPTOPOSTALCODE in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the CLO CADD CAD4 element (Customer's Postal Code). A line is displayed, visually illustrating the link between the SHIPTOPOSTALCODE field and the CLO CADD CAD4 element.

How to map the SHIPTOLOC field

The SHIPTOLOC (ship to location) contains additional reference information about the Ship To address. This field can be directly mapped to the CLO segment (Customer's Location), by using the Link function.

Use this procedure to map the customer's own location code.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the application field SHIPTOLOC in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the CLO CLOC CLO1 element (Customer's Own Location Code). A line is displayed, visually illustrating the link between the SHIPTOLOC field and the CLO CLOC CLO1 element.

How to map the BOLNUM field

You can map the BOLNUM field directly to the ODD:2 DELN DEL0 element (Delivery Note Number), by using the Link function.

Use this procedure to map the delivery note (Bill of Lading) number.

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the application field BOLNUM in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."

(Continued on next page)

Step	Action
3	Click the ODD:2 DELN DEL0 element (Delivery Note Number). A line is displayed, visually illustrating the link between the BOLNUM field and the ODD:2 DELN DEL0 element.

How to map the SHIPDATE field

You can map the SHIPDATE field directly to the ODD:2 DELN DEL1 element (Date of Despatch by Supplier), by using the Link function.

Use this procedure to map the date of despatch by supplier (Ship Date).

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the application field SHIPDATE in the INVBEGIN record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the ODD:2 DELN DEL1 element (Date of Despatch by Supplier). A line is displayed, visually illustrating the link between the SHIPDATE field and the ODD:2 DELN DEL1 element.

How to map the TERMSNETDUE field

The PYT segment contains settlement terms and it repeats up to 999,999 times. You only need to send one PYT segment to your partner because you are invoicing only one order for this message. Therefore, you will change the maximum usage of the PYT segment from to make it a nonrepeating segment.

You can map the TERMSNETDUE field directly to the PYT PAYY PAY2 element (Number of Days), by using the Link function.

Use this procedure to make the PYT a nonrepeating segment.

Step	Action
1	Click the PYT segment with the right mouse button to access the shortcut menu.
2	From the shortcut menu, select Properties to access the EDI Segment Properties dialogue box.
3	In the Max Usage box, type “1” to make the PYT a nonrepeating segment. This box currently contains “999999”.
4	Click OK to accept the change to the PYT segment.

**How to map the
TERMSNETDUE
field (contd)**

Use this procedure to map the number of days (terms net due days).

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 application field TERMSNETDUE in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the PYT PAYY PAY2 element (Number of Days). A line is displayed, visually illustrating the link between the TERMSNETDUE field and the PYT PAYY PAY2 element.

**How to map the
TERMSTEXT
field**

You can map the TERMSTEXT field directly to the PYT PAYT element (Terms of Payment) by using the Link function.

Use this procedure to map the terms of payment (Terms Text).

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 application field TERMSTEXT in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the PYT PAYT element (Terms of Payment). A line is displayed, visually illustrating the link between the TERMSTEXT field and the PYT PAYT element.

The INVCommentGroup: INVCOMMENT Record

Overview

Introduction

The INVCommentGroup (comment information) groups all the comment information for the invoice, including the message text. The INVCommentGroup is mandatory and occurs up to ten times per invoice translated.

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

BILLTOACCTC MT field

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

DOCTYPCMT field

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

INVNUMCMT field

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

Mapping the MESSAGETEXT Information

Overview The message text (comment) information is contained in the DNA segment (Data Narrative). The MESSAGETEXT1, MESSAGETEXT2, MESSAGETEXT3, and MESSAGETEXT4 fields can all be directly mapped to elements in the DNA segment.

How to map the MESSAGETEXT1 field

Use this procedure to map the general narrative line 1 (Message Text 1).

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 application field MESSAGETEXT1 in the INVCOMMENT record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the DNA GNAR GNA0 element (General Narrative Line 1). A line is displayed, visually illustrating the link between the MESSAGETEXT1 field and the DNA GNAR GNA0 element.

How to map the MESSAGETEXT2 field

Use this procedure to map the general narrative line 2 (Message Text 2).

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 application field MESSAGETEXT2 in the INVCOMMENT record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the DNA GNAR GNA1 element (General Narrative Line 2). A line is displayed, visually illustrating the link between the MESSAGETEXT2 field and the DNA GNAR GNA1 element.

How to map the MESSAGETEXT3 field

Use this procedure to map the general narrative line 3 (Message Text 3).

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 application field MESSAGETEXT3 in the INVCOMMENT record. When you click the field, the cursor changes to a "Link Arrow." (Continued on next page)

Step	Action
3	Click the DNA GNAR GNA2 element (General Narrative Line 3). A line is displayed, visually illustrating the link between the MESSAGETEXT3 field and the DNA GNAR GNA2 element.

**How to map the
MESSAGETEXT4
field**

Use this procedure to map the general narrative line 4 (Message Text 4).

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 application field MESSAGETEXT4 in the INVCOMMENT record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the DNA GNAR GNA3 element (General Narrative Line 3). A line is displayed, visually illustrating the link between the MESSAGETEXT4 field and the DNA GNAR GNA3 element.

The INVCommentGroup: INVDETAIL Record

Overview

Introduction

This section describes the actions necessary for you to map from each field in the INVDETAIL (detail) record. The INVDETAIL (detail) record contains line item number and description, quantity shipped, unit of measure, price, product code, and VAT calculation information. The detail record is mandatory and occurs up to 200,000 times per invoice translated (once for each line item invoiced).

BILLTOACCTDET 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. Therefore, 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. Therefore, 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. Therefore, you will not perform any mapping operations on this field.

How to Map the ITEMNUM Field

Overview The ITEMNUM field can be directly mapped to the ILD:2 SEQB element (2nd Level Sequence No.), by using the Link function, because the number of the line item is the second level sequence number. The second level of the invoice is the line item details.

Procedure Use this procedure to map the second level sequence number (item number).

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the application field ITEMNUM in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the ILD:2 SEQB element (2nd Level Sequence No.). A line is displayed, visually illustrating the link between the ITEMNUM field and the ILD:2 SEQB element.

Mapping the QTYSHIP Field

Overview

On the EDI side of the map, the quantity shipped information is contained in the ILD:2 QTYI QTY0 element (Number of Order Units Invoiced). The ILD:2 QTYI QTY0 element is defined as a string-type element in the standard. However, you have determined that you will send your partner a number-type element in N0 format (formatted with an implicit decimal point and no decimal places).

Therefore, you will need to change the properties of the ILD:2 QTYI QTY0 element to reflect the type and the format that you have determined that your partner will send you. Then, you can map the quantity shipped directly from the QTYSHIP field to the ILD:2 QTYI QTY0 element.

How to change the properties of the ILD:2 QTYI QTY0 element

Use this procedure to change the type and format of the ILD:2 QTYI QTY0 element.

Step	Action
1	Double-click the ILD:2 QTYI QTY0 element (Number of Order Units Invoiced) to display the Element Properties dialogue box.
2	Select the Validation tab.
3	From the data-type list, select “Number” to indicate that this is a number element. The system prompts you with a message that changing the field’s data type invalidates its format. Click Yes to acknowledge this message and proceed.
4	From the Format list, select “N0” to indicate that this element will be formatted with an implicit decimal point and no decimal place.

How to map the number of order units invoiced

After changing the type and format of the ILD:2 QTYI QTY0 element, you can map the QTYSHIP field directly to it.

Use this procedure to map the quantity shipped.

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 application field QTYSHIP in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the ILD:2 QTYI QTY0 element (Number of Order Units Invoiced). A line is displayed, visually illustrating the link between the QTYSHIP field and the ILD:2 QTYI QTY0 element.

How to Map the UNITPRICE Field

Procedure The UNITPRICE field can be directly mapped to the ILD:2 AUCT AUC0 element (Cost Price (Excluding VAT)).

Use this procedure to map the cost price (excluding VAT).

Step	
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 application field UNITPRICE in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the ILD:2 AUCT AUC0 element (Cost Price (Excluding VAT)). A line is displayed, visually illustrating the link between the UNITPRICE field and the ILD:2 AUCT AUC0 element.

How to Map the UOM Field

Procedure You mapped the quantity shipped to the QTYI composite and the unit price to the AUCT composite. Therefore, you want to map the unit of measure to both composites.

Use this procedure to map the unit of measure.

Step	
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 application field UOM in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the ILD:2 QTY1 CD04:2 element (Measure Indicator). A line is displayed, visually illustrating the link between the UOM field and the ILD:2 QTY1 CD04:2 element.
4	Click the application field UOM in the INVDETAIL record again. When you click the field, the cursor changes to a “Link Arrow.”
5	Click the ILD:2 AUCT CD04:3 element (Measure Indicator). A line is displayed, visually illustrating the link between the UOM field and the ILD:2 AUCT CD04:3 element.

How to Map the CUSTPROCEDURE Field

Procedure You can map the CUSTPROCEDURE (customer product code) field directly to the ILD:2 CPRO CPR1 element (Customer's Item Code), by using the Link function.

Use this procedure to map the customer's item code (customer product code).

Step	
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 application field CUSTPROCEDURE in the INVDETAIL record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the ILD:2 CPRO CPR1 element (Customer's Item Code). A line is displayed, visually illustrating the link between the CUSTPROCEDURE field and the ILD:2 CPRO CPR1 element.

How to Map the UPCCODE Field

Procedure You can map the UPCCODE field directly to the ILD:2 SPRO CPR1 element (Supplier's Code for the Traded Unit), by using the Link function.

Use this procedure to map the supplier's code for the traded unit (UPC code).

Step	
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 application field UPCCODE in the INVDETAIL record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the ILD:2 SPRO CPR1 element (Supplier's Code for the Traded Unit). A line is displayed, visually illustrating the link between the UPCCODE field and the ILD:2 SPRO CPR1 element.

How to Map the ITEMDESC Field

Procedure You can map the ITEMDESC (item description) field directly to the ILD:2 TDES TDE0 element (Traded Unit Descr. Line 1), by using the Link function.

Use this procedure to map the traded unit description (item description).

Step	
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 application field ITEMDESC in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the ILD:2 TDES TDE0 element (Traded Unit Descr. Line 1). A line is displayed, visually illustrating the link between the ITEMDESC field and the ILD:2 TDES TDE0 element.

Mapping the VATCODE Field

Overview

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

For this map, you need to map the appropriate VAT rate based on the code in the VATCODE field from a lookup table, to look up data that you will map (via standard rule) to the TEMPVATRATE field. If you need to use tables, you must decide whether to create a code table in the Application Integration subsystem that can only be used with this map, or to add the table to either the partner relationship or the internal system partner in Partner Editor. You will set up the table you use in this tutorial under the internal system partner (Internal System User) so it can be used globally.

In this tutorial, you need to extract information from a lookup table to be inserted into the outbound data. A lookup table is used when you want to supplement data before it is sent to your partner. Please see the Gentran:Server for Windows *User's Guide* for more information on lookup tables.

Note

You do not need to create this lookup table (INVATCO.LKP). It is included with the tutorial data that is provided when you install Gentran:Server for Windows.

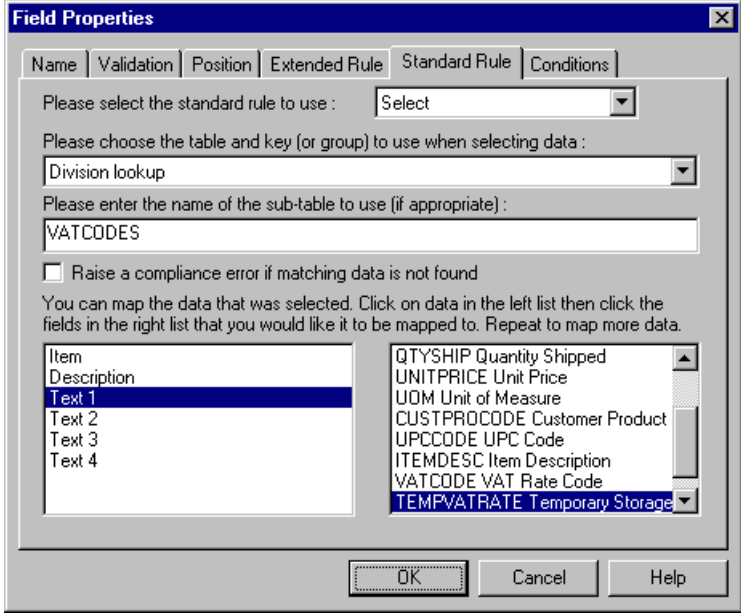
How to map the VAT rate

After you look up the appropriate VAT rate, you can map the VATCODE (VAT code) field directly to the ILD:2 CD12 element (VAT Rate Category Code), by using the Link function.

Use this procedure to look up the appropriate VAT rate for the specified VAT code.

Step	Action
1	Double-click the VATCODE field (in the INVDETAIL record). The Field Properties dialogue box is displayed.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, choose "Select."
4	From the Table list, select "Division lookup." This indicates that the key is the partner lookup table name for the Internal System User partner and enables you to map from any field on the Lookup Entry dialogue box of the Partner Editor.
5	In the sub-table box, type "VATCODES". This is the name of the lookup table you imported.

(Continued on next page)

(Contd) Step	Action
6	From the Map From list on the left, select “Text 1.” This is the box on the Lookup Entry dialogue box of the Partner Editor from which the contents will be mapped.
7	From the Map To list on the right, select “TEMPVATRATE.” This is the box to which the information from the Partner Editor will be mapped. The Standard Rule tab of the Field Properties dialogue box should now look like this: 
8	Click OK to map from the lookup table to the TEMPVATRATE field.

How to map the VAT rate category code

Use this procedure to map the VAT rate category code (VAT code).

Step	Action
1	If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2	Click the application field VATCODE in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the ILD:2 CD12 element (VAT Rate Category Code). A line is displayed, visually illustrating the link between the VATCODE field and the ILD:2 CD12 element.

How to Map the TEMPVATRATE Field

Overview You already used a division lookup table and a standard rule on the VATCODE field to map the appropriate VAT rate (based on the code in the VATCODE field) from the VATCODES division lookup table to the TEMPVATRATE field. To map from the partner database to the TEMPVATRATE field, the target field (TEMPVATRATE) needed to be a string-type field. However, in order to map the TEMPVATRATE field to the ILD:2 VATP element (VAT rate percentage), you need to convert the TEMPVATRATE field to a number-type. You will use an extended rule to perform the conversion and the mapping.

Procedure Use this procedure to create an extended rule to convert the TEMPVATRATE field from a string-type to a number-type field and map the TEMPVATRATE field to the ILD:2 VATP element (VAT rate percentage).

Step	Action
1	Double-click the ILD:2 VATP element (VAT rate percentage) to access the Element Properties dialogue box.
2	Select the Extended Rule tab.
3	In the Rule list, type the following: <pre>#VATP = ATON (\$INVDETAIL.#TEMPVATRATE) ;</pre> <p>“ATON” is a numerical function that converts strings into real numbers, which enables you to convert a string type field to a number-type field. The “#VATP =” maps the resulting real number to the ILD:2 VATP element. The system executes this rule when it processes the ILD:2 VATP element.</p>
4	Click Compile to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it. Note The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list.
5	Click OK to exit the Element Properties dialogue box.

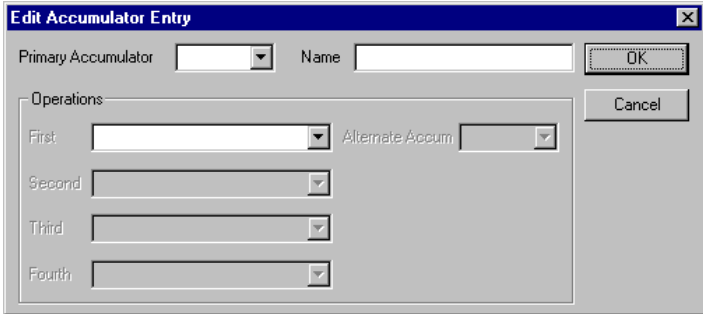
Mapping the TEMPLEXC Field

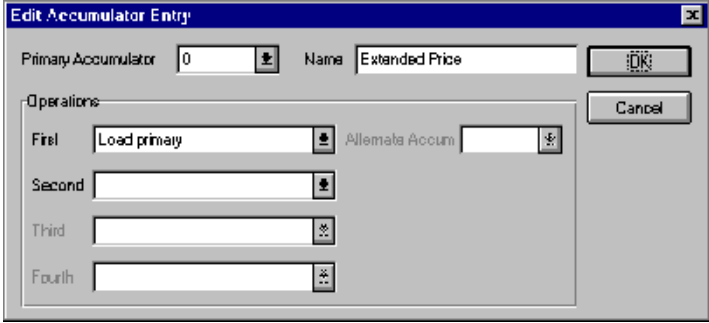
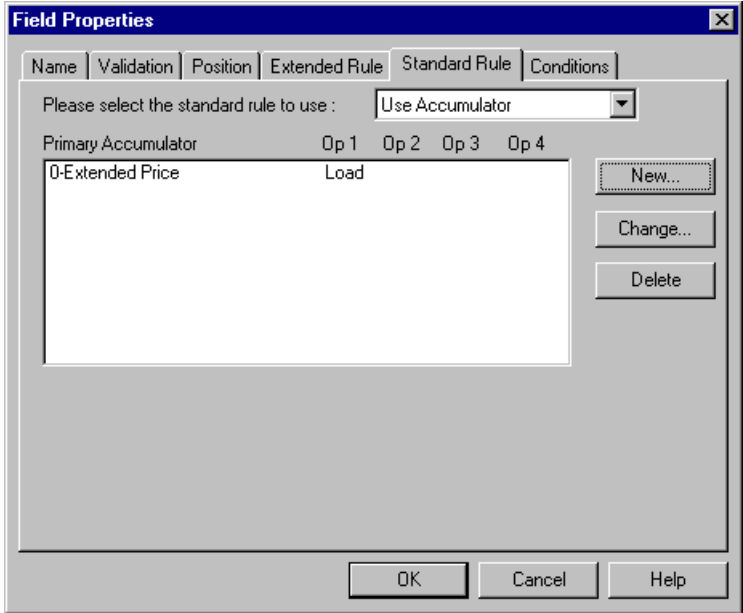
Overview

You need to calculate the extended price (quantity shipped multiplied by unit cost) for each line item and load that extended price into the TEMPLEXC field. Then, the TEMPLEXC field can be directly mapped to the ILD:2 LEXC element (Extended Line Cost (Excluding VAT)).

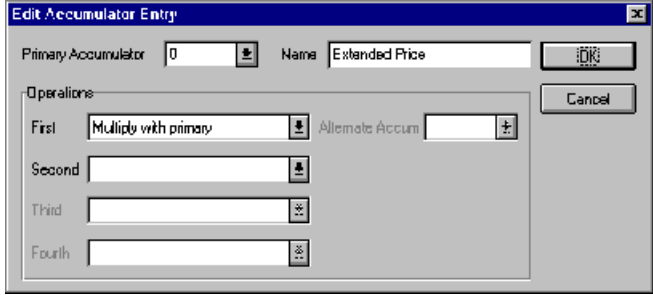
How to calculate the extended price

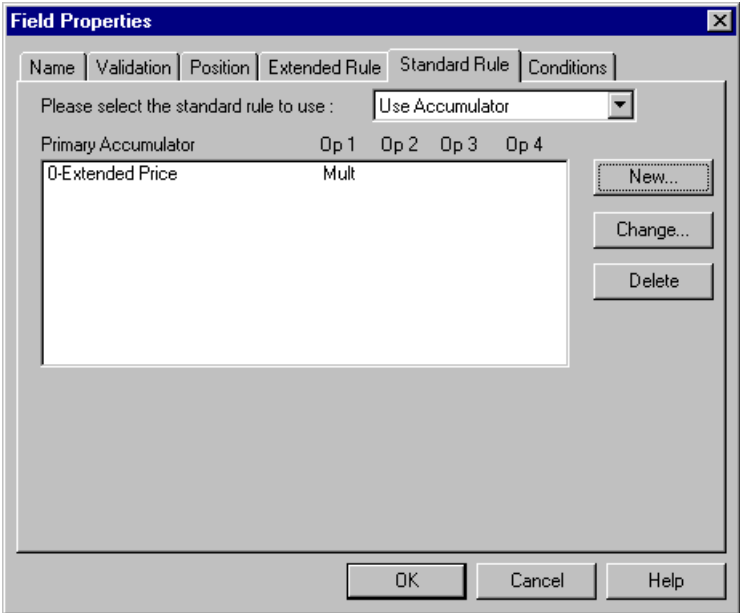
Use this procedure to multiply the quantity shipped for each line item by the unit price to obtain the extended price.

Step	Action
1	Double-click the QTYSHIP field (in the INVDETAIL segment). The Field Properties dialogue box is displayed.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, choose “Use Accumulator”
4	Click New to access the Edit Accumulator Entry dialogue box to create a new calculation for this element.
	
5	<p>From the Primary Accumulator drop-down list, select “0”. This number is a name by which the accumulator will be referred. Before any calculations are performed on an accumulator, its content is “0” (zero). When you use an accumulator, the system adds a new accumulator to the bottom of this drop-down list.</p> <p>Note There is <i>only one</i> set of accumulators for each map. This means that accumulator “0”, whether it is used in the Primary Accumulator or Alternate Accum field is the same accumulator with the same contents. If you assign calculations to accumulator “0” at the beginning of the map and then use accumulator “0” again later in the map, the content of that accumulator will be the result of the earlier calculation. Any additional calculations you assign to that accumulator will be performed on the contents resulting from an earlier calculation.</p> <p style="text-align: right;">(Continued on next page)</p>


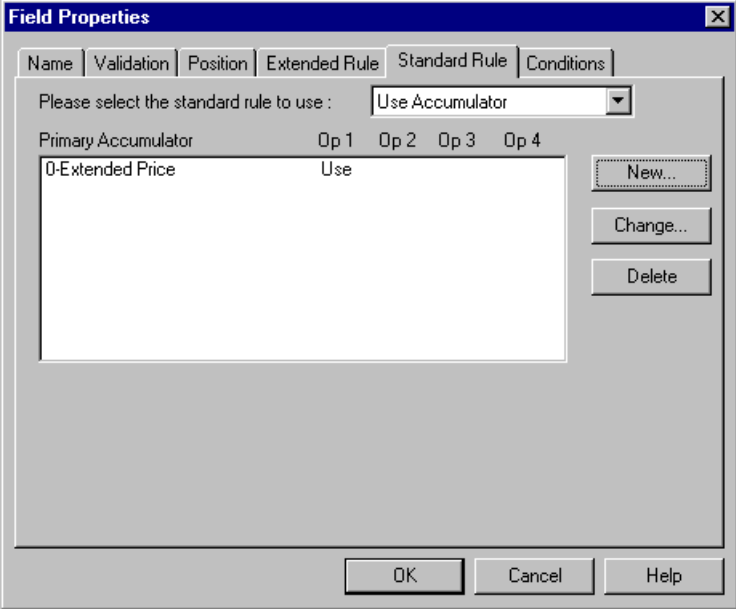
(Contd) Step	Action
6	In the Name box, type “Extended Price”. This is a descriptive alias that enables you to differentiate what the accumulators you create are used for.
7	<p>From the First list, select “Load primary.” This is the first operation that the system performs. This specifies that the system loads the contents of the field into the Primary Accumulator for each iteration of the INVDETAIL segment.</p> <p>Note The First box is active only after you select a Primary Accumulator.</p> <p>The Edit Accumulator Entry dialogue box should now look like this:</p> 
8	<p>Click OK to add the accumulator. The Standard Rule tab of the Field Properties dialogue box should now look like this:</p>  <p style="text-align: right; color: green;">(Continued on next page)</p>



(Contd) Step	Action
9	Click OK on the Field Properties dialogue box to add the standard rule to the QTYSHIP field.
10	Double-click the UNITPRICE field (in the INVDETAIL segment). The Field Properties dialogue box is displayed.
11	Select the Standard Rule tab to access standard rule options.
12	From the standard rule list, choose “Use Accumulator”
13	Click New to access the Edit Accumulator Entry dialogue box to create a new calculation for this element.
14	<p>From the list, select primary accumulator “0.”</p> <p>Note There is <i>only one</i> 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.</p>
15	<p>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 UNITPRICE field with the contents of the primary accumulator, and stores the result in the primary accumulator for each iteration of the INVDETAIL segment.</p> <p>Note The First box is active only after you select a Primary Accumulator.</p> <p>The Edit Accumulator Entry dialogue box should now look like this.</p>  <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
16	<p>Click OK to add the accumulator. The Standard Rule tab of the Field Properties dialogue box should now look like this:</p> 
17	Click OK on the Field Properties dialogue box to add the standard rule to the UNITPRICE field.
18	Double-click the TEMPLEXC field (in the INVDETAIL segment). The Field Properties dialogue box is displayed.
19	Select the Standard Rule tab to access standard rule options.
20	From the standard rule list, choose “Use Accumulator”
21	Click New to access the Edit Accumulator Entry dialogue box to create a new calculation for this element.
22	From the list, select primary accumulator “0.”

(Continued on next page)

(Contd) Step	Action
23	<p>From the First list, select “Use primary.” This specifies that the system loads the contents of the primary accumulator into the TEMPLEXC field.</p> <p>The Edit Accumulator Entry dialogue box should now look like this:</p> 
24	<p>Click OK to add the accumulator. The Standard Rule tab of the Field Properties dialogue box should now look like this:</p> 
25	<p>Click OK on the Field Properties dialogue box to add the standard rule to the TEMPLEXC field. This loads the extended price for the current line item into the TEMPLEXC field.</p>

How to map the extended line cost (excluding VAT)

After you calculate the extended price, you can map it from the TEMPLEXC field directly to the ILD:2 LEXC element (Extended Line Cost (Excluding VAT)).

Use this procedure to map the extended line cost (excluding VAT).

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 application field TEMPLEXC in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the ILD:2LEXC element (Extended Line Cost (Excluding VAT)). A line is displayed, visually illustrating the link between the TEMPLEXC field and the ILD:2 LEXC element.

The INVCommentGroup: TEMPVATSTL Record

Overview

Introduction

This section describes the actions necessary for you to map from each field in the TEMPVATSTL record (temporary storage for VAT subtotals). The TEMPVATSTL record contains temporary storage fields to enable VAT subtotal calculations. The temporary storage record is conditional and occurs up to 20 times per invoice translated (once for each possible VAT code subtotal).

BILLTOACCTTS field

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

DOCTYPTS field

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

INVNUMTS field

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

How to Map the SEQA Field

Overview The SEQA field will be used in an extended rule later in this tutorial, to calculate the subtotals for the various VAT tax rates used in this invoice, and to produce a summary of all the VAT rates. The SEQA field can be directly mapped to the STL SEQA:7 element (First Level Sequence Number), by using the Link function.

Procedure Use this procedure to map the first level sequence number (SEQA).

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 application field SEQA in the TEMPVATSTL record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the STL SEQA:7 element (First Level Sequence Number). A line is displayed, visually illustrating the link between the SEQA field and the STL SEQA:7 element.

How to Map the VATRATE Field

Overview The VATRATE field will be used in an extended rule later in this tutorial, to calculate the subtotals for the various VAT tax rates used in this invoice, and to produce a summary of all the VAT rates. The VATRATE field can be directly mapped to the STL VATP:2 element (VAT Rate Percentage), by using the Link function.

Procedure Use this procedure to map the VAT rate percentage (VATRATE).

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 application field VATRATE in the TEMPVATSTL record. When you click the field, the cursor changes to a “Link Arrow.”
3	Click the STL VATP:2 element (VAT Rate Percentage). A line is displayed, visually illustrating the link between the VATRATE field and the STL VATP:2 element.

How to Map the VATRATECODE Field

Overview The VATRATECODE field will be used in an extended rule later in this tutorial, to calculate the subtotals for the various VAT tax rates used in this invoice, and to produce a summary of all the VAT rates. The VATRATECODE field can be directly mapped to the STL CD12:2 element (VAT Rate Category Code), by using the Link function.

Procedure Use this procedure to map the VAT rate category code (VATRATECODE).

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 application field VATRATECODE in the TEMPVATSTL record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the STL CD12:2 element (VAT Rate Category Code). A line is displayed, visually illustrating the link between the VATRATECODE field and the STL CD12:2 element.

How to Map the NUMLINES Field

Overview The NUMLINES field will be used in an extended rule later in this tutorial, to calculate the subtotals for the various VAT tax rates used in this invoice, and to produce a summary of all the VAT rates. The NUMLINES field can be directly mapped to the STL NRIL element (Number of Line Items), by using the Link function.

Procedure Use this procedure to map the number of line items (NUMLINES).

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 application field NUMLINES in the TEMPVATSTL record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the STL NRIL element (Number of Line Items). A line is displayed, visually illustrating the link between the NUMLINES field and the STL NRIL element.

How to Map the LVLA Field

Overview The LVLA field will be used in an extended rule later in this tutorial, to calculate the subtotals for the various VAT tax rates used in this invoice, and to produce a summary of all the VAT rates. You will use the Link function to directly map the LVLA field to three different elements in the EDI file: the STL LVLA element (Lines Sub-Total (Excluding VAT)), the STL EVLA element (Extended Sub-Total Amount (Excluding VAT)), and the STL ASDA element (Extended Sub-Total Amount (After Settlement Discount Has Been Approved)).

Procedure Use this procedure to map the line subtotal amount (LVLA).

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 application field LVLA in the TEMPVATSTL record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the STL LVLA element (Lines Sub-Total (Excluding VAT)). A line is displayed, visually illustrating the link between the LVLA field and the STL LVLA element.
4	Click the application field LVLA in the TEMPVATSTL record again. When you click the field, the cursor changes to a "Link Arrow."
5	Click the STL EVLA element (Extended Sub-Total Amount (Excluding VAT)). A line is displayed, visually illustrating the line between the LVLA field and the STL EVLA element.
6	Click the application field LVLA in the TEMPVATSTL record again. When you click the field, the cursor changes to a "Link Arrow."
7	Click the STL ASDA element (Extended Sub-Total Amount (After Settlement Discount Has Been Approved)). A line is displayed, visually illustrating the link between the LVAL field and the STL ASDA element.

How to Map the VATA Field

Overview The VATA field will be used in an extended rule later in this tutorial, to calculate the subtotals for the various VAT tax rates used in this invoice, and to produce a summary of all the VAT rates. The VATA field can be directly mapped to the STL VATA element (VAT Amount Payable), by using the Link function.

Procedure Use this procedure to map the VAT Amount Payable (VATA).

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 application field VATA in the TEMPVATSTL record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the STL VATA element (VAT Amount Payable). A line is displayed, visually illustrating the link between the VATA field and the STL VATA element.

The INVCommentGroup: INVSUMMARY Record

Overview

Introduction

This section describes the actions necessary for you to map from each field in the INVSUMMARY (summary) record. The INVSUMMARY (summary) record contains the total invoice 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 needs to use to generate the invoice. Therefore, 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 needs to use to generate the invoice. Therefore, 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 needs to use to generate the invoice. Therefore, you will not perform any mapping operations on this field.

How to Map the TOTALAMOUNT Field

Overview For the system to create output segments, there must be at least one explicit link to an element in that segment, and data to support that link. Therefore, even though you are mapping the total amount with extended rules later in this tutorial, you must add an explicit link with an application field that contains data. The TOTALAMOUNT field can be directly mapped to the TLR TPSI element (Total Amount Inc Settlement), by using the Link function, so that the system will be able to create the TLR segment.

Procedure Use this procedure to map the total payable including settlement (total amount).

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 application field TOTALAMOUNT in the INVSUMMARY record. When you click the field, the cursor changes to a "Link Arrow."
3	Click the TLR TPSI element (Total Payable Inc. Settlement). A line is displayed, visually illustrating the link between the TOTALAMOUNT field and the TLR TPSI element.

Producing the VAT Summary

Overview To calculate the subtotals (STL) for the various VAT rates, you must first determine which VAT code is used for each line item. By grouping the line items that use the same VAT code, you can determine the VAT rate percentage, and then calculate the total VAT amount payable for each VAT code. Then, you can add the extended cost for each line item that uses the same VAT code together. Finally, by adding the total VAT amount payable (VATA) with the total extended cost (ASDA), you have calculated the sub-total amount (APSI) for each group of line items with the same VAT code.

To perform these calculations, you also will need to track the sequence numbers of each subtotal (STL), the VAT code and corresponding VAT percentage rate for each STL, and the total number of line items that use each VAT code.

This table provides an example of an application file that contains six line items that use three different VAT codes.

Line Item Number	Item	VAT Code	Extended Cost (Quantity Shipped * Unit Price)
1	Item 1	S	100
2	Item 2	E	50
3	Item 3	S	200
4	Item 4	Z	70
5	Item 5	S	150
6	Item 6	E	300

This table provides the corresponding VAT percentage rates for the VAT codes in the sample application file.

VAT Code	Description	VAT Percentage Rate
S	Standard rate	17.5
E	Export item	0
Z	Zero rate	0

(Continued on next page)

Overview (contd)

Using this information, you can calculate the require VAT summary information.

STL Sequence Number	VAT Code	Number of Line Items Using this VAT Code	Total Extended Cost (ASDA)	Total VAT Amount (VATA)	Subtotal Amount (APSI)
1	S	3	450	18	468
2	E	2	350	0	350
3	Z	1	70	0	70

Extended rules

To perform the required VAT calculations in this tutorial, you will use two extended rules. The first extended rule you will define is an On End rule on the INPUT (application file) group that will be executed when the system concludes its processing of the application file. For each VAT code used, the first rule will calculate the total VAT amount payable and the extended cost for each set of line items. The second extended rule you will define is on the STL APSI element, and will be executed when the system concludes its processing of that element. The second rule will add the total VAT amount payable (VATA) to the total extended cost (ASDA), to calculate the sub-total amount (APSI) for each group of line items with the same VAT code.

Extended rule on the INPUT group

The first extended rule (on the INPUT group) will involve many different calculations. So, you determine that it will be easiest if you break the various tasks involved into manageable pieces, and write the rule for that part before building all the parts into the whole rule. This process is outlined in the steps below.

Note

The extended rules are indicated with `courier new` typeface (➤ is used for a line continuation character).

(Continued on next page)

Extended rule on the INPUT group (contd)

This table illustrates the extended rule.

Stage	Description
1	<p>You need to define two temporary variables (“i” and “j”), a variable defined to count the number of line items (“invlimit”), and a variable to count the number of STL segments (“stllimit”) that is initialised to zero.</p> <pre>integer i,j,invlimit,stllimit; invlimit = count(\$INVDETAIL[*]); stllimit = 0;</pre> <p>You need to visit each iteration of the INVDETAIL record.</p> <pre>i=1; while i<=invlimit do begin . . i=i+1 end</pre>
2	<p>For each iteration of the INVDETAIL record, you need to determine the VAT code used, and ascertain whether or not a TEMPVATSTL (STL) record exists for that VAT code.</p> <pre>j=1; while j<=stllimit do begin if \$INVDETAIL[i].#VATCODE = ➔\$TEMPVATSTL[j].#VATRATECODE then begin . . end j=j+1; end</pre>
3	<p>If a TEMPVATSTL record does exist for that VAT code, you need to add the extended cost for that line item into the total extended cost for this STL, and increase the number of line items in this STL by one.</p> <pre>begin \$TEMPVATSTL[j].#LVLA = \$TEMPVATSTL[j].#LVLA + ➔\$INVDETAIL[i].#TEMPLEXC; \$TEMPVATSTL[j].#NUMLINES = ➔\$TEMPVATSTL[j].#NUMLINES + 1 break; end</pre> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Stage	Description
4	<p>If a TEMPVATSTL record does not exist for that VAT code, you need to create an STL for that VAT code, set the sequence number to the line item number, set the VAT percentage, set the STL number, and set the total extended cost of that line item.</p> <pre> \$TEMPVATSTL[j].#SEQA = j \$TEMPVATSTL[j].#VATRATE = ↪ATON(\$INVDETAIL[i].#TEMPVATRATE); \$TEMPVATSTL[j].#VATRATECODE = \$INVDETAIL[i].#VATCODE; \$TEMPVATSTL[j].#LVLA = \$INVDETAIL[i].#TEMPLEXC; \$TEMPVATSTL[j].#NUMLINES = 1; </pre>
5	<p>Finally, you need to calculate the VAT amount that is payable for each STL.</p> <pre> j=1 while j<= stllimit do begin \$TEMPVATSTL[j].#VATA = (\$TEMPVATSTL[j].#LVLA * ↪\$TEMPVATSTL[j].#VATRATE) / 100.0; j = j + 1; end </pre>

How to create an extended rule to perform VAT calculations

Now you need to combine all the pieces that you wrote into one extended rule, including the necessary keywords to define loops. The extended rule should be executed when the system concludes its processing of the application file (INPUT group).

Use this procedure to create the extended rule.

Step	Action
1	Click the INPUT group with the right mouse button to access the shortcut menu.
2	From the shortcut menu, select Extended Rules to display the Positional File Properties dialogue box (Loop Extended Rules tab).
3	Select On End to specify that the rule will be executed when the loop terminates. The system executes this rule when it has concluded its processing of the application file.

(Continued on next page)

(Contd) Step	Action
4	<p>In the Editor box, type the following (or copy from the PET_TINV.MAP provided with the tutorial data in the C:\GENSRVNT\TUTORIALS\ subfolder.</p> <pre> integer i,j,invlimit,stllimit; invlimit = count(\$INVDETAIL[*]); stllimit = 0; i=1; while i <= invlimit do begin j=1; while j <= stllimit do begin if (\$INVDETAIL[i].#VATCODE = \$TEMPVATSTL[j].#VATRATECODE then begin \$TEMPVATSTL[j].#LVLA = \$TEMPVATSTL[j].#LVLA + ↳\$INVDETAIL[i].#TEMPLEXC; \$TEMPVATSTL[j].#NUMLINES = ↳\$TEMPVATSTL[j].#NUMLINES + 1; break; end j=j+1 end if j > stllimit then begin \$TEMPVATSTL[j].#SEQA = j; \$TEMPVATSTL[j].#VATRATE = ↳ATON(\$INVDETAIL[i].#TEMPVATRATE); \$TEMPVATSTL[j].#VATRATECODE = ↳\$INVDETAIL[i].#VATCODE; \$TEMPVATSTL[j].#LVLA = \$INVDETAIL[i].#TEMPLEXC; \$TEMPVATSTL[j].#NUMLINES = 1; stllimit = j; end i=i+1; end j=1; while j<= stllimit do begin \$TEMPVATSTL[j].#VATA = (\$TEMPVATSTL[j].#LVLA * ↳\$TEMPVATSTL[j].#VATRATE) / 100.0; j = j + 1; end </pre> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Step	Action
5	<p>Click Compile to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.</p> <p>Note The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warning generated in the compilation process are displayed in the Compile Errors list.</p>
6	Click OK to add the extended rule to the INPUT group (application file).

How to create an extended rule to calculate the amount payable

The second extended rule will add the total VAT amount payable (VATA) with the total extended cost (ASDA), to calculate the sub-total amount (APSI) for each group of line items with the same VAT code. The extended rule should be executed when the system concludes its processing of the STL APSI element.

Use this procedure to create the extended rule.

Step	Action
1	Double-click the STL APSI element (Payable Sub-Total Amount Including Settlement Discount) to access the Element Properties dialogue box.
2	Select the Extended Rule tab.
3	<p>In the Rule box, type the following:</p> <pre>#APSI = #ASDA + #VATA;</pre> <p>The system executes this rule when it processes the STL APSI element.</p>
4	<p>Click Compile to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.</p> <p>Note The Compile function will compile the rule interactive, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process are displayed in the Compile Errors box.</p>
5	Click OK to add the extended rule to the STL APSI element.

Producing the Invoice Trailer

Overview

You already used an explicit link (the TOTALAMOUNT field to the STL TPSI element) so the system will be able to create the TLR segment (Invoice Trailer). Now, you need to calculate the total amount (extended cost) of all the line items in the invoice (LVLT) and the total VAT amount payable for all the line items in the invoice (TVAT). This calculation is the sum of all the STL segments that you calculated in the last section of this tutorial.

Also, because you are not using settlement discounts, you can set the EVLT (Total Extended Amount Excluding Settlement Discount) and the ASDT (Total Amount Excluding VAT Including Settlement) equal to the LVLT (Lines Total Amount (Excluding VAT)). You can also set the TPSI equal to the ASDT (Total Amount Excluding VAT Including Settlement) plus the TVAT (Total VAT Amount Payable).

Extended rule to calculate totals

You will use an extended rule on the TLR NSTL element (Number of Total Segments) to calculate these totals. This rule will involve many different calculations. So, you determine that it will be easiest if you break the various tasks involved into manageable pieces, and write the rule for that part before building all the parts into the whole rule. This process is outlined in the steps below.

Note

The extended rules are indicated with `courier new` typeface (↵ is used for a line continuation character).

This table illustrates the extended rule.

Stage	Description
1	<p>You need to define a variable to count the number of TEMPVATSTL records ("limit"), define a temporary variable ("i") that is set to one, and initialise the TLR LVLT and the TLR TVAT elements to zero.</p> <pre>integer i,limit; limit = count(\$TEMPVATSTL[*]); i=1; #LVLT = 0; #TVAT = 0;</pre>
2	<p>You need to visit each iteration of the TEMPVATSTL record.</p> <pre>while i <= limit do begin i = i + 1; end</pre> <p style="text-align: right;">(Continued on next page)</p>

(Contd) Stage	Description
3	<p>You need to calculate the total amount (extended cost) of all the line items in the invoice (LVLT) and the total VAT amount payable for all the line items in the invoice (TVAT).</p> <pre>#LVLT = #LVLT + \$TEMPVATSTL[i].#LVLA; #TVAT = #TVAT + \$TEMPVATSTL[i].#VATA;</pre>
4	<p>You need to set the NSTL (Number of Total Segments) equal to the number of TEMPVATSTL records.</p> <pre>#NSTL = limit;</pre>
5	<p>You need to set both the EVLT (Total Extended Amount Excluding Settlement Discount) and the ASDT (Total Amount Excluding VAT Including Settlement) equal to the LVLT (Lines Total Amount (Excluding VAT)).</p> <pre>#EVLT = #LVLT; #ASDT = #LVLT;</pre>
6	<p>Finally, you need to set the TPSI equal to the ASDT (Total Amount Excluding VAT Including Settlement) plus the TVAT (Total VAT Amount Payable).</p> <pre>#TPSI = #ASDT + #TVAT;</pre>

How to create an extended rule to calculate TLR totals

Now you need to combine all the pieces that you wrote into one extended rule, including the necessary keywords to define loops.

Use this procedure to create the extended rule.

Step	Action
1	Double-click the TLR NSTL element (Number of Total Segments) to access the Element Properties dialogue box.
2	Select the Extended Rule tab.

(Continued on next page)

(Contd) Step	Action
3	<p>In the Rule box, type the following:</p> <pre>integer i,limit; limit = count(\$TEMPVATSTL[*]); i=1; #LVLV = 0; #TVAT = 0; while i <= limit do begin #LVLV = #LVLV + \$TEMPVATSTL[i].#LVLA; #TVAT = #TVAT + \$TEMPVATSTL[i].#VATA; i=i+1; end #NSTL = limit; #EVLV = #LVLV; #ASDT = #LVLV; #TPSI = #ASDT + #TVAT;</pre> <p>The system executes this rule when it processes the TLR NSTL element.</p>
4	<p>Click Compile to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.</p> <p>Note The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process are displayed in the Compile Errors box.</p>
5	Click OK to add the extended rule to the TLR NSTL element.

Finalising the Map

Overview

Introduction

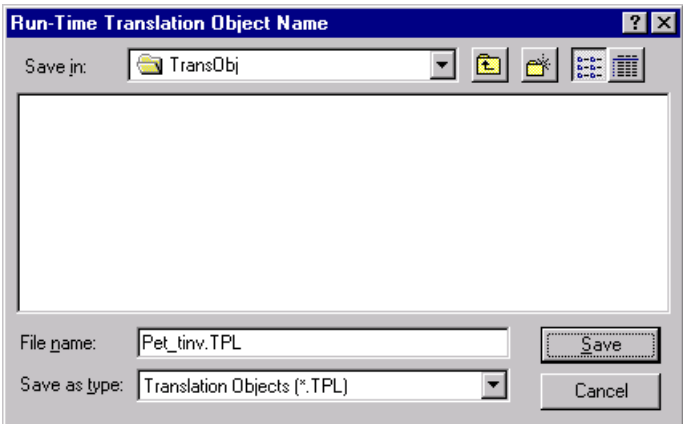
After you create the import map, define your application file format, and perform the appropriate mapping operations, you need to finalise the map. To complete the mapping process, you need to save the map, compile the translation object, print and review the Gentran:Server for Windows report, and test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


How to Compile the Map

Overview The Compile function compiles the map and generates a translation object. The map that you created using Gentran:Server for Windows (PET_TINV.MAP) 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 for Windows system before you can use it. You will use the Compile function after the map is completed and saved.

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

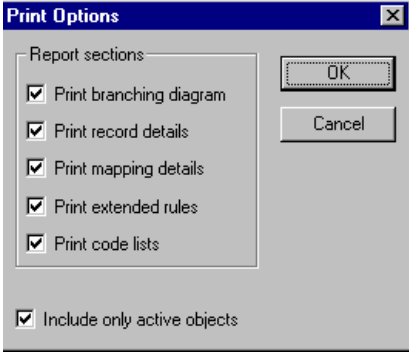
Step	Action
1	From the File menu, select Save to save the source map, prior to using the Compile function.
2	From the File menu, select Compile to display the Run-Time Translation Object Name dialogue box. 
3	The File name box will already contain “PET_TINV.TPL”. This is the name of the translation object, using the default .TPL file extension and our recommendation that you name the translation object (.TPL file) the same filename (one to eight characters long) as you named the map (.MAP file). Preserving the same filename (with different file extensions) means that the relationship between the source map and the compiled translation object will remain evident. <p>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.</p> <p style="text-align: right; color: green;">(Continued on next page)</p>

(Contd) Step	Action
4	<p>Change the Drive and Folder where the compiled translation object will be stored, if necessary.</p> <p>Caution Do <i>not</i> 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.</p>
5	<p>Click Save and the system compiles the map and generates a translation object. The Compile Errors dialogue box is displayed. Verify that no errors occurred and click OK to exit the dialogue box. The date on which the translation object was compiled is automatically loaded into the Compiled on box on the Translation Object Details dialogue box.</p>  <p>The screenshot shows the 'Translation Object Details' dialog box. The 'Summary' section includes: Author: Sterling, Description: PET T ANA001 9 INVOIC 1, and Translation Object Function: Import. The 'Flags' section has three checkboxes: System (unchecked), Use Configurable Trimming (unchecked), and Gentran:Server for Windows 2.x Compatible Rule Execution (unchecked). The 'Version Control' section shows Major version: 1, Minor version: 0, and Compiled on: 25-Oct-1995. The 'EDI Associations' section has a table with Input and Output columns. The Output column contains: A Tradacoms, ANA001 TRADACOMS UNIT, INVOIC INVOICE DETAILS, 9, and INVOIC.</p>
6	<p>From the File menu, select Save to save the source map with the Compiled on date.</p>
7	<p>You must register this translation object with the Gentran:Server for Windows system before you can use it.</p> <p>Reference See the Gentran:Server for Windows documentation for more information on registering a translation object.</p>

How to Print the Report

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

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

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

Testing the Translation Object

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

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

Step	Action
1	Register the HEADER_T.TPL and PET_TINV.TPL translation objects with Gentran:Server for Windows.
2	If you have not already done so, import the PETZONE3.PAR partner relationship into Gentran:Server for Windows. The default full path for the partner file is: C:\GENSRVNT\TUTORIAL\PETZONE3.PAR
3	Import the division lookup table (INVATCO.LKP) for the system partner (Internal System User) into Gentran:Server for Windows. The default full path for the lookup table is: C:\GENSRVNT\TUTORIAL\INVATCO.LKP
4	Verify (in Partner Editor) that the PET T ANA001 9 INVOIC Import translation object is selected for the Outbound Relationship.
5	Ask your system administrator to add the system import translation object (HEADER_T.TPL) to the System Configuration program.
6	Use the Import option in Gentran:Server for Windows to process the data file (PET_TINV.TXT) through the translation object. The data file will be located in the TUTORIAL subfolder under the folder where Gentran:Server for Windows is installed.
7	After the message is translated, it will be 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, importing files and viewing raw EDI data.

See the Gentran:Server for Windows *Administration Guide* for more information about how your system administrator will need to modify the system configuration program.

How to Populate the SDT and CDT Segments

Introduction

The T_INVGRP.MAP group build map that we provide for you (on the Gentran:Server product CD) to use for outbound TRADACOMS messages contains a standard rule on both the SDT01 and CDT01 elements. These standard rules look up information by the Reference Code from a Partner Location table, and the information that the system finds in this partner table is used to populate the SDT and CDT segments in the Outbound TRADACOMS Group.

If you want to populate the SDT01 and CDT01 elements in your outbound maps with data, then you need to create two partner location tables are needed—one for the SDT segment and one for the CDT segment.

SDT elements mapped

This table describes each partner location table field and its corresponding SDT element (from the outbound SDT segment) that is populated with data.

Location Table Field	Outbound Group Build Map SDT Element
Name	SNAM - SUPPLIER'S NAME
Address1	SAD0 - SUPPLIER'S ADDRESS LINE 1
Address2	SAD1 - SUPPLIER'S ADDRESS LINE 2
Address3	SAD2 - SUPPLIER'S ADDRESS LINE 3
City	SAD3 - SUPPLIER'S ADDRESS LINE 4
Zip	SAD4 - SUPPLIER'S POST CODE
Reference Code 2	VAT0 - SUPPLIER'S VAT REGISTRATION NUMBER

CDT elements mapped

This table describes each partner location table field and its corresponding CDT element (from the outbound CDT segment) that is populated with data.

Location Table Field	Outbound Group Build Map SDT Element
Name	CNAM - SUPPLIER'S NAME
Address1	CAD0 - SUPPLIER'S ADDRESS LINE 1
Address2	CAD1 - SUPPLIER'S ADDRESS LINE 2
Address3	CAD2 - SUPPLIER'S ADDRESS LINE 3
City	CAD3 - SUPPLIER'S ADDRESS LINE 4

(Continued on next page)

(Contd) Location Table Field	Outbound Group Build Map SDT Element
Zip	CAD4 - SUPPLIER'S POST CODE
Reference Code 2	VAT0:2 - SUPPLIER'S VAT REGISTRATION NUMBER

Procedure

Use this procedure to set up your partner to populate the SDT and CDT elements in the Outbound Tradacoms Group.

Reference

See the Gentran:Server User's Guide for more information on using partner locations.

Step	Action
1	From the Tools menu, select Partner Editor . System Response The system displays the Partner Editor dialog box.
2	Select the partner profile for which you want to create the location table and click Outbound . System Response The system displays the Outbound Relationship dialog box.
3	From the Relationship list, select the appropriate relationship and click Group . System Response The system displays the Outbound Group Select dialog box.
4	From the Groups list, select the appropriate relationship and click Edit . System Response The system displays the Outbound Group Entry dialog box.
5	Click Envelope . System Response The system displays the Outbound BAT Functional Group dialog box.
6	Note the values listed in Field 3 and Field 4 on this dialog box.
7	Select the partner profile for which you want to create the location table and click Locations . System Response The system displays the Location Select dialog box. (Continued on next page)

(Contd) Step	Action
8	<p>Click New.</p> <p>System response The system displays the Location Entry dialog box.</p>
9	<p>In the Name box, type a unique identifier that defines this location within the partner.</p>
10	<ul style="list-style-type: none"> ▶ For the Reference Code 1 box for the SDT Partner Location Table, type the value from Field 3 on the Outbound BAT Functional Group dialog box. ▶ For the Reference Code 1 box for the CDT Partner Location Table, type the value from Field 4 on the Outbound BAT Functional Group dialog box.
11	<p>Complete the remaining boxes, as necessary.</p> <p>Note If you use Reference Code 2, it must also be unique.</p>
12	<p>Click Save.</p> <p>System response Saves the location information and displays a blank Location Entry dialog box.</p>
13	<p>To create the second location table, repeat steps 8 through 12.</p>
14	<p>Click Exit to return to the Partner Editor dialog box.</p>

Glossary

?In Documents	This Gentran:Server for Windows browser contains a list of messages 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 for Windows browser contains a list of messages 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 message.
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, <i>you</i> 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 organisation for the United States.
application file	If you are creating an import or export map, you must define your application to Gentran:Server for Windows. In Gentran:Server for Windows terminology, your application file is also referred to as a <i>fixed-format file</i> or a <i>positional file</i> . Your application file must contain all the information that you either need to extract from your partner’s message (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	Application systems are computer systems, outside of EDI, designed to fulfil specific business functions. These functions 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 messages (transaction sets), interchanges, translation objects, etc. A document browser displays only messages.
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 using right-handed or left-handed mouse operation.
code list tables	Code list tables 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 to and received to one telephone number in one continuous period of connection. One session could include sending two or three interchange envelopes to a network, each for a different trading partner.
Communicator	The Communicator is the Gentran:Server for Windows communications software that enables you to send, receive, resend, establish communication port definitions, establish communication profiles, establish communication scripts, and view and delete communications sessions.

compile	This function compiles the map and creates a translation object. The map that you create using Gentran:Server for Windows is a <i>source map</i> . When that source map is compiled, the result is a <i>compiled translation object</i> . This translation object will need to be registered with the Gentran:Server for Windows system before you can use it.
compliant	Compliant means that the message 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 condition.
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 default is a predefined value that programs use, <i>unless</i> you specifically override them.

dialogue box	A dialogue box (or dialogue) contains additional information or options that require action. If you specify any options, you can select a button to execute the command. Some dialogue boxes display warnings or messages indicating why a requested task cannot be accomplished. If the dialogue box has a title bar, you can move the dialogue box to another location on your desktop.
document	<p>A document is one transaction set containing actual data and treated as a single entity. The amount of data does not affect whether a set of data is a document (message), 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.</p> <p>If a trading partner sends you a communication containing ten purchase orders, you have received ten documents. If the communication contained fifteen invoices, you have received fifteen documents.</p>
double-click	To double-click, 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 using right-handed or left-handed mouse operation.
drop-down combo box	A drop-down combo box includes a text box and an attached, integrated drop-down list. Select an option from the drop-down list, or if the correct option is not available, type it in the text box.
drop-down list box	A drop-down list box is displayed as a rectangular box with the current choice. Click the down-arrow icon or press the UP or DOWN arrow keys to move through the options, and a list of available choices is displayed. Scroll bars are displayed if there are more items than can fit in the box.
EANA	International Article Numbering Association, Brussels, Belgium.
EDI	Electronic Data Interchange (EDI) is the process by which companies can exchange business messages directly from application to application by computer, without producing paper messages.
EDI standard	An EDI standard consists of the rules for turning a business message into an EDI message.
EDIA	The Electronic Data Interchange Association, formerly known as the Transportation Data Coordinating Committee (TDCC).

EDIFACT EDIFACT (Electronic Data Interchange for Administration, Commerce, and Transport) is the standards organisation of the United Nations Economic and Social Council.

element An element is the smallest piece of usable information defined by the standards. Examples might include a quantity, unit price, or description. An individual element can have somewhat different meanings depending on context. Therefore, elements are normally not considered to have useful meaning until they are combined into segments. An element is the EDI map component that is mapped (linked) to a corresponding application field to move data to and from the EDI file.

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

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

envelope An envelope is 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 message) is contained within a transaction set envelope.
- **Functional Group:** A functional group envelope is an envelope containing related business messages. The standards define which transaction sets should be placed together into a functional group envelope.
- **Interchange Envelope:** An interchange envelope is 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 messages 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	A field 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 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 for Windows 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 reduce the font size if you need to view more of the map on your monitor, enlarge the font size, or change the type and style to be more easily readable by you.
functional group	A functional group is 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 group is 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 are visually represented the same way as other groups and sub-groups in Gentran:Server for Windows.

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 messages, to the EDI standard-formatted messages that your partners expect to receive from you. An import map is necessary for outbound processing.
import translation object	See import map.
In Documents	This Gentran:Server for Windows browser contains a list of messages that the system received but that have not yet been processed by the user. Once the messages are processed (printed or exported) the messages will be transferred to the In Drawer.
inbound mapping	The system translates your trading partner's EDI standard-formatted business message to your application file format, so you can receive messages from your partners. To translate inbound data, you need to create an export map.
In Drawer	This Gentran:Server for Windows browser contains a list of messages that were received and processed.
integer	An integer is a number that has an implied decimal point (e.g., "2.01" is formatted as "201").
interchange	The interchange contains all the functional groups of messages (transaction sets) sent from one sender to one receiver in the same transmission.
Interchanges browser	This Gentran:Server for Windows 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 messages within the interchanges. You can access the Interchanges browser at any time.
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	Literal constants 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	A location table is created in Partner Editor to contain address-related information about the partner.
lookup table	A lookup table is created in Partner Editor 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 <i>or</i> the outer loop. For example, if the Y loop is nested within the X loop, and the Y loop has cycled through fifteen iterations and the X loop has cycled through three 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. Depending on whether or not 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.
map	A map is a set of instructions you define in Gentran:Server for Windows that indicates the corresponding relationship between your application file and the EDI standards, and defines how the system should translate data.
menu	A menu is 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	The menu bar contains the list of menus for the application.

message	<p>A message is one transaction set containing actual data and treated as a single entity. The amount of data does not affect whether something is a message (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 message.</p> <p>If a trading partner sends you a communication containing ten purchase orders, you have received ten documents. If the communication contained fifteen invoices, you have received fifteen documents.</p>
network	<p>Network is 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.</p>
number	<p>A number-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.</p>
ODETTE	<p>The Organization for Data Exchange by Tele-Transmission in Europe.</p>
Out Documents	<p>This Gentran:Server browser contains a list of messages that are ready to be sent. After the messages are successfully sent, they are automatically transferred to the Out Drawer.</p>
outbound mapping	<p>The system translates your application file format to EDI standard formats, so you can send messages to your partners. To translate outbound data, you need to create an import map and a system import map in Gentran:Server.</p>
Out Drawer	<p>This Gentran:Server browser contains a list of messages that were successfully sent by the system.</p>
partner	<p>A partner is another firm with which your company trades messages. Also referred to as a trading partner.</p>
Partner Editor	<p>This Gentran:Server function enables you to define, edit, and delete all partner information for your company and all of your trading partners.</p>
positional file	<p>See application file.</p>

post	This Gentran:Server for Windows function enables you to move compliant messages from the WorkSpace to the Out Documents.
predefined	On a screen entry translation object, predefined is a default value for a particular element. You can change the value as necessary.
printout	A printout is produced when data received from a trading partner passes through a print translation object. You do not have to actually print the data; you can also send a printout 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 specialised definition. Gentran:Server for Windows specifies that only one-to-one (no loops) or many-to-many (loop) mapping relationships are valid.
qualifier	A qualifier 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 for Windows function enables you to manually initiate a communications session to receive data from Commerce:Network through an invocation of the Connection Manager.
record	A 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 for Windows function enables you to create a turnaround message (using the Turn Around translation object in the partner relationship) in response to one or more selected message(s) in the In Drawer.
responsible agency	A responsible agency is an organisation that develops and updates standards for EDI communications. These organisations include ANSI, EDIA, AIAG, UCS, EDIFACT, ODETTE, and VICS.
segment	A segment is a group of related elements or composite data elements that are combined to communicate useful data. Segments are defined by the EDI standards. A segment can occur once or can repeat multiple times. For example, a catalogue 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 for Windows 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 for Windows function enables you to manually start a communications session to send data to Commerce:Network through an invocation of Connection Manager. Only selected messages are enveloped and sent. If no messages are selected, <i>all</i> messages will be sent. Successfully sent messages 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 dialogue. Syntax tokens enable you to define what type of characters should be used while compliance checking each element/field (e.g., alphanumeric within a certain range, numeric within a certain range, etc.).
system import map	The translator uses the system import map to determine which trading relationship (established in Partner Editor) corresponds to each message in the application file, so the system will know which import map to use to process the message. 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 for Windows system.
TDCC	See EDIA.
TDF	TDF (Transaction Data File) is a file that serves as a filter between your message files and the Gentran:Server for Windows 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 messages that are received by Gentran:Server for Windows 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	The toolbar 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	A trading partner is another firm with which you company trades messages. Also referred to as a partner.
transaction set (message)	A business form as defined by the standards. Examples include an ANSI 850 purchase order or a TRADACOMS 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. Transaction set is also known as “message” in Europe.
translation object	<p>A translation object is 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.</p> <p>Inbound Translation Objects:</p> <ul style="list-style-type: none">• Turn Around: This translation object is used when a message is received, to create the natural response message that contains as many elements from the received message as possible.• Export File: This translation object indicates that when a message is received, it will be exported to a specified file format.• Print: This translation object is used to print messages. <p>Outbound Translation Objects:</p> <ul style="list-style-type: none">• Import: This translation object is used to import data from an application file.• Print: This translation object is used to print messages.• Data Entry: This translation object is used to enter data into the Message 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 transmission chain is a path an EDI communication could follow, including one company, one trading partner, and one or more network services.
turnaround document	A turnaround document is a message into which data elements from the source message have been automatically transferred using a turnaround map.
turnaround map	A series of instructions that the system uses to create a turnaround message (a logical response message to the source) from an inbound (source) message, by transferring data from the source message to elements in the target message (translation object).
UCS	UCS (Uniform Communications Standard) is the standard used by the grocery industry.
update	<p>This standard rule enables you to update a specific field in a document record, envelope segment, interchange, group, current partner, or message (if the map side format type is EDI), with the contents of the element or field.</p> <p>Caution</p> <p>Please note that this function updates the internal Gentran:Server for Windows database tables. We recommend that you use this function only if you are sure you really want to update the internal database tables.</p> <p>Typically, you will <i>only</i> use this function if you want to update the document name and reference in the document table. Any other use of this function could have disastrous consequences!</p>
use accum	<p>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, e.g., quantity, price, etc.). This function also enables you to map the accumulated total into a control total field, and use accumulators.</p> <p>Accumulators are used generally for counting the occurrences of a specific element or generating increasing or sequential record or line item numbers.</p>
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.
value-added network	Value-added network (VAN) is also known as a third party network. A service, such as Commerce:Network, that accepts and holds transmissions from companies until it is convenient for a trading partner to accept them.
version	Each standards-setting body updates its standards on a regular basis. Each formal update is referred to as a version.
VICS	VICS (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” messages. It also contains recently imported or screen entry messages.
X12	The ANSI committee that sets and publishes standards for EDI.
