Gentran:Server® for UNIX® and Workstation

Application Integration User's Guide
Version 6.1

Sterling Commerce
An IBM Company

Copyright Notice

August 2004

Table of Contents

Designing your Map

Defining Trading Partnership Rules for Application Files

Using Standard Rules

Common Statements and Examples

Alphabetic Language Reference

Working with Trading Partnerships

Overview

Trading Partnership Administration

Creating Trading Partnerships

Using Standard Cross Reference Tables

Creating Optional Records

Running Translation

About This Guide

Contents

Introduction

Welcome Welcome to the Gentran:Server® for UNIX® and Workstation Application Integration subsystem. Application Integration is Sterling Commerce's mapping software.

Application Integration provides you with the tools you need to develop maps that translate:

◗ Your application files to an EDI standard format for documents you send to your trading partners (outbound mapping)

◗ An EDI standard format to your application format for documents that you receive from your trading partners (inbound mapping)

◗ An EDI standard format to another EDI standard format and an application format to another application format.

We believe that you will find this software and the supporting materials easy to use and directly beneficial to your business.

What's in This Manual

Assumptions

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.

◗ Windows-specific attributes, such as dialog boxes, icons, windows, and buttons.

This manual also assumes that you are familiar with your internal application format, data mapping concepts, and the Gentran:Server for UNIX or Workstation product.

Description of contents

This manual is intended to assist you in performing various tasks in the Application Integration subsystem. This task-oriented approach is intended to answer any questions you may have about the Application Integration subsystem with step-by-step instructions.

The *Gentran:Server for UNIX and Workstation Application Integration User's Guide* is organized into chapters. This table describes the contents.

Chapter	Description
About This Guide	Introduces the content, organization, and conventions in this guide. This chapter also describes how to get technical support.
Application Integration Basics	Introduces you to mapping and to the Application Integration subsystem. It contains instructions for starting the subsystem and for opening and closing maps.
Designing your Map	Explains how to create a new map; how to customize, format, and link the map components; and how to compile a map to create a translation object.
Using Standard Rules	Explains how to use the Application Integration standard rules. This chapter includes examples of using certain standard rules.
Using Extended Rules	Defines what extended rules are and when they are used, and explains the proprietary programming language syntax.
	(Continued on next page)

(Contd) Chapter	Description
Working with Trading Partnerships	Describes the records in a Trading Partnership and explains how to create and maintain these records, how to make changes to several records at once (mass changes), and how to maintain the Standard Cross Reference Table.
Running Translation	Explains how to run inbound and outbound translation from the Translation menu.
Archiving Translation Data	Describes how to choose the default archiving options and run the archiving program, work with archived data, and reconcile functional acknowledgement.
Using the Task Scheduler	Contains information about running batch files or scripts that contain Gentran:Server command-line commands and operating-system commands at a specific day and time.
Appendix A, Error Messages	Describes the Application Integration error messages and specifies corrective actions.
Glossary	Contains mapping and Gentran:Server terms and concepts.
Index	Provides you with a list of terms, concepts, functions, and processes to enable you to quickly find them in this guide.

Online Help System

Additional documentation for Gentran:Server for UNIX is contained in the online Help system. The online Help documentation includes all the dialog box element definitions, detailed processing information, and all the "how to" information contained in this manual.

Related Publications

Gentran:Server documentation

This table describes additional documentation for the Gentran:Server software.

Document	Description
Upgrade and Data Conversion Guide	Instructions for upgrading from previous versions of Gentran:Server Workstation and Gentran:Server for UNIX. Also includes instructions for converting the files that are part of the upgrade.
Installation Checklist	Description of the recommended sequence in which you should install and configure system components.
Gentran:Server for UNIX Getting Started Guide	Instructions for installing the Gentran:Server software and performing setup tasks, such as setting up security. Instructions for starting and exiting Gentran:Server and for setting preferences and default values. Also includes instructions for checking files in and out and saving files.
Gentran:Server Workstation Getting Started Guide	Instructions for installing the Gentran:Server Workstation software and performing setup tasks. Instructions for starting and exiting Gentran:Server and for setting preferences and default values. Also includes instructions for checking files in and out and saving files.
Mapping and Translation Guide	Instructions for performing mapping and translation tasks using the Gentran:Server Visual Mapper. **Note** This guide is provided only if you maintain maps created with Gentran:Server version 5.3 or prior.
NCPCP User's Guide	Instructions for mapping and translating NCPDP files with the Application Integration system.

(Contd) Document	Description
XML User's Guide	Instructions for mapping and translating XML files with the Application Integration system. **Note** This guide is provided only if your organization has the Gentran:Server XML translation option.
ODBC User's Guide	Instructions for mapping and translating ODBC files with the Application Integration system. **Note** This guide is provided only if your organization has the Gentran:Server ODBC translation option.
GENCOD User's Guide	Instructions for mapping and translating GENCOD files with the Application Integration system and the Visual Mapper.
VDA User's Guide	Instructions for mapping and translating VDA files with the Application Integration system and the Visual Mapper.
Technical Reference Guide	Describes processes, lists command-line commands in alphabetical order, and describes file record layouts and data type formats.
Data Flow Administration Guide	User instructions for configuring data flows using the Gentran:Server for UNIX software. **Note** This guide is provided only if you have the Gentran:Server EC Workbench or higher product level.
Maintenance and Troubleshooting Guide	Instructions for maintaining your Gentran:Server installation. Also provides troubleshooting information to help determine the cause and solution of problems that may occur.
Advanced Data Distribution Guide	Instructions for configuring and using the Gentran:Server Advanced Data Distribution product. **Note** This guide is provided only if you have Gentran:Server with Advanced Data Distribution.
FTP Daemon User's Guide	Instructions for configuring and using the FTP Daemon tool with the Advanced Data Distribution product. (Continued on next page)

(Contd) Document	Description
Online Help	Context-sensitive help screens describing the Gentran:Server dialog boxes for the mapping and translation features. Also includes procedures for using the mapping and translation and the data flow administration software.
Readme file	Information about recent enhancements included with this software release. This file is in the /*readme* directory on the Windows client computer.

Other documentation

This table lists other documentation you may need to refer to when installing and setting up Gentran:Server.

Description	Source
Instructions for installing and using the operating system on your computer.	Your hardware vendor The computer manufacturer

Documentation Conventions

Typographic conventions

This table describes the typographic conventions used in this guide.

Convention	Use
Italics	This typeface is used for titles of other manuals and documents, names of files and file extensions, and to emphasize important information. **Example** *Gentran:Server for UNIX Getting Started Guide*
Bold	Bold type is used for program names, menu names, button names, and entries you are to make on-screen. **Example** A **password** is a set of characters a user must enter to gain access to a system.

Application Integration Basics

Overview

Introduction

In this chapter   This chapter introduces you to mapping and the Application Integration subsystem.

Key terms   This table lists the key terms used in this chapter.

Term	Description
compile	The process used to convert a map into a translation object.
DDF (Data Definition Format)	A file that defines a file format used in a map. It includes the hierarchical and looping structure of the data, the map objects and the attributes of the objects.
links	The visual lines that connect each field on the input side of the map to a field on the output side of the map.
loop	A mapping structure in which the records or groups repeat in sequence until either the group data ends or the maximum number of times that the loop is allowed to repeat is exhausted.
map	A file object that defines the corresponding relationship between document components in two different formats.
map version	The incremented number that indicates the rendition of the map. A lower number represents an earlier version; a higher number represents a later version.
translation object	A file containing information that instructs the translator how to convert a file from one format to another. When you compile a map you created, the result is a translation object.
translator	The Gentran:Server subsystem that translates data from one format into another.

Getting Started

Before you begin

Before you use the Application Integration subsystem, you should be familiar with the information in the *Gentran:Server for UNIX Getting Started Guide* or *Gentran:Server Workstation Getting Started Guide*.

Process table

This table lists the stages in the process used to get started with mapping and the Application Integration subsystem.

Stage	Description
1	Understand what a map is and the components it contains. **Reference** See Introducing Mapping for information.
2	Learn how to open and navigate the Application Integration subsystem. **Reference** See Introducing the Application Integration Subsystem for information.
3	Continue with the Designing your Map chapter.

Introducing Mapping

Maps

Definition

A map defines a set of relationships between document components in two different data formats. You define maps in the Gentran:Server Application Integration subsystem.

Input and output files

The input format and the output format can define the components of an:

◗ EDI standard file

◗ Application file with fixed-length fields

◗ Application file with variable-length fields

◗ ODBC file (if you have the ODBC translation option)

◗ XML file (if you have XML translation option)

Basic map types

These are the basic types of maps:

Map type	Defines the relationship between components of a document in…
standard-to-application	Standard format and an application file format.
standard-to-standard	Standard format and a different standard.
application-to-standard	Application file format and a standard format.
application-to-application	Application file format and another application file format.

ODBC map types

These are map types available with the ODBC translation option:

Map type	Defines the relationship between components of a document in…
standard-to-ODBC	Standard format and an ODBC format
application-to-ODBC	Application file format and ODBC format.

(Contd) Map type	Defines the relationship between components of a document in…
ODBC-to-standard	ODBC file format to a Standard format.
ODBC-to-application	ODBC format to an application format.
ODBC-to-ODBC	ODBC format to another ODBC format

Reference
See the *Gentran:Server for UNIX and Workstation ODBC User's Guide* for information about ODBC maps.

XML map types   These are map types available with the XML translation option:

Map type	Defines the relationship between components of a document in…
standard-to-XML	Standard format and an XML format
application-to-XML	Application file format and XML format.
XML-to-standard	XML file format to a Standard format.
XML-to-application	XML format to an application format.
XML-to-XML	XML format to another XML format

Reference
See the *Gentran:Server for UNIX and Workstation XML User's Guide* for information about XML maps.

Format options   A map defines an input format and an output format. Gentran:Server has four format options for the input or output side of a map:

◗ Delimited EDI

◗ Positional

◗ XML (available if you have the XML translation option)

◗ ODBC (available if you have the ODBC translation option)

This table shows the Gentran:Server format option for each type of input or output file.

Input/Output File	Gentran:Server format
EDI standard	Delimited EDI
Variable-length application file	Delimited EDI (Continued on next page)

Input/Output File	Gentran:Server format
Fixed-length application file	Positional
XML (available if you have the XML translation option)	XML
Database file (available if you have the ODBC translation option)	ODBC

References

See the *Gentran:Server for UNIX and Workstation XML User's Guide* for information on the format option for XML input and output files.

See the *Gentran:Server for UNIX and Workstation ODBC User's Guide* for information about ODBC input and output files.

Example map   This is an example of a map.
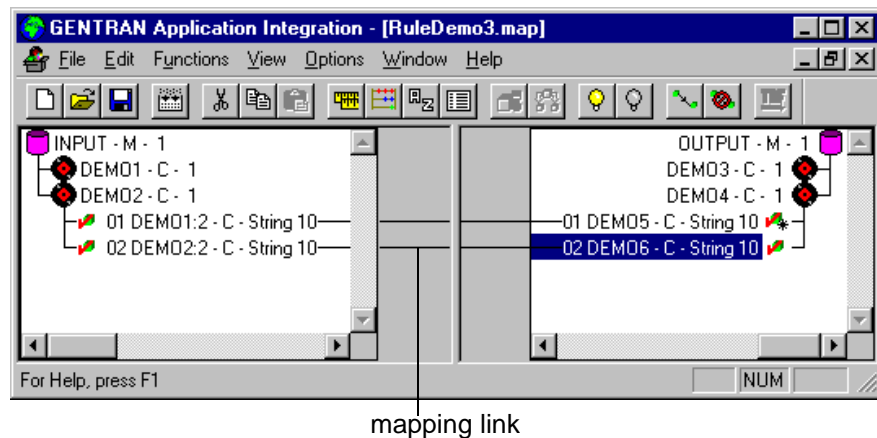
mapping link

Mapping links   Mapping links are the visual lines that connect each field on the input side of the map to a field on the output side of the map. The default setting is to hide the links. You can choose another default display setting on the Links tab of the Application Integration Preferences dialog box.

Mapping operations

Mapping operations are sets of directions you attach to a mapping component on the input or output side of a map. Mapping operations include:

◗ simple or complex calculations

◗ constant values to load in a document

◗ qualifying relationships.

You attach mapping operations to a mapping component with either a Standard rule or an Extended rule.

Translation Objects

Definition

A translation object is a set of rules that instructs the translator how to convert a file from one format to another, such as from an application format to a standard format. You can use the same translation object on any platform that Gentran:Server supports.

Relationship to a map

A map is the source file of a translation object.

To transform a map into a translation object, you must compile the map. Compiling changes the information in the map into a set of instructions that the translator understands.

Where compiling takes place

The compiler is part of the mapper and resides on your client or PC workstation machine. You compile maps on the client machine if you have Gentran:Server for UNIX or on your workstation if you have Gentran:Server Workstation.

The Translation Process

Diagram   This illustration depicts the translation process for inbound standard-to-application and outbound Application-to-standard processing.

```
                  ┌─────────────────────────────────────────────────┐
                  │  Standard-to-Application Translation Object      │
                  └─────────────────────────────────────────────────┘
                                        │
                                        ▼
              Standard          ┌───────────────────┐       Application
Inbound  ───────────────────►  │                   │  ───────────────────►
                               │   Gentran:Server   │
Outbound ◄───────────────────  │                   │  ◄───────────────────
              Standard          └───────────────────┘       Application
                                        ▲
                                        │
                  ┌─────────────────────────────────────────────────┐
                  │  Application-to-standard Translation Object      │
                  └─────────────────────────────────────────────────┘
```
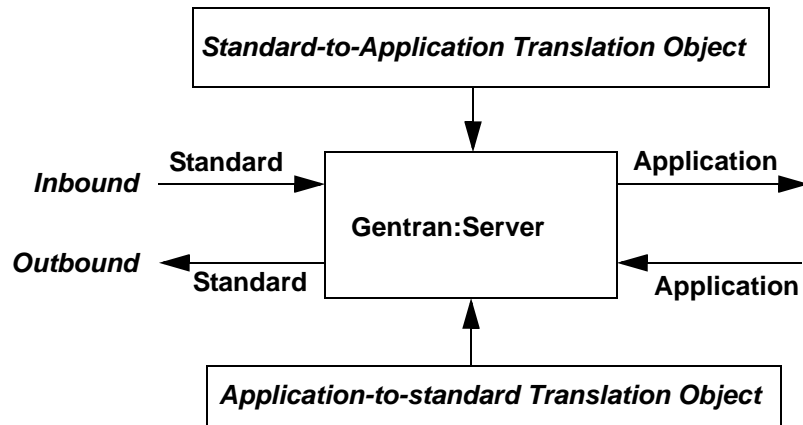
Data Definition Format (DDF)

Introduction

A Data Definition Format (DDF) is a file that defines a file format used in a map. You can use a DDF to define your application, proprietary standard, and XML formats to Gentran:Server.

Contents

A DDF includes the hierarchical and looping structure of the data, key field relationships, the map objects (groups, records, fields, etc.), and the attributes of the objects (names, descriptions, data types, etc.).

Format

DDF files are expressed in XML.

Saving a file definition as a DDF

If you have a map with a file format already defined, you can save the file format of the selected side of a map as a DDF file. This enables you to use it in another map.

When you save a side of a map as a DDF file, the system uses the Gentran:Server DDF Document Type Definition file, GENTRAN_DDF.DTD, to define the format structure of the DDF. The information in the GENTRAN_DDF.DTD file for storing a side of a map as a DDF includes which elements are required, how they are arranged, and which elements they can contain.

The GENTRAN_DDF DTD file is automatically installed on your system during the Gentran:Server installation process.

Reference
See How to Save a File Definition for instructions.

Using a DDF in a map

To use a DDF file in a map, you can either:

◗   Load an existing DDF and modify it to define a new data format for the input or output file format in a new map.

◗   Use the Open File Definition function to replace the input or output side of a map with a previously-defined file definition.

> **Reference**
> See How to Load a File Definition for more information on loading a DDF file definition.

DDF restrictions   When you use a DDF in Gentran:Server, these restrictions apply:

◗   Generally, the DDF does not contain standard or extended rules, links, or any other map information that does not specifically relate to the file format. The one exception to this rule is the use of code lists and constants. If the map references a "Use codelist" standard rule on a field and instructs the system to raise a compliance error if the code is not found in the code list, Gentran:Server saves the standard rule and the associated code list. When you save the input or output side of a map as a DDF, the system saves code lists and constants defined in the map to the DDF.

Reference
See Using the Use Code Standard Rule for more information on code lists and using the "Use codelist" standard rule.

◗   Gentran:Server assigns acceptable default attribute values to objects when the DDF does not include the attributes.

Example
If you do not specify a value for Minimum Length, the system assigns zero.

Positional Application File Definition Map Components

Introduction   If the input or output of your map is an application file, you must define your application to the Gentran:Server Application Integration subsystem. Your application file definition must contain all the information that you need to extract from your partner's document (if the map is for inbound processing) or to send to your partner (if the map is for outbound processing), so that your system can accurately process the data.

Your application file has either:

◗   Fixed-length fields (positional format) or

◗   Variable-length fields (delimited EDI format).

This topic describes the map components for positional application file definitions.

Reference
If your application file has variable-length fields, you must define your application in delimited EDI format. See Standard and Delimited File Definition Map Components for a description of the components of an application file in delimited EDI format.

Map components   Positional file definitions contain three basic map components. Each is represented by an icon in the Application Integration subsystem.

◗   groups

◗   records

◗   fields

Map icons   This table lists the icons that visually represent the map components of a positional application file definition.

Icon	Description
	A positional **file definition** is the primary node in the file structure. All other components are subordinate to it.
	A **group** is a looping structure that contains related records and groups. "Looping" means that the records or groups repeat in sequence until either the group data ends or the maximum number of times that the loop is allowed to repeat is exhausted. **Nested looping structure** A group that is subordinate to another group is a **subgroup.** This corresponds to a nested looping structure – a loop within a loop.
	A **record** contains a collection of related fields. A record can occur once or can repeat multiple times.
           	A **field** is the smallest piece of information defined in the application file definition. A field is the application map component that you map (link) to a corresponding field to show the input and the corresponding output when the data is translated. **Note** When a field has a link performed against it, the system displays a red check mark over the field icon. When a field has a mapping operation performed against it, the system displays a black asterisk over the field icon. When a field has a mapping operation and a link performed against it, the system displays both a red check mark and a black asterisk over the field icon.

Standard and Delimited File Definition Map Components

Introduction

The file format for standard file definitions and application file definitions with variable-length fields is **delimited EDI**. The file definition must contain all the information that you expect to receive from your trading 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 specific standard map components that you use depend on the type of map you are creating.

Standard file definitions

If the input or output of your map is a standard file, there are three possible ways to create the file definition:

◗ Define the file to Gentran:Server.

 If you want to use a specialized (proprietary) version of a standard that is not available in the Gentran:Server standards database, you must define the standard file definition yourself.

◗ Have the system generate the file definition for you and accept all its components.

 If you select a standard for the input or output side of a map, the Application Integration subsystem generates a standard file definition for you. The system bases the file definition on the standard (agency), version, and transaction set, and, for a TRADACOMS standard only, the release you select when you create your map. The system includes all the groups, segments, composites, and elements that the standards agency defined for the version of the document you selected.

◗ Have the system generate a file definition for you and then customize it.

 If you are creating an application-to-standard or standard-to-application map, you probably need to customize the system-generated standard file definition. You can modify the properties of the map components and use specialized Gentran:Server functions to manipulate the standard file definition structure.

References
See How to Create a New Map for information on generating a standard file definition when you create a new map.

See Defining a Standard File Format or Variable-Length Application File for information about customizing the properties of the map components in a system-generated file definition.

Application file definitions

If the side is an application file with variable-length fields, you must define the file to Gentran:Server, using a delimited EDI format.

Reference
See Defining a Standard File Format or Variable-Length Application File for information about using Gentran:Server functions to create a delimited EDI file definition.

Map components

This table lists the map components that Gentran:Server uses to visually represent the delimited EDI file format for standard file definitions and application file definitions with variable-length fields.

Icon	Description
	The **file** icon is the primary component. All other components are subordinate to the file icon.
	A **group** is a looping structure that contains related segments and 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. The EDI standards define groups. **Nested looping structure** A group that is subordinate to another group is a **subgroup.** This corresponds to a nested looping structure – a loop within a loop.
	A **segment** contains a group of related elements or composite data elements that combine to communicate useful data. The EDI standards define segments. A segment can occur once or can repeat multiple times. *(Continued on next page)*

(Contd) Icon	Description
	A **composite** is a data element that contains two or more component data elements or subelements. The EDI standards that use them (EDIFACT, TRADACOMS, and certain ANSI X12 standards) define the composites.
	An **element** is the smallest piece of information defined in a standard. An individual element can have different meanings, depending on the context. For this reason, elements usually do not have useful meaning until they are combined into segments.

An element is the delimited EDI map component that you map (link) to a corresponding element on the opposite side of the map to show the input and the corresponding output when the data is translated.

Note
When an element has a link performed against it, the system displays a red check mark over the element icon. |
| | When an element has a mapping operation performed against it, the system displays a black asterisk over the element icon. |
| | When an element has a mapping operation and a link performed against it, the system displays both a red check mark and a black asterisk over the element icon. |

Introducing the Application Integration Subsystem

Overview

Basic translation types

The Gentran:Server Application Integration subsystem enables you to create maps and generate translation objects for these types of translation:

◗ Standard-to-application

◗ Standard-to-standard

◗ Application-to-standard

◗ Application-to-application

ODBC translation option

If you have the ODBC translation option, then the Application Integration subsystem also enables you to create maps and generate translation objects for these types of translation:

◗ Standard-to-ODBC

◗ Application-to-ODBC

◗ ODBC-to-standard

◗ ODBC-to-application

◗ ODBC-to-ODBC

Reference
See the *Gentran:Server for UNIX and Workstation ODBC User's Guide* for information to create maps and generate translation objects for ODBC translation.

XML translation option

If you have the XML translation option, then the Application Integration subsystem also enables you to create maps and generate translation objects for these types of translation:

◗ Standard-to-XML

◗ Application-to-XML

◗ XML-to-application

- ◗ XML-to-standard
- ◗ XML-to-XML
- ◗ XML-to-ODBC (if you also have the ODBC translation option)
- ◗ ODBC-to-XML (if you also have the ODBC translation option)

Reference
See the *Gentran:Server for UNIX and Workstation XML User's Guide* for information to create maps and generate translation objects using XML.

Linking translation objects to Trading Partner relationships

Once you have generated a translation object, you must establish correct trading relationships for all trading partners from which you receive or to which you send data. During the process of establishing a Trading Partnership, you must specify the name of the appropriate translation object that the translator uses to translate the data.

Reference
See Working with Trading Partnerships for instructions on creating a Trading Partnership.

The Application Integration Window

Introduction   The Application Integration subsystem provides the tools you need to create maps and compile them to produce translation objects. You start tasks from the Application Integration Window.

Application Integration Window   This is the Application Integration Window with an open map displayed.

Main menu bar —

Toolbar —

Status bar —
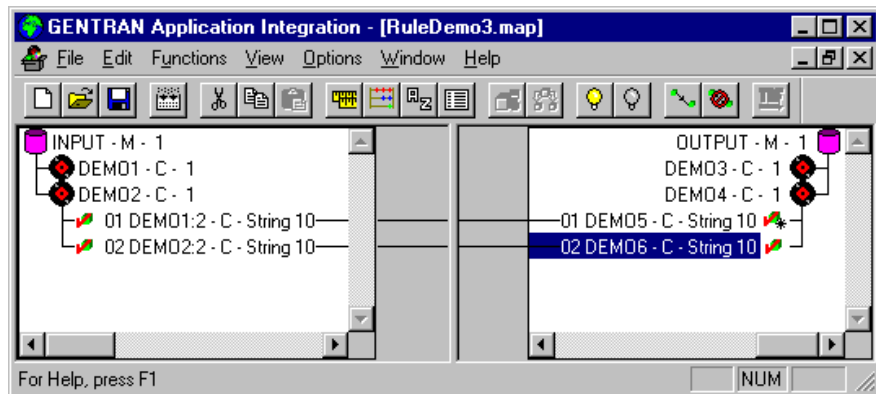
Note
In the Gentran:Server for UNIX product, the window title bar includes the environment and host name.

Parts and functions

This table lists the parts of the Gentran:Server - Application Integration Window and their functions.

Part	Function
Main menu bar	Contains drop-down menus. Unavailable items appear dimmed.
Main Toolbar	Enables you to access some of the most common functions in the Application Integration subsystem. Unavailable items appear dimmed. **Note** 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.

How to Open, Exit, and Navigate Application Integration

Introduction

This topic explains how to:

◗   Open the Application Integration subsystem

◗   Exit the subsystem

◗   Use the menus and toolbar to move about the subsystem.

Opening Application Integration

You can open the Application Integration subsystem in these ways:

◗   Select **Application Integration** from the Gentran:Server Edit menu.

◗   Click the Application Integration button on the Gentran:Server main toolbar.

Warning
Do not attempt to open the Application Integration subsystem in any other way. The subsystem functions correctly only when opened in the ways listed.

Exiting Application Integration

To exit the Application Integration subsystem and return to Gentran:Server, select **Exit** from the File menu.

Moving about in Application Integration

Use the menus and toolbar to navigate within the Application Integration subsystem.

For a description of the options available on each menu, see the online Help system.

Note
When you start Gentran:Server Application Integration, the main menu bar contains a subset of menu items. The system displays a full set of menu items after you create a new map or open an existing map.

Using the Toolbar

This is the Application Integration Toolbar.

Toolbar

This table describes the items in the Application Integration Toolbar.

Item	Name	Function
	New Map	Starts the New Map Wizard.
	Open Map	Enables you to open an existing map.
	Save Map	Saves the map.
	Compile Map	Compiles the map and creates a translation object.
	Cut	Cuts the selected component and places it on the clipboard.
	Copy	Makes a copy of the selected component and places it on the clipboard.
	Paste	Pastes a cut or copied component.
	Edit Session Rules	Displays the Edit Session Rules dialog box.
	Edit Constants	Displays the Edit Constant dialog box.
	Edit Syntax Tokens	Displays the Syntax Tokens dialog box.
	Edit Code Lists	Displays the Code Lists dialog box.
	Promote	Extracts one iteration (instance) of a group or repeating record from the selected loop.

(Contd) Item	Name	Function
	Split	Breaks a group or repeating record into two loops.
	Toggle Activate	Turns the Activate function on and off.
	Toggle Deactivate Mode	Turns the Deactivate Mode on and off.
	Toggle Link Mode	Turns the linking function on and off.
	Unlink	Deletes the link between components.

How to Open and Close a Map

Introduction   This topic explains how to use the Application Integration subsystem to:

◗ Open a map

◗ Close a map

Opening a map   Use this procedure to open a map in the Application Integration subsystem.

Step	Action
1	Select **Open** from the Application Integration **File** menu. **System response** The system displays the File Open dialog box.
2	Select the file name from the File Open dialog box. **Notes** In the Gentran:Server for UNIX product, the Application Integration subsystem can open only those maps that have been created on or checked out to the client.
3	Click **Open**. **Note** If you attempt to open a map generated by another version of the mapper, the Application Integration subsystem displays a warning message indicating that the map cannot be opened in the current mapping editor.

Closing a map   To close a map in the Application Integration subsystem, select **Close** from the **File** menu.

Setting Preferences and Default Values

Application Integration Preferences Dialog Box

Introduction

The Application Integration Preferences dialog box is used to set various default values and options. This dialog box has seven tabs.

Application Integration display preferences

You can customize how the Gentran:Server Application Integration subsystem displays your maps. You can choose:

◗   Which component descriptions your system displays (group, record, field)

◗   Foreground and background colors of map components

◗   Font type, style, and point size

◗   Whether links are displayed

◗   The map version number to auto-increment

◗   The default date format

Tab descriptions

The Application Integration Preferences dialog box has seven tabs that you can use to set various default values and options. This table describes the tabs and their functions.

Tab	Function
Tree	Customizes the look of your map format tree.
Links	Sets the way in which links are shown between input and output components.
Standard Formats	Sets the default date formats that the system uses for elements read from the standards database.
Positional Defaults	Specifies the default data type format, pad character, and alignment for fields.
Files	Sets automatic backup for map files and specifies the default directories that the browser looks in for maps and translation objects.

Tab	Function
Confirmation	Selects the situations for which Application Integration will prompt you to confirm an action.
Version	Specifies when the system increments a map's version number.

Tree tab   This illustration shows the Tree tab.
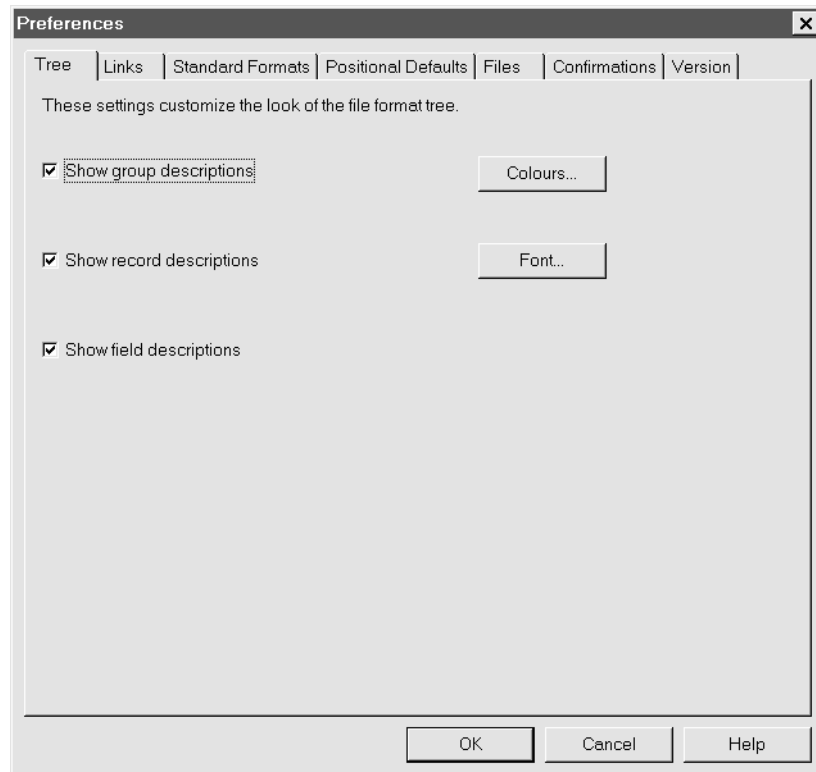
Tree tab fields and functions   This table describes the fields of the Tree tab and their functions.

Field	Function
Show group descriptions	Displays descriptions of group components in a map tree.
Show record descriptions	Displays descriptions of record components in a map tree.

(Contd) Field	Function
Show field descriptions	Displays descriptions of field components in a map tree.
Colours	Displays the Colours dialog box, which enables you to set the colours your system uses to display components.
Font	Displays the Font dialog box, which enables you to set the font your system uses to display map text.

Colours dialog box

This illustration shows the Colours dialog box.
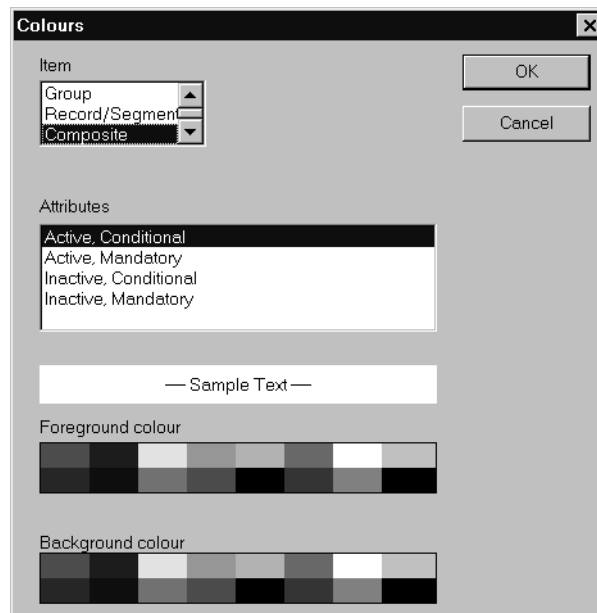
Colours dialog box fields and functions

This table describes the fields of the Colours dialog box and their functions.

Field	Function
Item	Identifies the item to which you want to apply colours.
Attributes	Further identifies the item to which you want to apply colours.
Sample Text	Displays an example of the colours selected for the item.

(Contd) Field	Function
Foreground colour	Selects the colour of the text of the item.
Background colour	Selects the background colour of the item.
OK	Saves your changes and closes the dialog box.
Cancel	Cancels any changes you made and closes the dialog box.

Fonts dialog box   This illustration shows the Fonts dialog box.
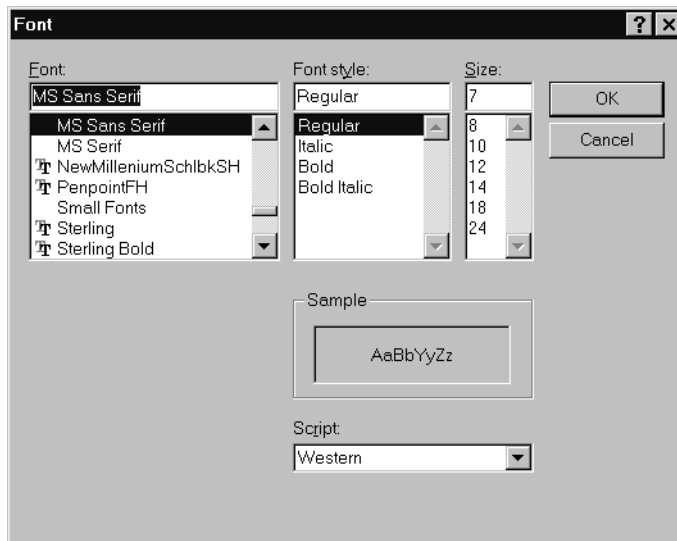
Fonts dialog box fields and functions   This table describes the fields of the Fonts dialog box and their functions.

Field	Function
Font	Selects the font from the list of available fonts on your system.
Font Style	Selects the style of the font.
Size	Selects the size of the font.
Sample	Displays a sample of the text.
Script	Selects the type of script. Currently, only Western is available.

Field	Function
OK	Saves your changes and closes the dialog box.
Cancel	Cancels any changes you made and closes the dialog box.

Links tab   This illustration shows the Links tab.

Links tab fields and functions   This table describes the fields of the Links tab and their functions.

Field	Function
Show no links	Turns the visual display of links off.
Show links to or from the currently selected element	Shows the links only for the selected element.
Shows links to or from all visible elements	Shows links only for the visible elements.

(Contd) Field	Function
OK	Saves your changes and closes the dialog box.
Cancel	Cancels any changes you made and closes the dialog box.

Standard Formats tab

This illustration shows the Standard Formats tab.
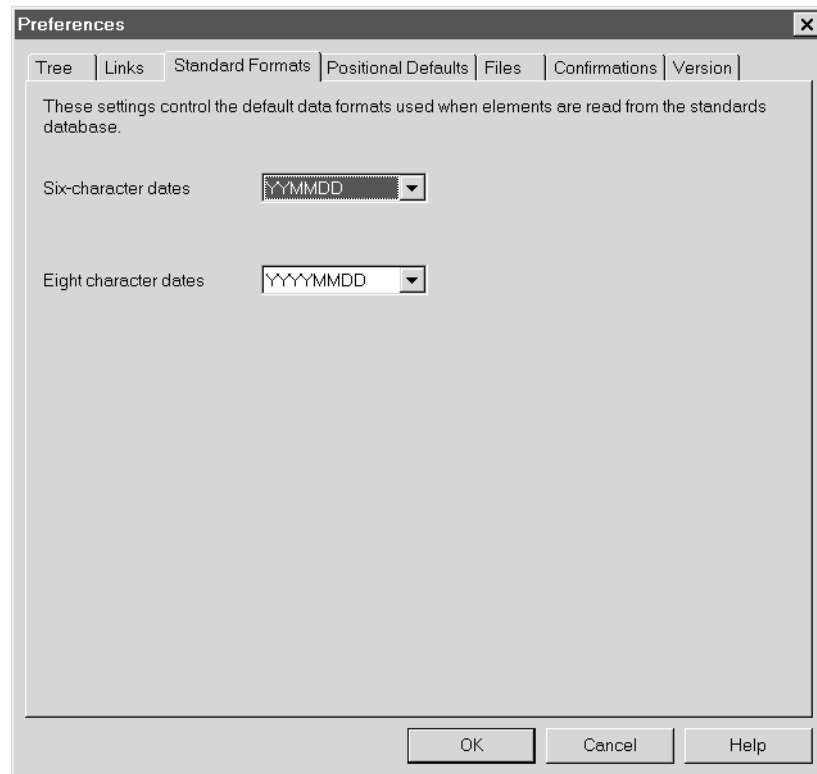
Standard Formats tab fields and functions

This table describes the fields of the Standard Formats tab and their functions.

Field	Function
Six-character dates	Selects the default format for six-character dates.
Eight-character dates	Selects the default format for eight-character dates.

Positional Defaults tab

This illustration shows the Positional Defaults tab.
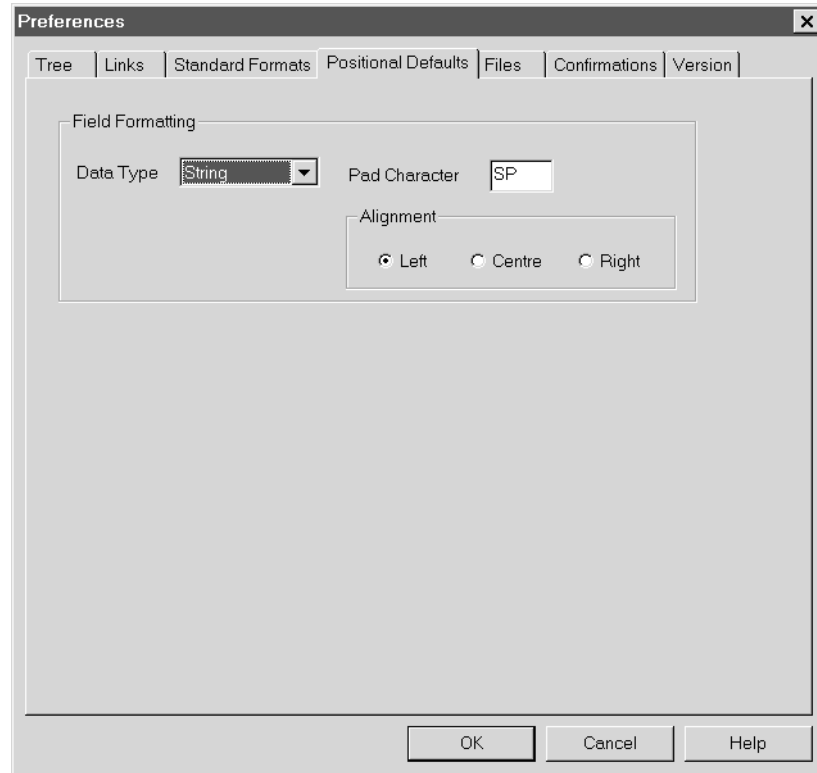
Positional Defaults tab fields and functions

This table describes the fields of the Positional Defaults tab and their functions.

Field	Function
Data Type	Selects the default data type for positional data.
Pad Character	Identifies the default pad character.
Alignment	Selects the default alignment for positional data.

Files tab   This illustration shows the Files tab.
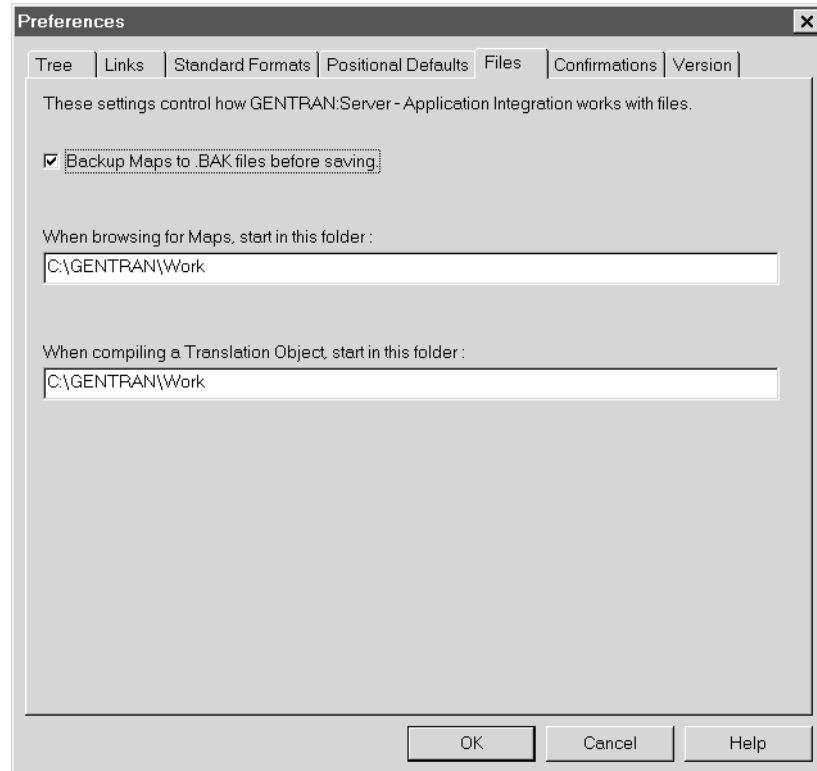
Files tab fields and functions   This table describes the fields of the Files tab and their functions.

Field	Function
Backup Maps to .BAK files before saving	Makes a backup copy of the map that does not include the changes made since the last time the file was saved.
When browsing for maps, start in this folder	Specifies the default folder that the browser looks in for maps.
When compiling a translation object, start in this folder	Specifies the default folder that the system uses to store working copies of translation objects (prior to checking them in).

Confirmations tab

This illustration shows the Confirmations tab.
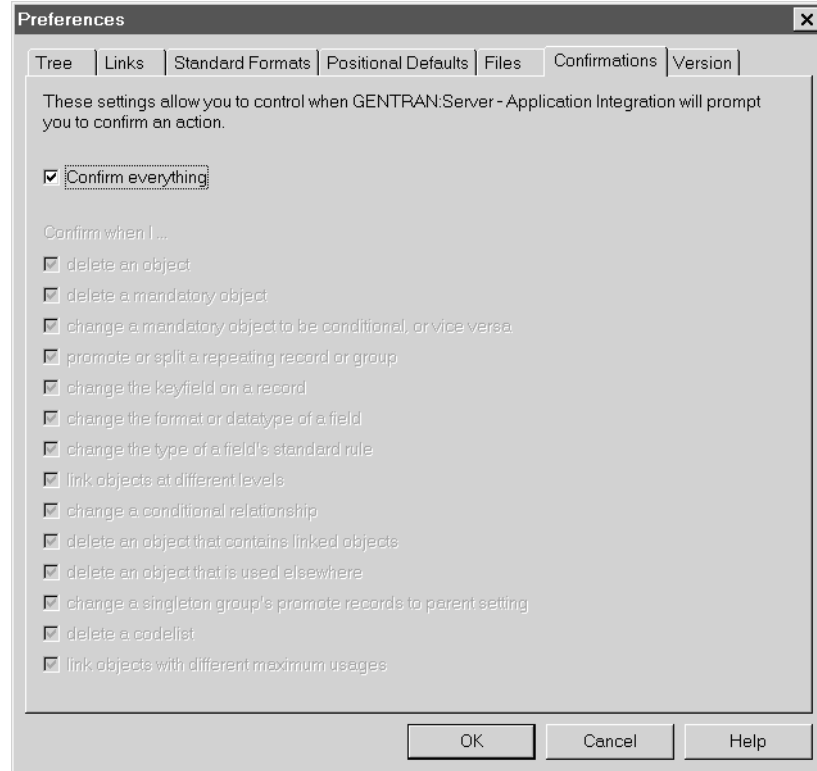
Confirmations tab fields and functions

This table describes the fields of the Confirmations tab and their functions.

Field	Function
Confirm everything	Selects all the confirmation options.
Confirm when	Lists the activities for which you can select a confirmation prompt.

Version tab   This illustration shows the Version tab.
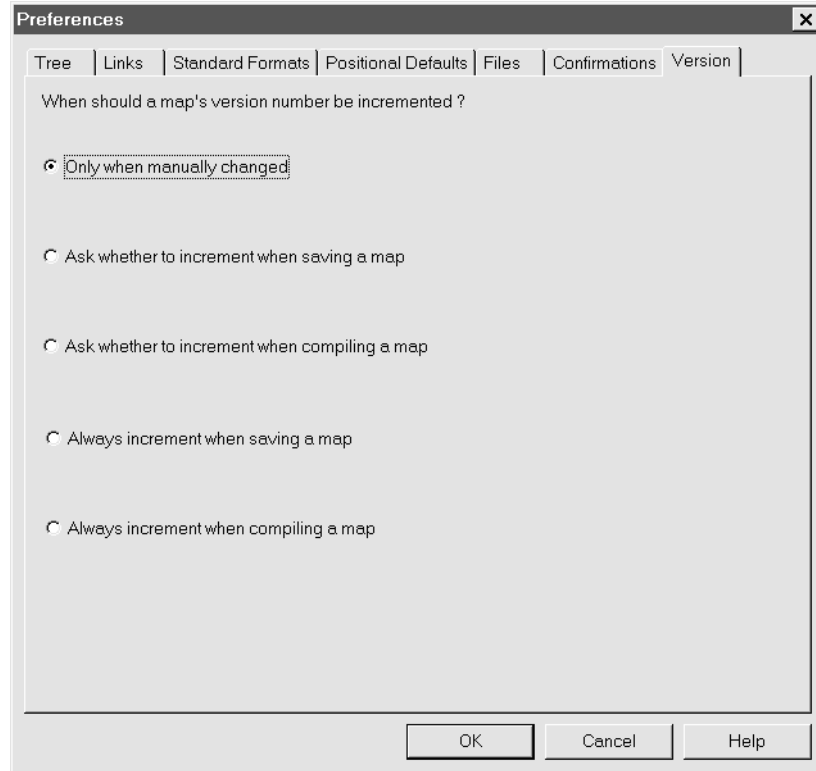
Version tab fields and functions   This table describes the fields of the Version tab and their functions.
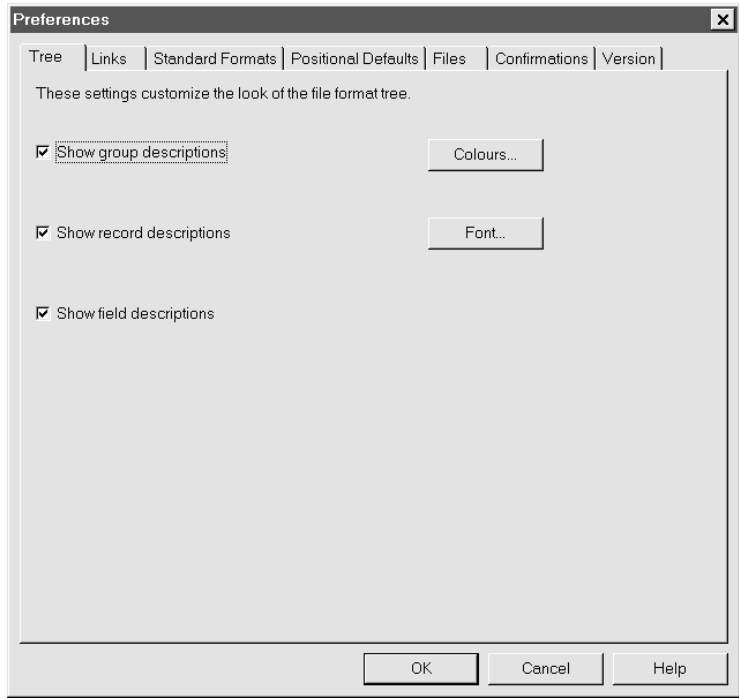
Field	Function
Only when manually changed	Increments the map version number only when you manually update the version.
Ask whether to increment when saving a map	Prompts you to increment the map version number when you save a map.
Ask whether to increment when compiling a map	Prompts you to increment the map version number when you compile a map.
Always increment when saving a map	Automatically increments the map version number when you save a map.
Always increment when compiling a map	Automatically increments the map version number when you compile a map.

How to Set the Application Integration Preferences

Introduction   The Application Integration subsystem enables you to set a number of default display, formatting, and prompting options.

Procedure   Use this procedure to set the Application Integration Preferences.

Step	Action
1	Open the Application Integration subsystem by clicking on the AI icon on the Gentran:Server toolbar.
2	Click **Options** on the menu bar and select **Preferences**. **System Response** Application Integration displays the Preferences dialog box.
3	Click the tab you want to set or change.
4	Select the settings.
5	Repeat Steps 3 and 4 for other tabs you want to change.
6	Click **OK** to save your changes.
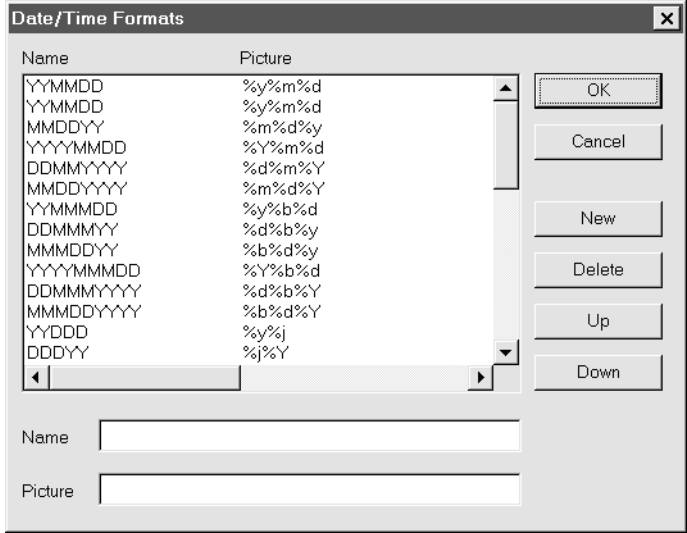
How to Set the Application Date/Time Formats

Introduction

This procedure explains how to add formats to and delete formats from the list of date and time formats that you can use in map fields.

Adding a new format

Use this procedure to set the Application Integration Preferences.
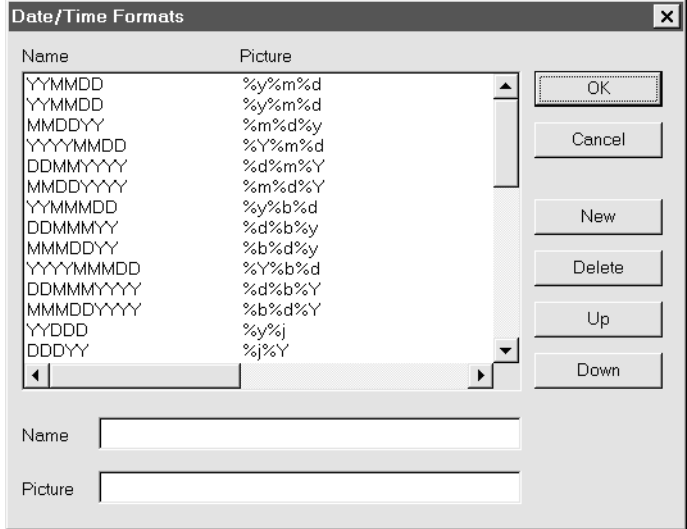
Step	Action
1	Open the Application Integration subsystem by clicking on the AI icon on the Gentran:Server toolbar.
2	Click **Options** on the menu bar and select **Date Formats**. **System Response** Application Integration displays the Date/Time Formats dialog box.
3	Click **New**. **System Response** Application Integration displays an entry marker at the bottom of the list.
4	Use the **Up** and **Down** buttons to move the bar to the place in the list where you want to insert the new format.
5	Type the format into the **Name** box and type the picture representation into the **Picture** box. (Continued on next page)

(Contd) Step	Action
6	Click **New**. **System Response** Application Integration adds the new format to the list.
7	Repeat Step 4 through 6 for each additional format.
8	Click **OK** to save your changes and exit the dialog box.

Deleting a format   Use this procedure to delete a format from the list of formats.

Step	Action
1	Open the Application Integration subsystem by clicking on the AI icon on the Gentran:Server toolbar.
2	Click **Options** on the menu bar and select **Date Formats**. **System Response** Application Integration displays the Date/Time Formats dialog box.
3	Select the format you want to delete and then click **Delete**. **System Response** Application Integration removes the format from the list.
4	Click **OK** to save your changes and exit the dialog box.

How to Set the Default Date Formats

Introduction
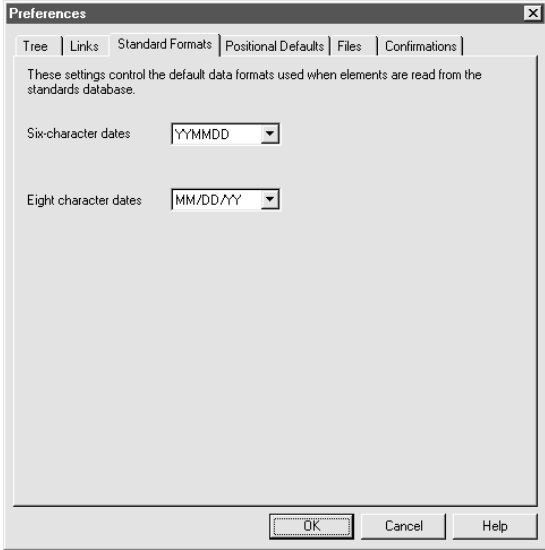
You should establish the default date format for all date fields *one time only*, preferably before you create your first map.

New maps vs. existing maps

This global feature changes the default date format for all new maps. The feature does not change the format of the date fields in existing maps.

Procedure

Use this procedure to establish the default date format.

Step	Action
1	From the Options menu, select **Preferences**. **System response** The system displays the Preferences dialog box.
2	Select the **Standard Formats** tab. **System response** The system displays the date format options. (Continued on next page)

(Contd) Step	Action
3	From the **Six-character dates** and **Eight character dates** lists, select the appropriate six-character and eight-character default date formats. **Tip** To change the order in which the date formats appear in the Six-character dates and Eight-character dates lists or to add a new date format to the lists, select Date Formats from the Options menu.
4	Click **OK** to accept the default date formats for the six- and eight-character date boxes and exit the Preferences dialog box.

Overriding the default date format
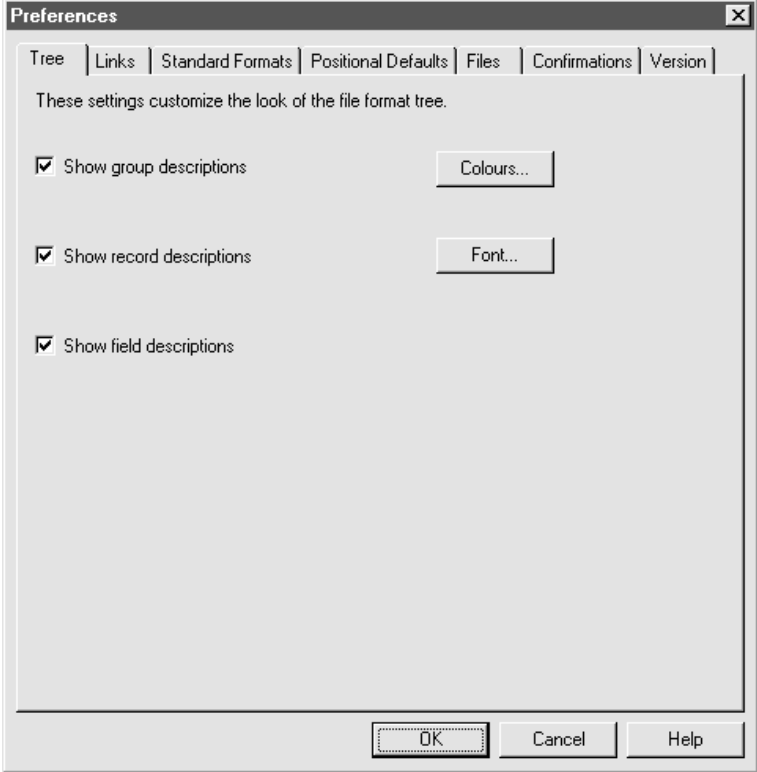
You can override the default date format on the Field Properties dialog box.

How to Customize Component Display Options

Introduction   Typically, you want to have all the group, record, and field descriptions displayed for reference. However, for a variety of reasons, you may not want the descriptions displayed. For instance, depending on the size of your monitor, it may be easier to see the entire map if the descriptions are not displayed.

Procedure   Use this procedure to choose which map components (groups, records, fields) Application Integration displays.

Step	Action
1	From the **Options** menu, select **Preferences**. **System Response** The system displays the Preferences dialog box with the Tree tab selected. (Continued on next page)

(Contd) Step	Action
2	Do you want to turn on the default display of group, record, and field descriptions? ◗ If YES, in the List section of the dialog box, click the check box in front of each item you want to display. To turn off the default display of group, record, and field descriptions, click the appropriate check box again to remove the check mark. ◗ If NO, continue with the next step. **Tip** You may want to experiment with shrinking the size of the font for the map. See How to Customize Global Fonts before you turn off the display of descriptions.
3	Do you want to change the default display colours? ◗ If YES, see How to Customize Component Colours for more information. ◗ If NO, continue with the next Step.
4	Do you want to change the default display font? ◗ If YES, see How to Customize Global Fonts for more information. ◗ If NO, continue with the next Step.
5	Click **OK** to save changes and exit the Preferences dialog box.

How to Customize Component Colours

Introduction   The Colours feature enables you to select foreground and background colours to visually define the various map components. The use of this feature is optional.

Procedure   Use this procedure to choose colours for all maps.

Step	Action
1	From the **Options** menu, select **Preferences**. **System response** The system displays the Tree tab of the Preferences dialog box.
2	Click **Colours**. **System response** The system displays the Colours dialog box.
3	From the Item list, select the type of map component (Group, Record/Segment, Composite, Field/Element) you want to change. (Continued on next page)

(Contd) Step	Action
4	From the **Attributes** list, further define the type of map component by selecting whether it is active or inactive, conditional or mandatory. The valid selections are: ◗ Active, Conditional ◗ Active, Mandatory ◗ Inactive, Conditional ◗ Inactive, Mandatory
5	Select the foreground and background colours for the component.
6	Click **OK** to use the component colours in all maps.

How to Customize Global Fonts
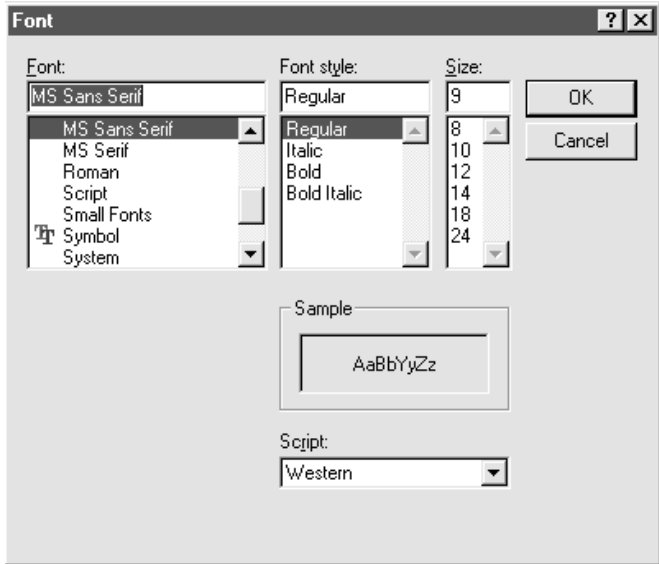
Introduction   The Font feature enables you to change the font that is used in the display of all maps. The default font that Gentran:Server uses is MS Sans Serif 9 point.

You can change the font type, style, and point size. This gives you the flexibility to shrink the font if you need to view more of the map on your monitor, enlarge the font, or change the type and style so that it is easier for you to read.

Procedure   Use this procedure to customize the display font for all maps.

Step	Action
1	From the **Options** menu, select **Preferences**. **System Response** The system displays the Tree tab of the Preferences dialog box.
2	Click **Font**. **System Response** The system displays the Font dialog box.
3	From the **Font** box, select the type of font. **Note** The default is MS Sans Serif. (Continued on next page)
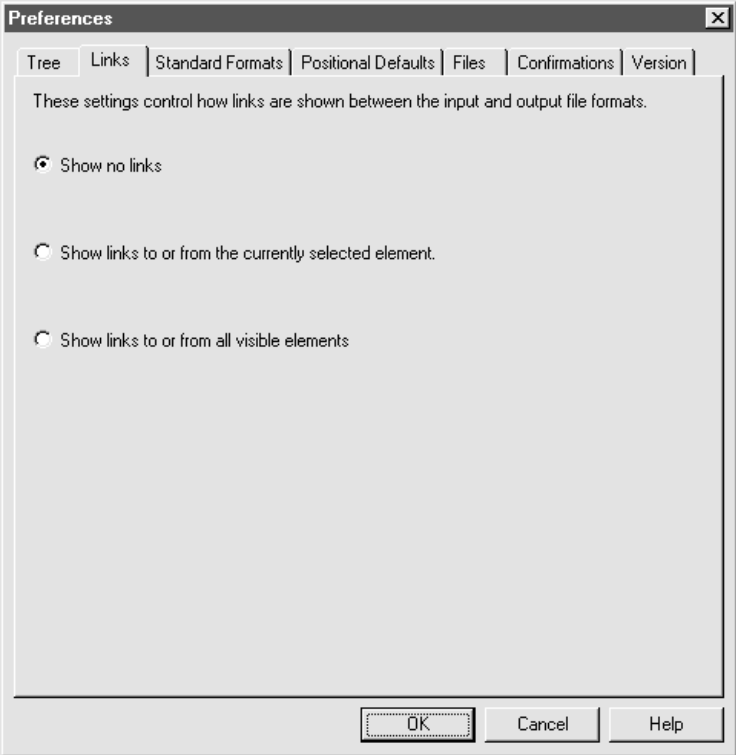
(Contd) Step	Action
4	From the **Font style** box, select the style. **Note** The default is Regular.
5	From the **Size** box, select the point size. **Note** The default is 9.
6	Click **OK** to use the font selections in all maps and exit the Font dialog box.

How to Customize the Display of Links

Introduction   Mapping Links are the visual lines that connect each field on the input side of the map to a field on the output side of the map. You can set the display to show no links, all links, or only the links for the selected field.

Procedure   Use this procedure to customize how all maps display mapping links.

Step	Action
1	From the **Options** menu, select **Preferences**. **System response** The system displays the Preferences dialog box.
2	Select the **Links** tab. **System Response** The system displays the linking options.

(Contd) Step	Action
3	Select the links display option that you want to set as the default for all maps. ◗ **Show no links** Do not display mapping links. ◗ **Show links to or from the currently selected element** Display the mapping links for only the currently selected field. (This option enables you to concentrate on the selected field.) ◗ **Show links to or from all visible elements** Display all the mapping links.
4	Click **OK** to save changes and exit the Preferences dialog box.

How to Set the Default Directories for Maps

Introduction   You can configure the Application Integration subsystem to look in a particular directory for maps when you choose to open a map. You can also set the default directory that the system uses to store compiled maps (translation objects).

Procedure   Use this procedure to set the default directories for maps.

Step	Action
1	From the **Options** menu, select **Preferences**. **System Response** The system displays the Preferences dialog box.
2	Select the **Files** tab. **System Response** The system displays the Files options.
3	Do you want the system to automatically make backup copies of your map files? ◗ If YES, click **Backup Maps to BAK files before saving**. ◗ If NO, continue with the next step.
4	Do you want to specify the default folder that the system looks in for maps? ◗ If YES, type the directory path in the **When browsing for maps, start in this folder** box. ◗ If NO, continue with the next step.
5	Do you want to specify the default folder that the system uses to store translation objects? ◗ If YES, type the directory path in the **When compiling translation objects, start in this folder** box. ◗ If NO, continue with the next step.
6	Click **OK** to save changes and exit the Preferences dialog box.

How to Customize the Auto-increment Map Version

Introduction   You can configure Application Integration to automatically increment the version number of a map when you save or compile a map.

Options   This list describes the available options for automatically incrementing the map version number.

- ◗ Only when manually changed
- ◗ Ask whether to increment when saving a map
- ◗ Ask whether to auto-increment when compiling a map
- ◗ Always auto-increment when saving a map
- ◗ Always auto-increment when compiling a map

Limitations   The auto-increment function updates the minor (point) version number of the map on the Translation Object Details dialog box, from .1 up to .255. If the point version number exceeds .255, the system updates the point version number to zero and increases the major version number.
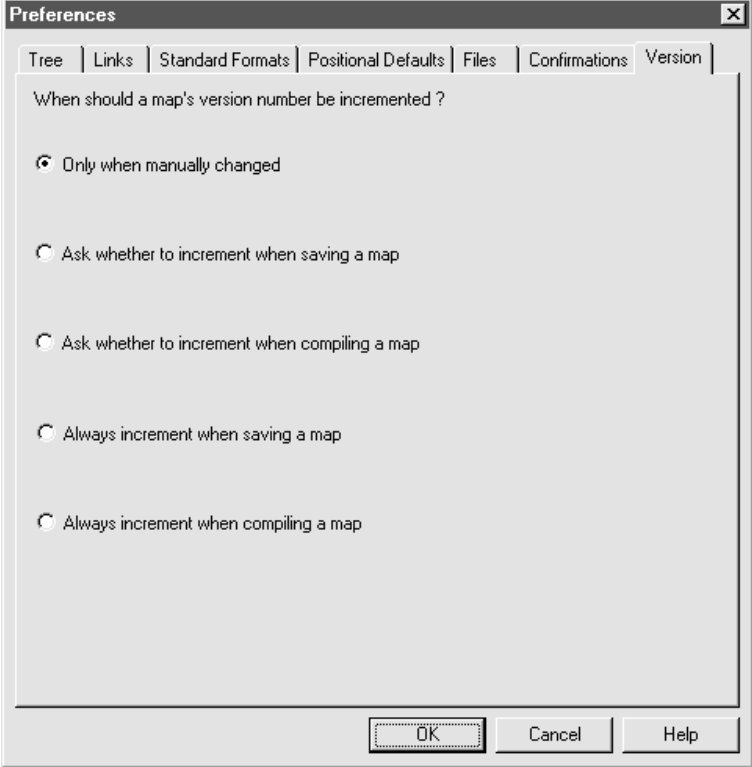
Example
The system increments version number 1.254 to 1.255 and increases 1.255 to 2.0.

Procedure   Use this procedure to set the auto-increment map version function.

Step	Action
1	From the Options menu, select **Preferences**. **System response** The system displays the Tree tab of the Preferences dialog box.

(Contd) Step	Action
2	Select the Version tab. **System Response** The system displays the auto-increment version options.
3	Select an option and click **OK**. **System response** The system sets the global auto-increment option you selected.

Designing your Map

Structuring Your Map

Overview

Introduction

In this chapter

This chapter explains the basic mapping functions:

◗ Analyzing mapping requirements

◗ Creating a map

◗ Defining file formats

◗ Formatting your data

◗ Completing a map

◗ Editing translation object details

Key terms

This table lists the key terms used in this chapter.

Term	Description
activate	The process of turning on groups, segments, composites, and elements that a standard does not define as mandatory, but that you have determined that you need to use in mapping.
Auto-trim	The Application Integration feature that enables you to automatically activate and deactivate map components on the EDI standard side of a map by using a sample EDI standard file as a model.
file definition	A file that defines how data needs to be formatted for an application to process it. These files have a .DDF extension. File definitions contain a layout of the records, fields, and groups in an application file.
loop	A sequence of repeating records or segments .
many-to-many mapping relationships	A mapping relationship that has a looping structure. (Continued on next page)

(Contd) Term	Description
map object	A component object of a map. For example: ◗ Positional, or Delimited EDI file ◗ group ◗ segment ◗ record ◗ element ◗ pcdata ◗ attribute ◗ attribute container ◗ content particle
nested looping structure	A mapping structure that has a loop within another loop.
one-to-one mapping relationship	A mapping relationship with the group, segment and field structure of the input side of a map matching that of the output side, and all fields are mapped directly across from the input to the output.
promote	To extract one iteration (instance) of a group or repeating record from a loop.
string-type field or element	A field or element that contains one or more printable characters. A syntax token defines the format of a string-type field or element.
syntax token	A symbol or expression that defines ranges of characters and numbers that are allowed to be used for a string-type field.
URL (Uniform Resource Locator)	An internet address that locates an individual resource file on the internet.

Building a Map Process

Before you begin   The first time you use the Gentran:Server Application Integration subsystem, you should set global default values to control the way the system displays your maps. At the least, you should specify the default date format you want the system to use.

Reference
See the topic Setting Preferences and Default Values for more information on setting global default values.

Process Table   This topic provides the stages in building a map with the Gentran:Server Application Integration subsystem.

Stage	Description
1	Analyze your mapping requirements. ◗ Obtain a layout of your input file and determine how it corresponds with the layout of the output file. ◗ Determine how you want to move data to or from each field. **Reference** See the Analyzing Mapping Requirements topic for information.
2	Create a new map. **Reference** See the How to Create a New Map topic for instructions on creating a map.
3	If a side of your map is a standard format, manually activate the appropriate groups, segment, and elements or use auto trim to automatically activate components based on a sample EDI standard file. **References** ◗ See the How to Manually Activate EDI Standard Map Components topic for more information on activation. ◗ See the How to Automatically Activate Standard Map Components topic for more information on auto trim. *(Continued on next page)*

(Contd) Stage	Description
4	If a side of your map is an application, load or create the application file format. **References** See the How to Create a New Map and How to Load a File Definition topics for instructions on loading an existing file definition. See the Defining a Fixed-Format Application File section for instructions on defining a fixed-format application. See the Defining a Standard File Format or Variable-Length Application File topic for instructions on defining an application file with variable-length fields.
5	If you have the XML translation option and if a side of your map is in XML format, load or build the XML format definition. **References** See the How to Create a New Map and How to Load a File Definition topics for instructions on loading an existing file definition. See the The XML File Object and Entities topic in the *Gentran:Server for UNIX and Workstation XML User's Guide* for information about XML map objects. See the Defining an XML File Format topic in the *Gentran:Server for UNIX and Workstation XML User's Guide* for instructions on creating map objects to define an XML file format. See the How to Create a New Map topic for instructions on customizing a standard format to define an XML format.
6	Complete the input and output structures of your map by creating and removing loops. **References** See the Structuring Your Map topic for instructions on using loops. *(Continued on next page)*

(Contd) Stage	Description
7	Map the appropriate data for each application field. **References** See the How to Create Simple Links topic for instructions on simple mapping (linking). See the Using Standard Rules chapter for information on using standard rules to apply mapping operations. See the Using Extended Rules chapter for information on using extended rules to apply mapping operations.
8	Compile the map to create a translation object. **Reference** See the How to Compile a Map topic for instructions.
9	Print the mapping report. Validate and review the map, and make modifications as needed. **Reference** See the How to Print a Mapping Report topic for more information on printing the mapping report.
10	Create the Trading Partner records. **References** See the Working with Trading Partnerships topic for instructions.
11	Are you are using Gentran:Server for UNIX (client/server)? ◗ If YES, copy the translation object to the user directory on the host and then continue with Step 12. ◗ If NO, continue with Step 12. **Reference** See the How to Copy a Translation Object to the Host topic for instructions on moving the translation object.
12	Translate sample data files to locate potential problems. **Reference** See the Running Translation chapter for instructions. (Continued on next page)

(Contd) Stage	Description
13	View the archive records for EDI data. **Note** You must configure your system to archive data to use this feature. **Reference** See the Archiving Translation Data chapter for instructions.
14	View inbound and outbound functional acknowledgments. **Note** You must configure your system to send and receive functional acknowledgments to use this feature. **Reference** See the Archiving Translation Data chapter for instructions.

Analyzing Mapping Requirements

Overview

Introduction   The first stage in creating a map is to analyze the mapping requirements. This is the most important step in creating a successful map. A complete analysis provides you with all the information you need to create the map efficiently and logically.

Analysis process   This table lists the stages of mapping analysis for both inbound and outbound processing.

Stage	Description
1	If the processing input or output file is an application file format, analyze the application file format. **Reference** See the <u>How to Analyze Your Application File Format</u> topic for instructions.
2	If the processing input or output file is a standard file format, analyze your trading partner's EDI standard file format. **Reference** See the <u>How to Analyze the Standard File Format</u> topic for instructions.
3	If your company purchased the XML option and if the processing input or output file is an XML file format, analyze the file format. **Reference** See the <u>How to Analyze the XML Format</u> topic in the *Gentran:Server for UNIX and Workstation XML User's Guide* for instructions.
4	Correlate the two formats. **Reference** See the <u>How to Correlate the Formats</u> topic for instructions.

How to Analyze Your Application File Format

Introduction   If the input or output file in your map is an application file, you begin by analyzing your application file format. The purpose of the analysis is to determine which records and fields your application file format must contain to extract information from the file you receive or format the file you send.

Procedure   Use this procedure to analyze your application file format.

Step	Action
1	Does your company have an existing application file format for the input or output file? ◗ If YES, obtain the record layout and continue with Step 2. ◗ If NO, continue with Step 2.
2	Create an application file format for the application file on paper: ◗ Determine which fields are necessary to process the data correctly. ◗ Group the fields logically under records.
3	Continue with one of the following: How to Analyze the Standard File Format How to Analyze the ODBC Format How to Correlate the Formats **Note** If you have the XML translation option and you want to analyze the XML format, see How to Analyze the XML Format in the *Gentran:Server for UNIX and Workstation XML User's Guide*.

How to Analyze the Standard File Format

Introduction   If the input or output file is an EDI standard file, you must analyze the standard file format.

Procedure   Use this procedure to analyze the EDI standard file format.

Step	Action
1	Determine which EDI standard, version, and transaction set to use. You and your trading partners need to agree on the standard, version, and transaction set you will use to exchange documents.
2	Find out what information your trading partner is sending you in each document. List the data contained in each element.
3	Determine which segments and elements your company requires.
4	Obtain an EDI standards manual and review each standard and version that you plan to use. **Reference** Contact the EDI standard agencies for standards manuals.
5	List the map components that you need to activate. **Note** You must activate all segments that you expect to receive from a trading partner, even if the segments are not mandatory.
6	Continue with one of the following: How to Analyze the ODBC Format How to Correlate the Formats. **Note** If you have the XML translation option and you want to analyze the XML format, see How to Analyze the XML Format in the *Gentran:Server for UNIX and Workstation XML User's Guide*.

How to Correlate the Formats

Introduction   After you have analyzed your input and output file formats, you must correlate the formats.

Procedure   Use this procedure to correlate the input and output formats.

Step	Action
1	Using the file formats from your analysis, match each input field with its corresponding output field.
2	Determine which mapping operations you need for each field in your application file.
3	Based on the required mapping operations, select a method for mapping each field and note the method in your analysis. **Linking** Linking (simple mapping) enables you to map a field from the input side of the map to a field on the output side of the map. The system visually represents a link between two map components (fields) with a line connecting the two fields. **Standard rules** Standard rules give you access to mapping operation functions that are more complex than simple linking, but less involved than extended rules. **Extended rules** Extended rules enable you to use a Gentran:Server proprietary programming language to perform virtually any mapping operation you require.
4	Continue with the Creating a Map topic.
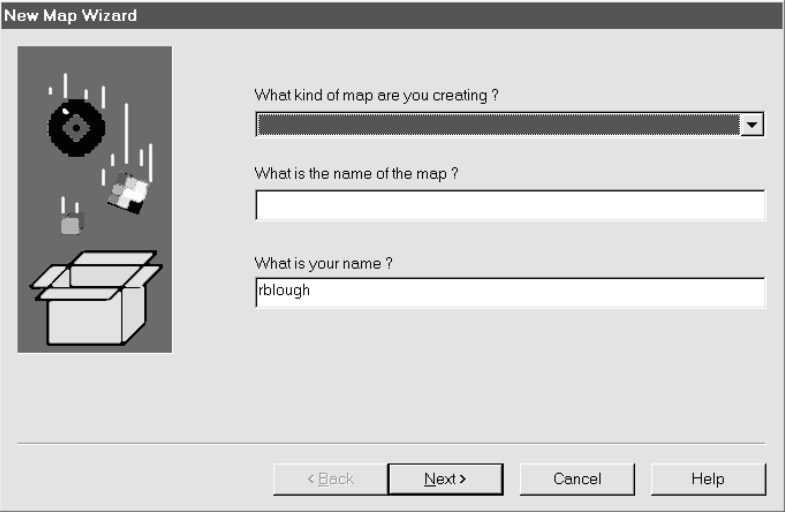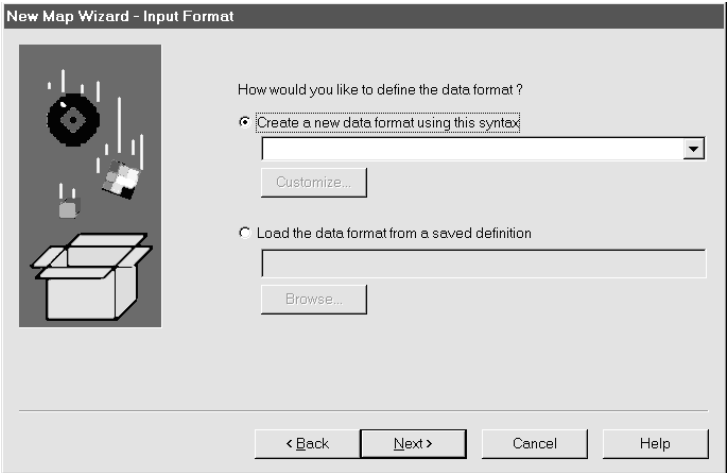
Creating a Map

How to Create a New Map

Introduction   This topic explains how to use the New Map Wizard to create a new map.

Starting the map   Use this procedure to start a new map.

Step	Action
1	From the Application Integration File menu, select **New**. **System Response** The system displays the New Map Wizard. (Continued on next page)

(Contd) Step	Action
2	Answer the following questions. ◗ *What kind of map are you creating?* Select the type of map (for example, application-to-standard, standard-to-application, or standard-to-standard, XML-to application, etc.). ◗ *What is the name of the map?* Type the unique name of the map. Omit the .MAP extension; the system will add the extension for you. ◗ *What is your name?* Type your name if it differs from the user name that the system supplied.
3	Click **Next.** **System Response** The system displays the New Map Wizard - Input Format dialog box.
4	Use this table to determine how to create the input side of the map.

IF you want to…	THEN go to…
create a new data format using a syntax that you define	the next step in this procedure.
load the data format from a saved file definition (.DDF)	Loading a saved file definition

<div align="center">(Continued on next page)</div>

(Contd) Step	Action		
5	Click **Create a new data format using this syntax**, and then select one of the following format options for the input side of your map:		
	IF your input file format is…	**THEN select…**	**AND then…**
	EDI standard	Delimited EDI	GO TO <u>Specifying an EDI standard</u>
	Variable-length application file	Delimited EDI	GO TO <u>Specifying a variable-length application file format</u>
	Fixed-length application file	Positional	Click **Next** and GO TO <u>Defining the output format</u>
	VDA or GENCOD	Positional	Click **Next** and GO TO <u>Defining the output format</u>
	Database	Database	Click **Next** and GO TO <u>Defining the output format</u>
	XML	XML	GO TO <u>Specifying an XML format</u>

Loading a saved file definition

Use this procedure to load a saved file definition.

Step	Action
1	From the Input Format or the Output format dialog box, select **Load the data format from a saved definition**.
2	Click **Browse** to display the Open File Definition dialog box.
3	From the Open File Definition dialog box, select the file definition (.DDF file) you want to load.
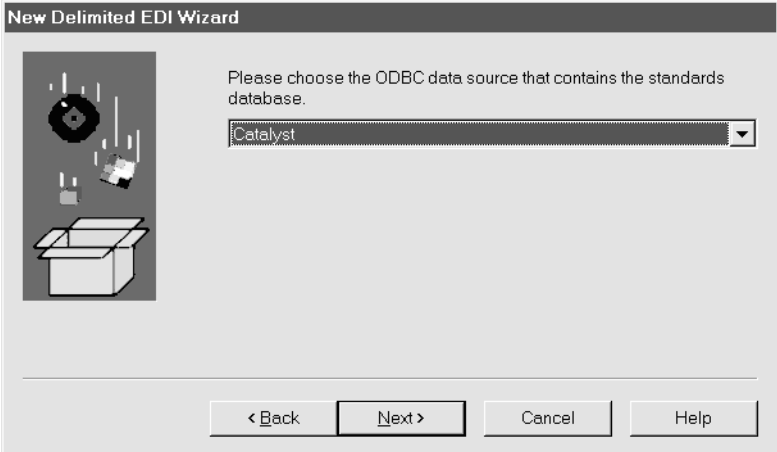
(Contd) Step	Action		
4	Click **Open.** **System Response**		
	IF the file definition format you loaded was…	**THEN you…**	**AND…**
	for the input side of the map, and is valid	click **Next** to display the Output Format dialog box	GO TO <u>Defining the output format</u>
	for the output side of the map, and is valid	click **Next**	click **Finish** to create the map. GO TO <u>What to do next</u>
	invalid	displays a message box that explains the problem and terminates the process.	Make sure that the .DDF file is in a format that the Application Integration subsystem understands. Repeat Steps 1 through 4, selecting a valid file definition.

Specifying an EDI standard

Use this procedure to specify a new EDI standard for the map.

Step	Action
1	Do you want to choose a pre-defined EDI standard for this side of the map? ◗ If YES, click **Customize** on the Input Format or Output Format dialog box; then continue with the next step. **System Response** The system displays the New Delimited EDI Wizard dialog box. ◗ If NO, GO TO Step 6.
2	On the New Delimited EDI Wizard dialog box, click **Next**.
3	Select the ODBC data source that points to the standards database.
4	Click **Next**. **Note** If the system displays a select or browse dialog box, select the directory folder that contains the standard.
5	Select the standards agency, version, and transaction set.
6	Click **Next**.
7	Click **Finish**. (Continued on next page)

(Contd) Step	Action		
8	Click **Next**. **System Response**		
	IF you specified the format for…	**THEN the system…**	**AND you should…**
	the input side of the map	displays the New Map Wizard - Output Format dialog box.	GO TO Defining the output format
	the output side of the map	displays the last dialog box in the wizard	Click **Finish**. GO TO What to do next

Specifying a variable-length application file format
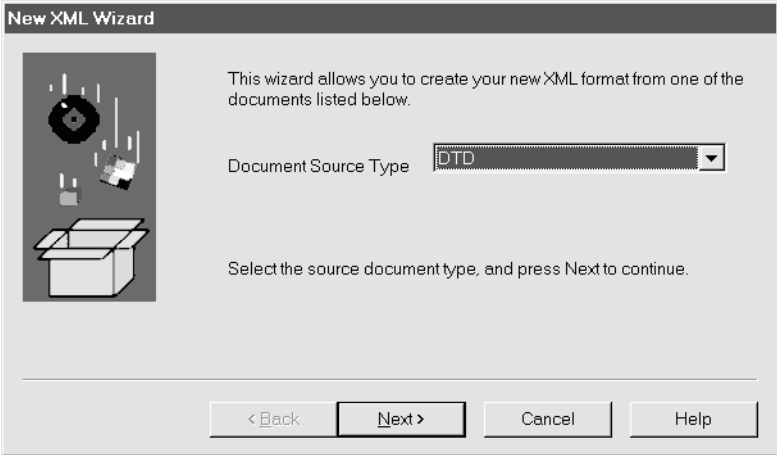
Use this procedure to specify a variable-length application file data format for the map.
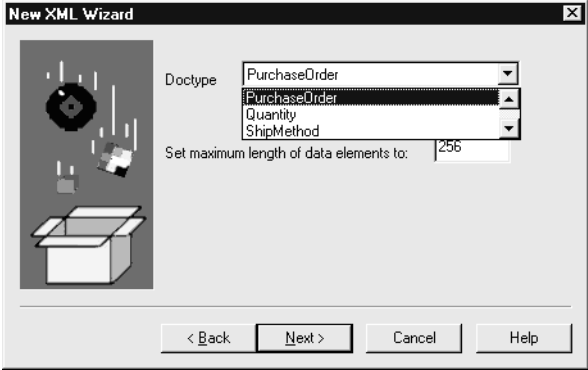
Step	Action		
1	Make sure that you selected **Delimited EDI** for the format. ⊙ Create a new data format using this syntax [Delimited EDI (ANSI X12, UN EDIFACT, Tradacoms, etc) ▼] [Customize...]		
2	Click **Next**.		
3	Click **Next**. **System Response**		
	IF you specified the format for…	**THEN the system…**	**AND you should…**
	the input side of the map	displays the New Map Wizard - Output Format dialog box	GO TO the Defining the output format topic.
	the output side of the map	displays the last dialog box in the wizard	Click **Finish.** GO TO the What to do next topic.

**Specifying an
XML format**

Use this procedure to specify an XML file format.
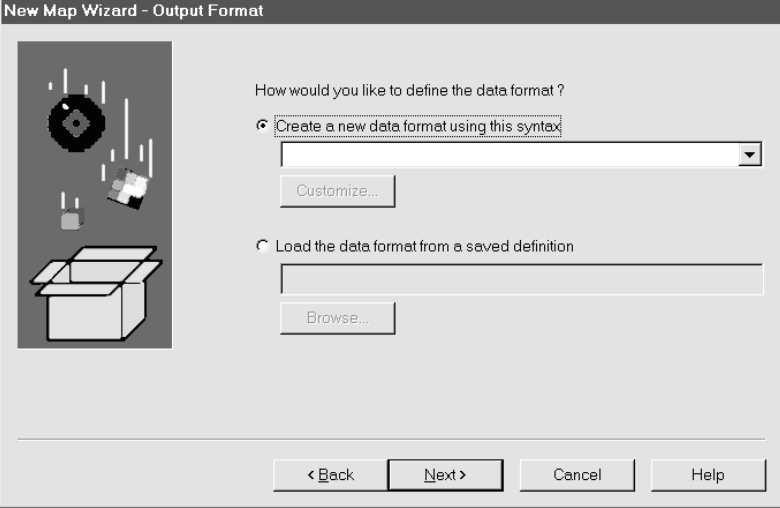
Step	Action
1	Do you want to create your XML format from a predefined document input type (a DTD)? ◗ If YES, click **Customize** on the Input Format or Output format dialog box and continue with the next step. **System Response** The system displays the New XML Wizard dialog box. ◗ If NO, GO TO Step 5. **System Response** The system displays the New Map Wizard - Output Format dialog box.
2	From the New XML Wizard dialog box, select the document input type and click **Next**.
3	Type the name of your DTD file or a URL pointing to the DTD and click **Next**. (Continued on next page)

(Contd) Step	Action
4	Select the **doctype**, set the maximum length of data elements, and then click **Next**. **System Response** The system displays the New XML Wizard (Doctype) dialog box.
5	Click **Finish**. **System Response** The system returns to the New Map Wizard - Input Format dialog box.
6	Click **Next**. **System Response**

IF you specified the format for…	THEN the system…	AND you should…
the input side of the map	displays the New Map Wizard - Output Format dialog box.	GO TO the <u>Defining the output format</u> topic.
the output side of the map	displays the final wizard dialog box	click **Finish** to create the map. GO TO the <u>What to do next</u> topic.

Defining the output format

Use this procedure to define the output side of the map.

Step	Action
1	On the New Map Wizard - Output Format dialog box, specify the format for the output side of the map.

IF you want to…	THEN go to…
load the data format from a saved file definition	the Loading a saved file definition topic.
create a new data format using a syntax that you define (Continued on next page)	the next step in this procedure.

(Contd) Step	Action		
2	Select one of the following format options for the output side of your map:		
	IF your output file format is…	**THEN select…**	**AND then…**
	EDI standard	Delimited EDI	GO TO the Specifying an EDI standard topic.
	Variable-length application file	Delimited EDI	GO TO the Specifying a variable-length application file format topic.
	Fixed-length application file	Positional	Click **Next.** Click Finish. GO TO the What to do next topic.
	VDA or GENCOD	Positional	Click **Next** and GO TO the What to do next topic.
	ODBC	ODBC	Click **Next.** Click Finish. GO TO the What to do next topic.
	XML	XML	GO TO the Specifying an XML format topic.

What to do next   After you create your map, you must define the input and output sides of the map. The steps you take are different, depending on the type of format you specified. Use this table to determine what to do next.

IF you want to…	THEN go to…
manually activate non-mandatory EDI standard groups, segments, and elements	How to Manually Activate EDI Standard Map Components
define a variable-length application file	Defining a Standard File Format or Variable-Length Application File
automatically activate EDI standard components based on a sample EDI file	How to Automatically Activate Standard Map Components
define a fixed-length application file	Defining a Fixed-Format Application File
define a standard file definition	Defining a Standard File Format or Variable-Length Application File
define a XML file definition	Defining an XML File Format in the *Gentran:Server for UNIX and Workstation XML User's Guide*
define an ODBC application file	Defining a Database File Format in the *Gentran:Server for UNIX and Workstation Application Integration ODBC User's Guide*.

How to Manually Activate EDI Standard Map Components

Introduction

When Gentran:Server generates the EDI standard sides of the map, the system includes and activates all the groups, segments, composites, and elements that the standard agency defines as mandatory for the document version you specified. You cannot deactivate mandatory groups, segments, composites, and elements.

During translation, the system processes only the components that are activated. Therefore, you must activate the non-mandatory groups, segments, composites, and elements to use them in your map. **You must activate all segments that you expect to receive from your trading partner, even if the segments are not mandatory.**

Tip
As an alternative to activation, you can use the Auto Trim feature if you have a sample EDI standard file defined.

Reference
See How to Automatically Activate Standard Map Components for more information on the Auto Trim feature.

Procedure

Use this procedure to activate groups, segments, composites, and elements.

Step	Action
1	From the **Functions** menu, select **Activate** Promote Split Activate Deactivate Link Unlink Generate TDF Generate UBFI *(Continued on next page)*

(Contd) Step	Action
2	On your map, open (double-click) each group that contains segments or groups that need to be activated. **Note** As an alternative to opening each map component (Steps 2, 5, and 7), you can select Expand All from the View menu to open every map component. You can also select a map component and select **Expand Branch** from the View menu to open that map component.
3	Select each inactive group that you need to use. **System Response** The system activates the groups. **Note** If you accidentally click a group, segment, composite, or element that you did not mean to activate, right-click the map component and select **Deactivate** from the shortcut menu.
4	Select each inactive segment that you need to use. **Note** This activates the segments.
5	Open (double-click) each segment that contains composites or elements that need to be activated.
6	Select each inactive composite that you need to use. **Note** This activates the composites.
7	Open (double-click) each composite that contains elements that need to be activated.
8	Select each inactive element that you need to use. **Note** This activates the elements.
9	Once you have activated all the necessary groups, segments, and elements, select **Activate** from the Functions menu to turn activation mode off. **Note** The function is off when the check mark no longer appears.

How to Automatically Activate Standard Map Components

Introduction   Instead of activating map components manually, you can use the Auto Trim feature to modify the EDI standard side of the map to match components in a sample EDI standard file that you select.

The Auto Trim feature examines the sample EDI standard file and then activates and deactivates map components on the EDI standard side of the map so that the two files match.

Before you begin   ◗   You must create the sample EDI standard file and save it as a file definition (with a .DDF file name extension) before using Auto Trim.

◗   Make sure that the sample EDI standard file uses the same standard, version, and transaction set (message) as the map that you want to Auto Trim.

◗   In addition, make sure that the file does not contain envelope segments.

Caveats   To use Auto Trim, you must define the Segment and Element Delimiters. If you do not supply the Tag Delimiter, the system substitutes the Segment Delimiter. If you want to use composite elements, you must specify the Sub Element Delimiter. If you want to use release characters, you must specify the Release Character.

Reference
See How to Verify and Set Delimiters for more information on specifying EDI delimiters.
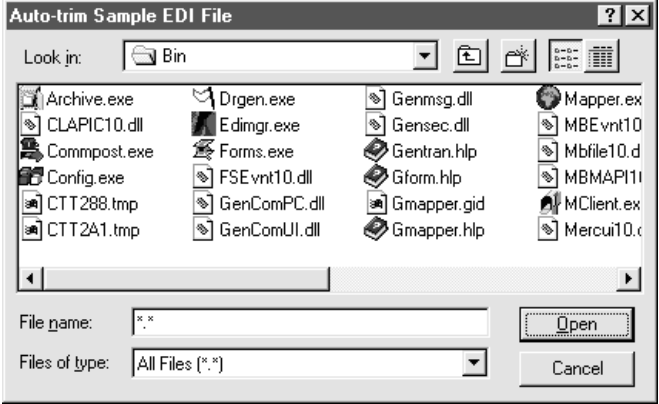
Warning
The sample EDI standard file must be compliant and must not contain envelope segments.

Procedure   Use the following procedure to use Auto Trim:

Step	Action
1	Right-click the **EDI File icon** on the EDI standard side of the map and select **Auto Trim** from the shortcut menu. **System Response** The system displays the Auto-trim Sample EDI File dialog box
2	From the **Look in** list, select the drive and folder where the EDI file is stored. **Note** The default is GENTRAN, the directory that contains the mapper.exe file.
3	If necessary, select the type of file (.DDF) that you want to display from the **Files of type** list.
4	From the list, select the sample EDI file that you want to load into the system.
5	Click **Open** to begin the auto trim process. **Note** When the auto trim process is complete, the system displays a message stating whether or not auto trim was successful.
6	Click **OK** to acknowledge the message.

Defining a Fixed-Format Application File

Overview

Introduction   If a side of your map is an application format, you must either define your application format to Gentran:Server or load a saved file definition. Your application file format or definition must define all the information that you need to extract from your partner's document (if the translation is inbound) or send to your partner (if the translation is outbound).

Component organization   The system arranges each map component sequentially in the order that it is most logical for the system to process. Groups contain related records or subgroups. The records contain related fields.

You must create each level of your application file definition sequentially, so plan to create the groups, subgroups, and records before you create the subordinate fields in your fixed-format file definition.

WARNING

Do not map to the control numbers.

Process   This table describes the process of defining a fixed-format application file.

Stage	Description
1	Obtain a layout of the necessary records, fields, and groups.
2	Create the groups and subgroups. **Reference** See How to Create a Group for instructions.
3	Create the first (header) record. **Reference** See How to Create the First Record for instructions.

(Contd) Stage	Description
4	Create the subsequent records. **Reference** See How to Create Subsequent Records for instructions.
5	Create the fields. **Note** If you want to read the Trading Partnership code from the file, you must create a field named TPCODE in the first record. **Reference** See How to Create Fields for instructions.
6	Define looping structures. **Reference** See Structuring Your Map for instructions.
7	Format your data. **Reference** See Formatting Your Data for instructions.

Saving the file format as a DDF

After you build your file format, Gentran:Server enables you to save the individual file format definition of the selected side of a map as a DDF file. You can use the file definition in future maps to quickly build either side of your map.

Reference
See How to Save a File Definition for instructions on saving a format definition as a file definition (.DDF) file.

See How to Load a File Definition for instructions.
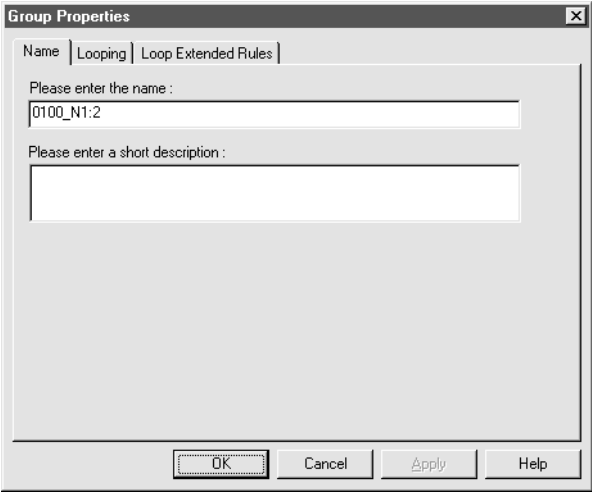
How to Create a Group

Introduction   A group contains related records, groups, or both that repeat in sequence until either the 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, this corresponds to a nested looping structure (a loop within a loop).

Procedure   Use this procedure to create a group.

Step	Action
1	Highlight the map component which precedes the group you are creating in the file layout.
2	From the **Edit** menu select **Create Sub** or **Insert**, depending on the level.
3	Select **Group** from the submenu. **System Response** The system displays the Group Properties dialog box (Name tab displayed by default) *(Continued on next page)*

(Contd) Step	Action
4	In the first box on the Name tab, type the group name. **Note** Do not use spaces or dashes (-) in the group name. You can use the underscore (_) to separate words.
5	In the second box on the Name tab, type the loop description. **Note** Use this box to provide a brief explanation of the loop.
6	Select the Looping tab to access looping options.
7	In the **Minimum Usage** box, type the minimum number of times the loop must be repeated. **Note** For a conditional loop, the minimum usage should always be 0 (zero). For a mandatory loop, the minimum usage should be 1 or greater.
8	In the **Maximum Usage** box, type the maximum number of times the loop can be repeated.
9	Is this a single iteration group? ◗ If YES, continue with Step 10. ◗ If NO, GO TO Step 11.
10	Do you want to specify that the subordinate records and groups should be extracted and located in the parent group when the group is compiled? ◗ If YES, select **Promote records to parent**. ◗ If NO, continue with Step 11.

11	Do you want to specify an extended rule for this group?
	
	◗ If YES, select the Loop Extended Rules tab.
	
	    Reference
	    See the chapter Using Extended Rules for more information on extended rules.
	
	◗ If NO, continue with the next step.

12	Click **OK** to create the group.

How to Create the First Record

Introduction

The first record in your application file definition is generally the header record. The first record is subordinate to the icon in the application side of the map. This shows how the records (and groups) are contained within it.

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

Trading Partnership code

If you want the system to look for the Trading Partnership code in the file, you have two options:

◗ Include a field named TPCODE in the first record in your application file definition to tell the translator where to look for the Trading Partnership Code. You must enter the name in all uppercase letters.

◗ Build Trading Partnership rules that extract information from multiple fields to cross-reference a Trading Partnership code.
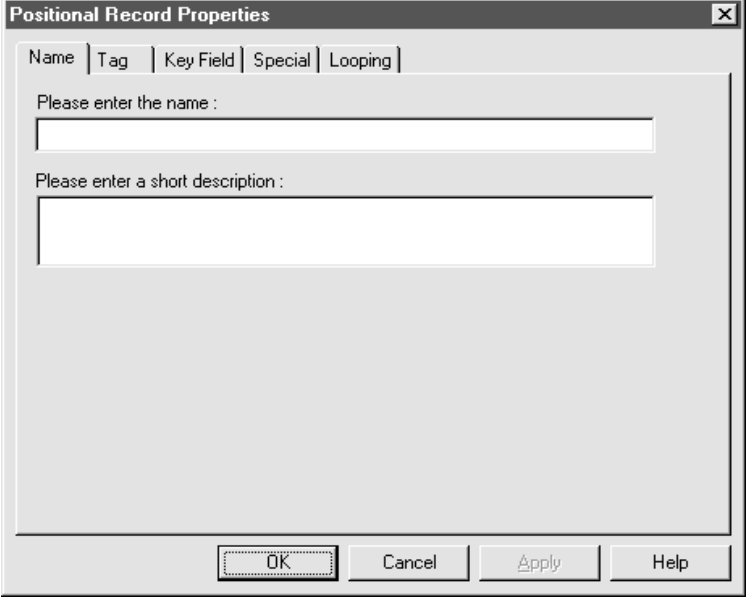
Reference
See Defining Trading Partnership Rules for Application Files for instructions on building Trading Partnership rules.

Procedure   Use this procedure to create the first record in the file definition.

Step	Action
1	Click the **Application File** icon to highlight it.
2	From the **Edit** menu, select **Create Sub** and then select **Record** from the submenu. **System Response** The system displays the Positional Record Properties dialog box. *[Positional Record Properties dialog box with Name, Tag, Key Field, Special, Looping tabs. Name tab shows "Please enter the name :" field and "Please enter a short description :" field. OK, Cancel, Apply, Help buttons.]*
3	In the first box on the Name tab, type the record name.
4	In the second box on the Name tab, type a description of the record. **Note** Use this box to provide a brief explanation of the record so that you can differentiate it from similar records.
5	Select the Tag tab to access tag information.

(Contd) Step	Action
6	In the **Tag** box, type the record identification code <TAG>.
	Note
	The record tag for each record enables the system to recognize that record and then determine the mapping requirements.
	Example
	The system recognizes a record as:
	<TAG>[Field_1][Field_2]
	However, the tag does not have to occur before the fields.
	Recommendation
	Define the record tag on the Positional Record Properties dialog box, instead of defining fields with the purpose of explicitly containing the record tag.
7	In the **Position** box, type the starting column position of the tag in the data record.
8	Do you want to specify floating or wildcard options?
	◗ If YES, select the Special tab and specify the options.
	◗ If NO, continue with the next step.
	Note
	A floating record does not have a fixed position in the file.
9	Is the record a looping structure?
	◗ If YES, select the Looping tab and continue with Step 10.
	◗ If NO, go to Step 12.
10	On the Looping tab in the **Min Usage** box, type the minimum amount of times the record must repeat.
	Recommendation
	Type **1** to make the header record mandatory.
	Note
	If the **Min Usage** box contains a 0 (zero), the record is conditional. If the **Min Usage** box contains a 1 or greater, the record is mandatory.

(Contd) Step	Action
11	In the **Max Usage** box, type the maximum amount of times it can repeat. **Recommendation** For the header record, we recommend that you accept the default of 1 so it remains a record that only occurs once (not a looping structure).
12	Click **OK** to create the record.

How to Create Subsequent Records

Introduction

After you create the first record, you can define subsequent records at the same level. You can also define groups, if your application includes looping structures. You may need to define a loop if your application requires a group of related records (and/or subgroups) that repeat in sequence.

Reference
See How to Create Fields for more information on defining groups.

Procedure

Use this procedure to create additional records.

Step	Action
1	Highlight the map component that precedes the record you are creating in the file layout.
2	From the **Edit** menu select **Create Sub** or **Insert**, depending on the level.
3	Select **Record** from the submenu. **System Response** The system displays the Positional Record Properties dialog box.
4	In the first box on the Name tab, type the record name.
5	In the second box on the Name tab, type a description of the record. **Note** Use this box to provide a brief explanation of the record so that you can differentiate it from similar records.
6	Select the Tag tab to access tag information.

(Contd) Step	Action
7	In the **Tag** box, type the record identification code <TAG>. **Comment** The record tag for each record enables the system to recognize that record and then determine the mapping requirements. **Example** The system recognizes a record as: <TAG>[Field_1][Field_2] Note that the tag does not have to occur before the fields. **Recommendation** Define the record tag on the Positional Record Properties dialog box, instead of defining fields with the purpose of explicitly containing the record tag.
8	In the **Position** box, type the starting column position of the tag in the data record.
9	Do you want to specify floating (the record does not have a fixed position in the file)? ◗ If YES, select the Special tab and complete the appropriate boxes. ◗ If NO, accept the defaults and continue with the next step.
10	Is the record a looping structure? ◗ If YES, select the Looping tab and continue with the next step. ◗ If NO, accept the defaults and GO TO Step 13.
11	On the Looping tab in the **Min Usage** box, type the minimum amount of times the record must repeat. **Recommendation** Type **1** to make the header record mandatory. **Note** If the **Min Usage** box contains a 0 (zero), the record is conditional. If the **Min Usage** box contains a 1 or greater, the record is mandatory. (Continued on next page)

(Contd) Step	Action
12	In the **Max Usage** box, type the maximum amount of times the record can repeat. **Comment** The value must be between 1 and 4294967295.
13	Click **OK** to create the record. **Note** Use this procedure to create any additional records.

How to Create Fields

Introduction
Each record you create contains a group of logically-related application fields. These fields define the structure and content of the data to the system.

The easiest way to add application fields to a record is to use the Positional Field Editor, because it allows you to add multiple fields. Generally, you create the fields for the first record in the application file definition, and then proceed with each sequential record.

If you have only one field to add, you can use either the Create sub field or the Insert Field functions.

Recommendation
Do not define fields with the purpose of using the field to explicitly contain the record tag. This is because the system takes the tag that you define in the record into account when the automatic sequencing (Auto Position) feature is used. We recommend that you define the record tag on the Positional Record Properties dialog box.

Reference
See How to Create Subsequent Records for more information on the Positional Record Properties dialog box.
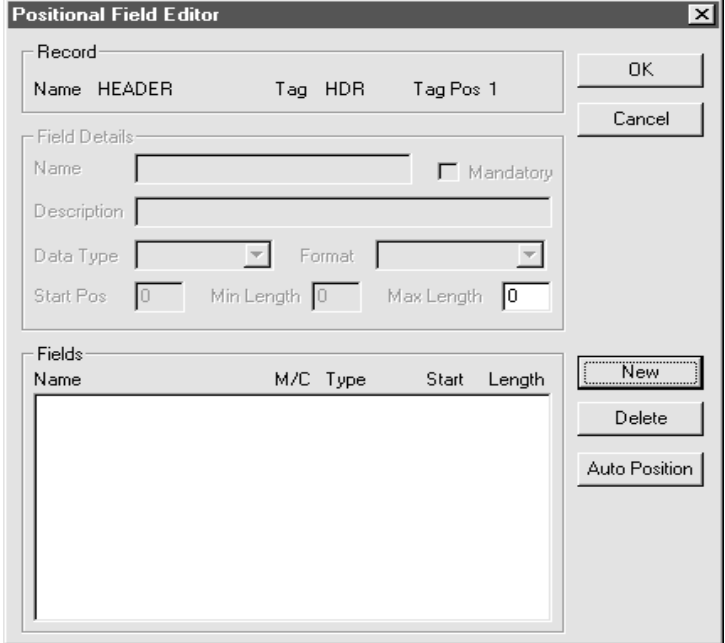
Trading Partnership code
If you want the system to look for the Trading Partnership code in the file, you must define a field named TPCODE (in the first record of an application file definition). Enter the name TPCODE in all uppercase type.

This field enables the translator to locate the Trading Partnership Code.

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

Step	Action
1	Right-click the application record and select **Edit Fields** from the shortcut menu. **System Response** The system displays the Positional Field Editor dialog box.
2	Is the field you are creating the first field in the record? ◗ If YES, click **New** and continue with the next step. ◗ If NO, highlight the field that precedes the field you are creating in the record layout and click **New**. **System Response** The system displays a highlight bar in the Fields section where the new field is positioned. **Note** You need to fill in the field values in the Field Details section. (Continued on next page)

(Contd) Step	Action
3	In the **Name** box, type the field name. **Notes** ◗ 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. ◗ Do not use spaces or dashes (-) in the field name. You can use the underscore (_) to separate words.
4	Do you want to designate the field as mandatory (must be present)? ◗ If YES, select the **Mandatory** check box. ◗ If NO, continue with the next step.
5	In the **Description** box, type a description of the field. **Note** Use the description to provide a brief explanation of the field so that you can differentiate it from similar fields.
6	From the **Data Type** list, select the type of the field. **Note** Valid values are: ◗ String=alpha field ◗ Number=numeric or real field ◗ Date/Time=date or time field
7	From the **Format** list, select the field format. **Note** The choices for this field depend on the type of field you selected from the Data Type list. If you choose Number or Date/Time in the **Data Type** box, you can select the data format from the Format list. If you selected String from the **Data Type** box, you should type a syntax token to denote that this field must be formatted as the specified syntax token dictates (the default syntax token is X). **Reference** See Formatting Your Data for more information on selecting your data format. *(Continued on next page)*

(Contd) Step	Action
8	Do you want to indicate the exact position of the field in the record?
	◗ If YES, type the starting position of the field in the **Start Pos** box.
	◗ If NO, continue with the next step.
	Notes You want to specify field start positions if, for example, you are using only a few fields but you want them positioned exactly in the record.
	The alternative to specifying the start position of each field is to add the fields sequentially in the record and then use the Auto Position function.
	Reference See Step 13 below for more information on the Auto Position function.
9	In the **Min Length** box, type the minimum number of characters that is acceptable for this field.
10	In the **Max Length** box, type the maximum length of the field.
11	Click **New**.
	System Response The system adds the field and creates a new field with blank values ready for you to identify.
12	Create the rest of the fields according to your record layout.
	Note Click **Delete** to stop adding fields.
	(Continued on next page)

(Contd) Step	Action
13	After adding the last field, do you want the system to automatically position the fields in the record? ◗ If YES, click **Auto Position**. **Note** This option automatically calculates the start position in the record of each field. The system bases the start positions on the assumption that each field is positioned directly after the previous field and is of the length specified in the **Max Length** box. Click **Yes** to acknowledge the warning message that fields are sequenced in order. **Warning** Use the Auto Position function only if you define a record tag, and if you define every field in the record in the sequence that each field occurs. **Reference** See How to Create Subsequent Records for more information on defining record tags. ◗ If NO, continue with the next step
14	Are you completely finished adding fields to the record? ◗ If YES, click **Close**. ◗ If NO, repeat Steps 2 through 13. **Note** Repeat this procedure to add fields to the other records you defined. **References** See How to Use a Constant in a Map for more information on mapping a constant to a field.
15	Do you want to define Trading Partnership code rules for this application or definition file? ◗ If YES, see the chapter Defining Trading Partnership Rules for Application Files for instructions. ◗ If NO, continue with another task.

Defining a Standard File Format or Variable-Length Application File

Overview

Introduction

The procedures for defining a standard file format and for defining a variable-length application file are similar. Both are based on a Delimited EDI syntax.

EDI standard format

When you create a map and specify that the input or output side is a standard format, Gentran:Server generates the EDI standard file format for you, based on the standard (agency), version, transaction set, and release you selected. The system includes all the groups, segments, composites, and elements that the standards agency defines for the version of the document you selected.

If you want to use a specialized version of a standard, you may have to define the EDI file to Gentran:Server. One way to define a proprietary standard is to customize a system-generated EDI standard file format. You can:

◗ change the properties of the map components

◗ use the Promote, Split, Copy, Cut, and Paste functions to define looping structures.

Variable-length application format

If the fields in your application format are variable-length rather than fixed-length, and you do not have a file definition, you must create a new data format, basing your application syntax on delimited EDI format.

When you create a new data format for your variable-length application file, you select **Delimited EDI** as the data format syntax and set the field separator (delimiter) in the map. The specific map components that you add use depend on the type of map you are creating.

Process   This table describes the process of defining a standard file format or a variable-length application file.

Stage	Description
1	Determine which map components you are using. Note the standard, version, and transaction set (document), and the groups, segments, composites, and elements your company requires.
2	Verify and set the EDI delimiters. **Reference** See the <u>How to Verify and Set Delimiters</u> topic.
3	Modify the Group Properties. **Reference** See the <u>How to Modify Group Properties</u> topic.
4	Modify the segment properties. **Reference** See the <u>How to Modify Segment Properties</u> topic.
5	Modify the composite properties. **Reference** See the <u>How to Modify Composite Properties</u> topic.
6	Modify the element properties. **Reference** See the <u>How to Modify Element Properties</u> topic.
7	Define and modify relational conditions. **Reference** See the <u>How to Define and Modify Relational Conditions</u> topic.
8	Define looping structures. **Reference** See the <u>Structuring Your Map</u> topic.
9	Format your data. **Reference** See <u>Formatting Your Data</u> for instructions.

**Saving the file
format as a DDF**

After you build your file format, Gentran:Server enables you to save the individual file format definition of the selected side of a map as a DDF file. You can use the file definition in future maps to quickly build either side of your map.

Reference
See How to Save a File Definition for instructions on saving a format definition as a file definition (.DDF) file.

See How to Load a File Definition for instructions.

How to Verify and Set Delimiters

Definition

Delimiters are flags that you define to the system as separating specific variable-length application file components.

Note
Delimiters are also used to separate components in standards files. You set the delimiters for standards in the Trading Partnership record.

Why delimiters are used

You must use delimiters for all application file definitions with variable-length fields. This is because the system compresses the data, removing the leading zeroes and trailing blanks. Since the fields vary in length, the system needs a flag to determine where one element ends and another begins. An element delimiter marks the beginning of a new element.

Reason for verification

If you are using a variable-length application file definition that contains composite elements or subelements, you must verify that Gentran:Server is specifying the correct delimiters so that your data is processed correctly.

New lines vs. streamed format

The **Stream segments** option on the Delimiters tab of the File Properties dialog box controls the format of output segments.
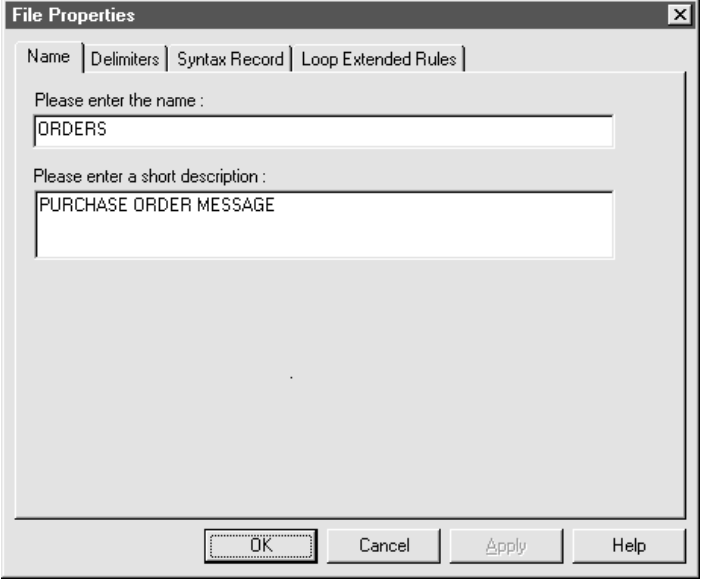
If you do not select this option, each segment ends with the segment terminator followed by a carriage return or line feed. This is the expected format for non-EDI standard data, such as variable-length application files. By default, this option is not selected.
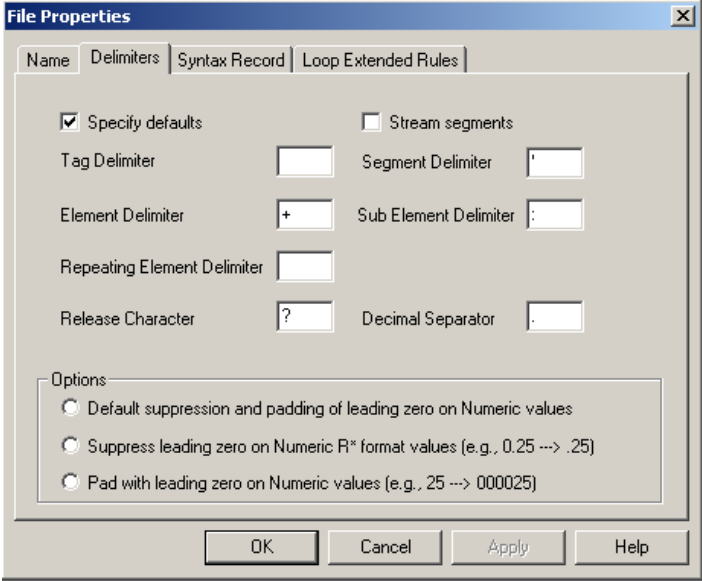
If you select this option, each segment follows another, without a carriage return or line feed to separate them. This is the expected format for EDI standard data.

Procedure   Use this procedure to verify and set delimiters.

Step	Action
1	Right-click the **EDI file icon** and select **Properties** from the shortcut menu. **System Response** The system displays the File Properties dialog box. (Continued on next page)

(Contd) Step	Action
2	Select the **Delimiters** tab to access delimiter options.
3	Select the **Specify defaults** check box.
4	Is the **Stream segments** check box selected? ◗　If YES, deselect it. ◗　If NO, continue with Step 5.
5	Compare the required delimiters for the application file you are using to those indicated. Do you need to change or set one or more delimiters? ◗　If YES, type either the character or the hexadecimal value in the correct box or boxes. ◗　If NO, continue with Step 6.
6	Is the file you used in Step 1 on the output side of a map? ◗　If YES, continue with Step 7. ◗　If NO, go to Step 10. (Continued on next page)

(Contd) Step	Action
7	Do you want to suppress leading zeros on all numeric values greater than 1 (default behavior)? ◗ If YES, verify that the **Default suppression and padding of leading zero on Numeric values** option is selected and go to Step 10. ◗ If NO, continue with Step 8.
8	Do you want to suppress leading zeros on numeric values using the R format? ◗ If YES, select the **Suppress leading zero on Numeric R* format values (e.g., 0.25 ---> .25)** option and go to Step 10. ◗ If NO, continue with Step 9.
9	Do you want to pad numeric values (regardless of format) with leading zeros? ◗ If YES, select the **Pad with leading zero on Numeric values (e.g., 25 ---> 000025)** option and go to Step 10. ◗ If NO, go to Step 7.
10	Click **OK** to exit the File Properties dialog box.

Delimiters tab fields and functions

This table describes the **Delimiters tab** fields and their functions.
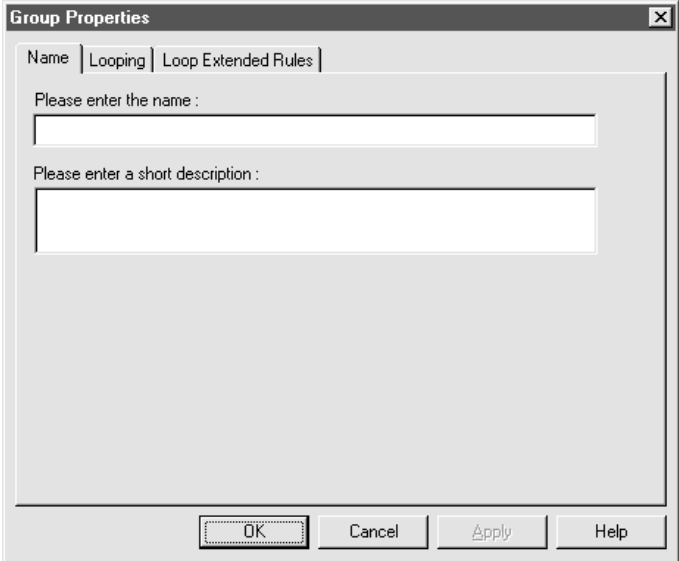
Field	Function
Specify defaults	Allows you to define the delimiters used to separate specific EDI components.
Stream segments	Specifies whether the translator writes segments without a carriage return/ line feed at the end. The default is not checked, which indicates that each segment is written with a carriage return/line feed at the end.
The following fields are only available if the **Specify defaults** check box is selected.	
Tag Delimiter	Default character that marks the end of each segment tag (identifier).

Segment Delimiter	Default character that marks the end of each segment.
Element Delimiter	Default character that marks the end of each element.
Sub Element Delimiter	Default character that marks the end of each subelement.
Repeating Element Delimiter	Default character that marks the start of a new repetition of an element or subelement.
Release Character	Character that, when used before any defined delimiter, restores the character used for the delimiter to its original meaning. For example, if the plus sign (+) is the element delimiter and question mark (?) is the release character, when ?+ is a part of the data, Gentran:Server reads the + as a part of the data instead of marking the end of the element.
Decimal Separator	Character that indicates the decimal point in a numeric field, such as a period (.) or comma (,).
The following radio buttons in the **Options** section are only available if the file is on the output side of the map.	
Default suppression and padding of leading zero on Numeric values	Leading zeros are suppressed for all values greater than 1. Decimal values less than 1 will retain a single leading zero (e.g., 000.25 --> 0.25). This is the default selection.
Suppress leading zero on Numeric R* format values (e.g., 0.25 ---> .25)	Same as default behavior except leading zeros are also suppressed on elements that use an R-format numeric with a value less than 1.
Pad with leading zero on Numeric values (e.g., 25 ---> 000025)	Numeric values, regardless of format, are padded with leading zeros out to the maximum length of the field.

How to Modify Group Properties

Introduction   A group 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. If you create a group that is subordinate to another group, this corresponds to a nested looping structure (a loop within a loop).

Procedure   Use this procedure to modify the properties of a group.

Step	Action
1	Right-click the group you want to modify and select **Properties** from the shortcut menu. **System Response** The system displays the Group Properties dialog box (Name tab displayed by default).
2	If you want to modify the group name, in the first box on the Name tab, type the new name. **Note** Do not use spaces or dashes (-) in the group name. You can use the underscore (_) to separate words. *(Continued on next page)*

(Contd) Step	Action
3	Do you want to modify the group description? ◗ If YES, type the new description in the second box on the Name tab. ◗ If NO, continue with the next step.
4	Select the **Looping** tab to access looping options.
5	In the **Minimum Usage** box, type the minimum number of times the loop must be repeated. **Note** For a conditional loop, the minimum usage should always be 0 (zero). For a mandatory loop, the minimum usage should be 1 or greater.
6	In the **Maximum Usage** box, type the maximum number of times the loop can be repeated.
7	Is this a single iteration group? ◗ If YES, select **Promote records to parent** if you want to specify that the subordinate records and groups should be extracted and located in the parent group when the group is compiled. ◗ If NO, continue with the next step.
8	Do you want to specify an extended rule for this group? ◗ If YES, select the Loop Extended Rules tab and define the rule; then continue with the next step. **Reference** See the chapter <u>Using Extended Rules</u> for more information on extended rules. ◗ If NO, continue with the next step.
9	Click **OK** to save changes to the group.
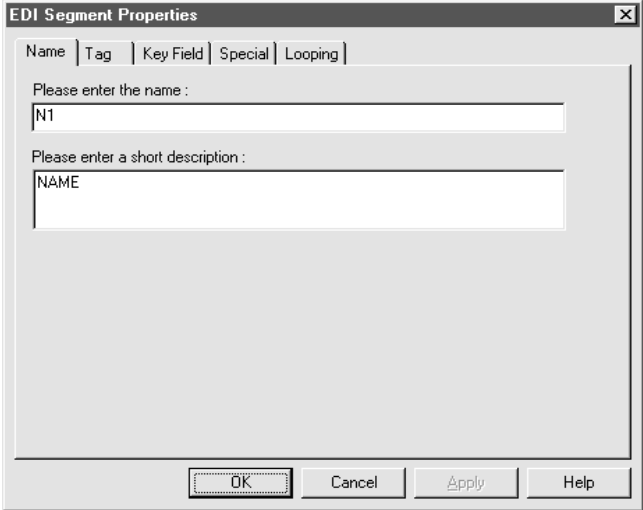
How to Modify Segment Properties

Introduction   You can modify the properties of any segment, including the minimum and maximum number of times the segment can repeat, whether the segment is mandatory or conditional, and whether it is a loop start or loop end segment.

Reference
See How to Use Loop Start and Loop End Segments for more information on using loop start and loop end segments.

Procedure   Use this procedure to modify the properties of a segment.

Step	Action
1	Right-click the segment you want to modify and select **Properties** from the shortcut menu **System Response** The system displays the EDI Segment Properties dialog box. (Continued on next page)

(Contd) Step	Action
2	If you need to modify the name of the segment, type the segment name in the first box on the Name tab. **Note** If a segment occurs more than once in a map it is identified by its name <ID>. The second and subsequent occurrences are identified by <ID>:n, where n is the number of the occurrence in the map.
3	If you want to modify the description of the segment, type a new description in the second box on the Name tab. **Note** Use this box to provide a brief explanation of the segment so that you can differentiate it from similar segments.
4	Select the **Tag** tab only if you want to access tag information.
5	If you want to modify the segment tag, type the segment identification code <TAG> in the **Tag** box. **Note** The segment tag for each segment enables the system to recognize that segment. **Example** The system recognizes a record as: <TAG>[Delimeter]<DATA>[Delimeter] <DATA>.......<DATA>[Segment Terminator]
6	Select the **Key Field** tab only if you want to use a key field. **Note** The key field function enables you to specify a second qualification in selecting a segment (the segment name is the first qualification). Data must be provided in the order designated through the use of key fields. Note that the Matching Rules section of the tab is disabled if the segment is on the output side of the map. (Continued on next page)

(Contd) Step	Action
7	Select the **Special** tab only if you want to specify that the segment: ◗ is floating (the segment does not have a fixed position in the file), or ◗ contains binary data. Otherwise, accept the defaults. **Reference** See How to Use Loop Start and Loop End Segments for information about processing binary data.
8	Is the segment a looping structure? ◗ If YES, select the **Looping** tab. ◗ If NO, accept the defaults and continue with step 12.
9	In the **Min Usage** box, type the minimum amount of times the segment must repeat. **Note** If the **Min Usage** box contains a 0 (zero), the segment is conditional. If the **Min Usage** box contains a 1 or greater, the segment is mandatory.
10	Do you want the segment to repeat (loop)? ◗ If YES, type the maximum amount of times it can repeat in the **Max Usage** box. ◗ If NO, continue with the next step.
11	Do you want to indicate that the segment is the beginning of a loop (Loop Start) or the end of a loop (Loop End)? ◗ If YES, select one of the following options. **Loop Start** (this segment marks the beginning of the loop) **Loop End** (this segment marks the end of the loop) ◗ If NO, accept the default of Normal for the Normal/Loop Start/Loop End options, which indicates that this segment is in the loop but is not the beginning or ending segment
12	Click **OK** to save changes to the segment.

How to Use Loop Start and Loop End Segments

Introduction

Certain EDI standards use Loop Start (LS) and Loop End (LE) segments. LS and LE segments differentiate between two or more loops of the same type. If the transaction contains LS and LE segments and depending on whether your map is inbound or outbound, you need to define the LS and LE segments for the loops you are using in the map in one of two different ways.
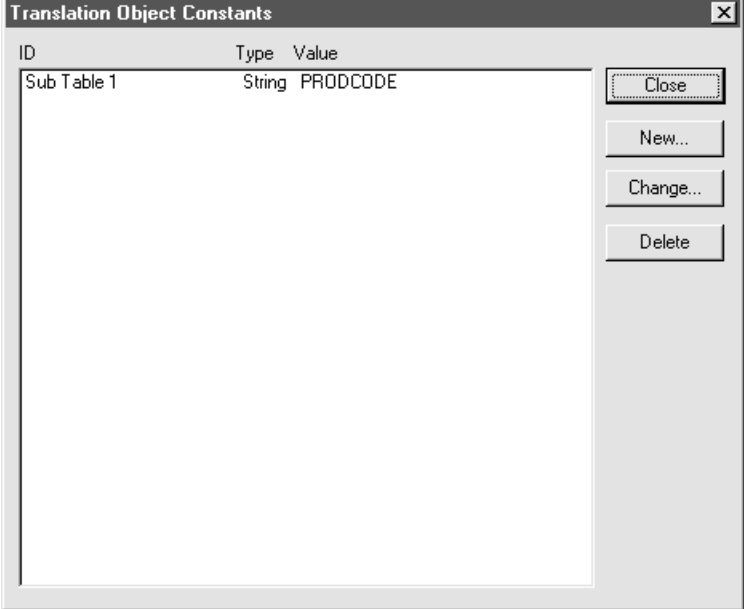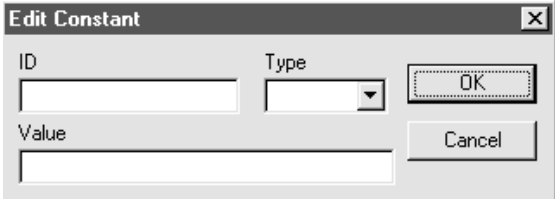
Tip
You need to communicate with your trading partner to determine which loop identification code your partner sends you (for an inbound map), or expects to receive (for an outbound map).

Procedure: defining a LS segment inbound

Use this procedure to define an LS segment for an inbound map.

Step	Action
1	Right-click the LS segment and select **Properties** from the shortcut menu. **System Response** The system displays the EDI Segment Properties dialog box.
2	Select the **Looping** tab to access the loop options and verify that the **Loop Start** option is selected. (Continued on next page)

(Contd) Step	Action
3	Click **Edit** (at the right of the Use constant list). **System Response** The system displays the Translation Object Constants dialog box.
4	Click **New**. **System Response** The system displays the Edit Constant dialog box.
5	In the **ID** box, type the literal constant identifier.
6	From the **Type** list, select **String**. **Comment** This is the category of this literal constant. (Continued on next page)

(Contd) Step	Action
7	In the **Value** box, type the value of the literal constant. **Note** For an inbound map, this is the loop identifier code that you expect to receive from your trading partner.
8	Click **OK** to add the constant to the system.
9	Select the constant you created from the list. **Note** The system matches the selected constant against the loop identifier code.
10	Click **OK** to exit the EDI Segment Properties dialog box.

Procedure: defining a LE segment inbound

Use this procedure to define an LE segment for an inbound map.
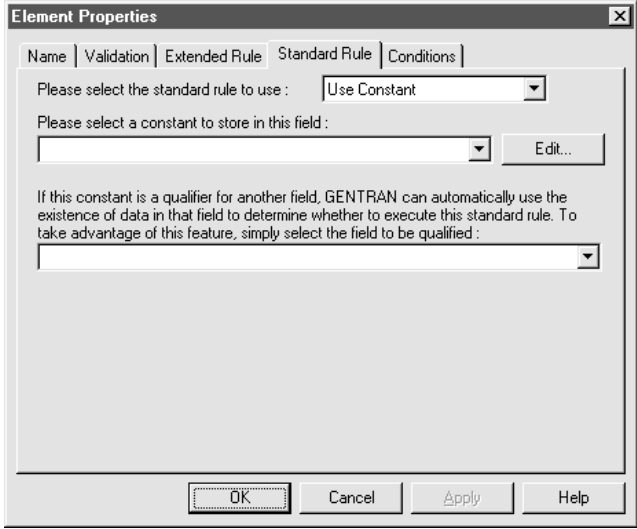
Step	Action
1	Right-click the LE segment and select **Properties** from the shortcut menu. **System Response** The system displays the EDI Segment Properties dialog box.
2	Select the **Looping** tab to access the loop options and verify that the **Loop End** option is selected.
3	Select the **Key Field** tab. **System Response** The system displays the key field options.
4	From the **Field** list, select the **Loop Identifier Code**. **Note** This list contains all the elements that are contained in the segment and gives you the ability to define the Loop End segment definition, by specifying that the loop identifier code must have the value you specify in the Matching rules section. (Continued on next page)

(Contd) Step	Action
5	Select the constant you created for the LS segment from the list. **Notes** ◗ The system matches the selected constant against the loop identifier code. ◗ The Matching rules section enables you to access all the literal constants and code lists currently defined for this map.
6	Click **OK** to exit the EDI Segment Properties dialog box.

Procedure: defining a LS segment outbound

Use this procedure to define an LS segment for an outbound map.

Step	Action
1	Double-click the **Loop Identifier Code** element in the LS segment. **System Response** The system displays the Element Properties dialog 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 **Use Constant**. **System Response** The system displays the constant options.
4	Click **Edit**. **System Response** The system displays the Translation Object Constants dialog box.
5	Click **New**. **System Response** The system displays the Edit Constant dialog box.
6	In the **ID** box, type the literal constant identifier.
7	From the **Type** list, select **String**. This is the category of this literal constant.
8	In the **Value** box, type the value of the literal constant. For an outbound map, this is the loop identifier code that your partner is expecting to receive from you.
9	Click **OK** to add the constant to the system.
10	Click **Close** to exit the Translation Object Constants dialog box. (Continued on next page)

(Contd) Step	Action
11	From the constant list, select the constant that you created. **Note** The system matches the selected constant against the loop identifier code.
12	Click **OK** to exit the Element Properties dialog box.

Procedure: defining a LE segment outbound

Use this procedure to define an LE segment for an outbound map.

Step	Action
1	Double-click the Loop Identifier Code element in the LE segment. **System Response** The system displays the Element Properties dialog box.
2	Select the **Standard Rule** tab to access the standard rule options.
3	From the standard rule list, select **Use Constant**. **System Response** The system displays the constant options.
4	From the constant list, select the constant that you created. **Note** The system matches the selected constant against the loop identifier code.
5	Click **OK** to exit the Element Properties dialog box.
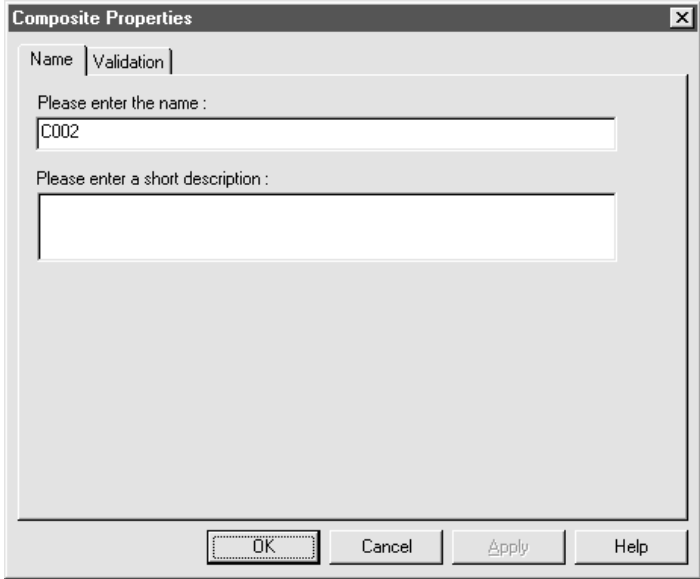
How to Modify Composite Properties

Introduction   A composite is a data element that contains two or more component data elements or subelements. You can modify the composite name and description, and whether or not it is mandatory.

Note
Composite elements are used only in EDI standard maps.

Procedure   Use this procedure to modify the properties of a composite.

Step	Action
1	Right-click the composite you want to modify and select **Properties** from the shortcut menu. **System Response** The system displays the Composite Properties dialog box. (Continued on next page)

(Contd) Step	Action
2	If you want to change the composite name, type the data element identification number in the first box on the Name tab. **Note** If a composite occurs more than once in a map it is identified by its name <ID>. The second and subsequent occurrences are identified by <ID>:n, where n is the number of the occurrence in the map.
3	If you want to change the composite description, type the new description in the second box on the Name tab.
4	Select the **Validation** tab if you want to specify that the composite is mandatory.
5	Select the check box on the Validation tab to indicate whether this composite must be present.
6	Click **OK** to save the changes to the composite.
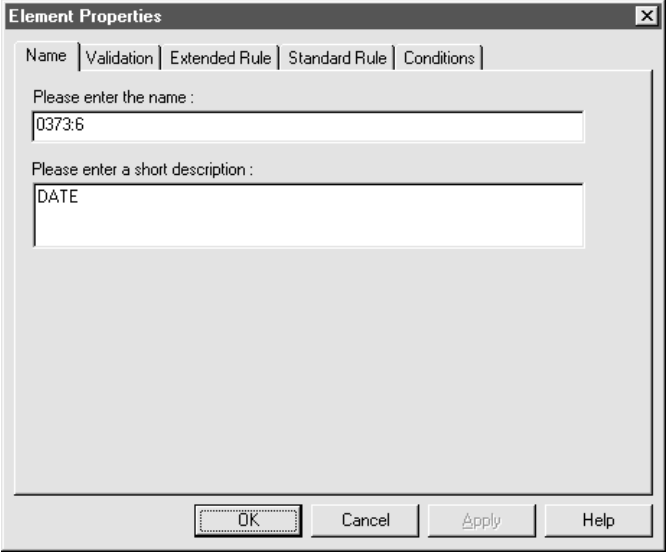
How to Modify Element Properties

Introduction   Each segment or composite contains a group of logically-related elements. These elements define the structure of the EDI standard data that your system needs to process the document.

Procedure: modifying element properties

Use this procedure to modify the properties of an element.

Step	Action
1	Double-click the element you want to modify. **System Response** The system displays the Element Properties dialog box. (Continued on next page)

(Contd) Step	Action
2	Do you want to change the name of the element? ◗ If YES, type the new name in the first box on the Name tab. The name is typically the element sequence number. ◗ If NO, continue with the next step. **Note** If an element occurs more than once in a map, the system identifies the first occurrence by the element's name <ID>. The second and subsequent occurrences are identified by <ID>:n, where n is the number of the occurrence in the map.
3	Do you want to change the description of the element? ◗ If YES, type the new description in the second box on the Name tab. ◗ If NO, continue with the next step. **Tip** Use the description to provide a brief explanation of the element so that you can differentiate it from similar elements.
4	Do you want to change the validation options? ◗ If YES, select the **Validation** tab and continue with the next step. ◗ If NO, GO TO Step 13.
5	Do you want to designate the element as mandatory (must be present)? ◗ If YES, select the first check box on the Validation tab. ◗ If NO, continue with the next step.
6	Do you want to change the minimum length of the element? ◗ If YES, type the minimum length in the **Minimum** box. ◗ If NO, continue with the next step. **Note** If the data is less than the minimum length, the system generates a compliance error during translation.
7	Do you want to change the maximum length of the element? ◗ If YES, type the maximum length in the **Maximum** box ◗ If NO, continue with the next step. (Continued on next page)

(Contd) Step	Action
8	Do you want to change the type of the element? ◗ If YES, from the **Data Type** list, select the type of the element. ◗ If NO, continue with the next step. **Note** Valid values are: ◗ String=alpha element ◗ Number=numeric or real element ◗ Date/Time=date or time element
9	Do you want to change the data format of the element? ◗ If YES, from the **Format** list, select how the data element is formatted. ◗ If NO, continue with the next step. **Note** The choices for this field depend on the type of element you selected from the **Data Type** list. If you choose **Number** or **Date/ Time** in the **Data Type** box, you can select the data format from the **Format** list. If you selected String from the **Data Type** box, you should type a syntax token to denote that this field must be formatted as the specified syntax token dictates. The default syntax token is X. **Reference** See Formatting Your Data for more information on formatting your data.
10	Click **OK** to save the changes to the element.

How to Define and Modify Relational Conditions

Introduction
You can use relational conditions to connect fields together for syntax or compliance reasons. For example, assume that Field A is invalid unless Field B is present. If you set up a condition that pairs Fields A and B, the system generates a compliance error if one of those fields is not present.

You can also view the conditional relationships between elements, as the standard provides.
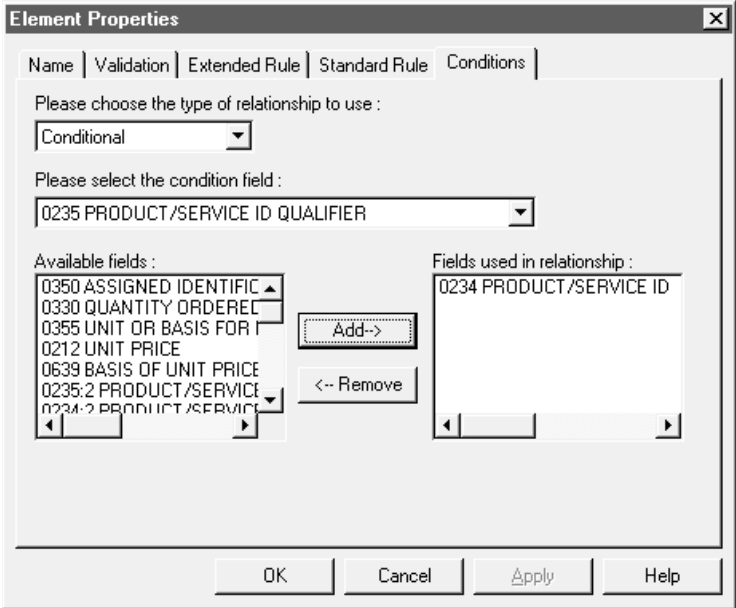
Warning
Though the system allows you to edit EDI conditional relationships, it generates a compliance error when you do edit them. We recommend that you only view EDI conditions.

Procedure
Use this procedure to define, modify, and view field and element relational conditions.

Step	Action
1	Double-click the field or element you want to modify. **System Response** The system displays the Field (or Element) Properties dialog box. (Continued on next page)

(Contd) Step	Action
2	Select the **Conditions** tab. **System Response** The system displays the relational condition options.
3	Select the field connection condition from the type of relationship list. **Note** Valid values are: ◗ **Paired/Multiple** – if any of the specified fields are present then all fields must be present ◗ **Required** – at least one of the specified fields must be present ◗ **Exclusion** – no more than one of the specified fields may be present ◗ **Conditional** – if the first Condition field is present, the rest of the fields must also be present ◗ **List Conditional** – if the first Condition field is present, at least one of the specified fields must also be present (Continued on next page)

(Contd) Step	Action
4	Select the first field from the condition field list. **Notes** ◗ This is the field on which the conditional relationship hinges if you chose Conditional or List Conditional from the type of relationship list. ◗ The condition field list is only active if you chose either Conditional or List Conditional from the type of relationship list.
5	From the **Available fields** list, select a field or fields and click **Add**. **System Response** The system moves the fields to the Fields used in relationship list, to include the fields as a part of the conditional relationship. **Notes** ◗ The Available fields list contains all the fields in the translation object that are valid to be used in a condition at this point. ◗ The Fields used in relationship list contains the fields that you selected to be a part of the conditional relationship. ◗ To remove the fields from the conditional relationship, select a field or fields from the Fields used in relationship list and click **Remove** to move the fields back to the Available fields list.
6	Click **OK** to add the conditional relationship to the field.

Structuring Your Map

Overview

Introduction

This section describes how to structure your map by:

◗ mapping or linking components on the input side of a map to components on the output side

◗ creating repeating groups of related segments (loops)

In this section, you learn how to use the:

◗ Cut, copy, and paste functions to create a loop

◗ Promote function to extract one instance of a group or repeating record

◗ Split function to break a group or repeating record into two loops

◗ Equalize function to make the size of each map side the same

Links

The Link function enables you to map a field from the input side of the map to a field 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.

Loop

In many documents, related segments are grouped to convey a complete piece of information. For example, a company name, street address, city, state, zip code, and contact name form a complete business mailing address. If the group of related segments can be repeated, it is called a loop.

In a Gentran:Server map, a loop is a group of related records or subgroups that repeat in sequence until either the data ends or the maximum number of times that the loop is allowed to repeat is exhausted.

You can create loops in your input or output format to represent repeating units of data.

Nested loop   A nested loop is a loop within another loop.

Example
When you create a group that is subordinate to another group, you create a nested looping structure (a loop within a loop).

How to Create Simple Links

Introduction   The Link function is used to map a component on the input side of a map to a component on the output side of the map.

Notes
You must have at least one direct link to every record on the output side of the map, so the translator can create the record.

If you need to link two input fields to the same output field because of conditions established in your map, you must use an extended rule.

WARNING

Do not map to the control numbers.

Procedure   Use this procedure to link two fields.

Step	Action
1	From the **Functions** menu, select **Link**.
2	Select a field on the input side of the map. **System Response** The cursor changes to a Link Arrow: LINK ⇨
3	Select the field on the output side of the map that the data from the field selected on the input side of the map is mapped to. **System Response** A line connects the two fields to visually represent the link. **Note** If a line does not appear when the two fields are linked, check the Preferences dialog box. **Reference** See How to Customize the Display of Links in this guide for more information. On the Links tab of the Preferences dialog box, you should specify Show all links possible.

How to Copy, Cut, and Paste Components

Introduction

The Copy, Cut, and Paste functions move information in the map. You can cut or copy a single map component (loop, record, or field) and paste it in another location in the map. You can also cut, copy, and paste a map component from one map to another.

Note
The map component you paste must be a valid component for the format. If you cut or copied an ODBC component, you can paste that component only into the ODBC side of a map. Conversely, if you cut or copied an application component, you can paste that component only into the application side of a map.

When to use

Use these functions to create nested looping structures.

Information retained

Copied map components retain all the information of the original map component. If the copied map component contains subordinate map components, such as subordinate fields, the system copies the subordinate map components, too.

Procedure

Use this procedure to cut, copy, and paste a map component.

Step	Action
1	Select the map component that you want to cut or copy.
2	From the **Edit** menu, select **Copy** or **Cut**. **System Response** The Application Integration subsystem places the map component on the Clipboard.
3	Do you want to paste the component into another map? ◗ If YES, open the map you want to modify. ◗ If NO, continue with the next step. (Continued on next page)

(Contd) Step	Action
4	Select the map component that you want the cut or copied selection pasted *after*. **Note** Make sure that the component you are pasting is valid for the side of the map. You can paste an ODBC component into the ODBC side of a map, but you cannot paste it into the application side of a map.
5	Click **Paste** on the Main Toolbar to paste the map component from the Clipboard to the map. **System Response**

	IF the map component is…	THEN the system…
	a group	prompts you to specify whether you want the contents of the Clipboard pasted as a child (subordinate) of the group or pasted at the same level as the group.
	not a group	pastes the component into the map.

6	Did you receive the prompt in Step 5? ◗ If YES, select the appropriate option and click **OK.** ◗ If NO, you are finished.

How to Promote Groups and Repeating Records

Introduction

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

Promote vs. copy vs. split

You can use the Copy, Cut and Paste functions to accomplish the same task. If you use Copy, Cut, and Paste, you must change the number in the **Maximum usage** box.

Depending on the circumstances, you can also use the Split function to extract one iteration.

References

◗ See How to Create Simple Links for information on copying, cutting, and pasting.

◗ See How to Split Groups and Repeating Records for information on splitting groups and repeating record.

Procedure

Use this procedure to promote a group or repeating record.

Step	Action
1	Select the group or repeating record from which you want to extract one iteration
2	Click **Promote** on the Main Toolbar. **System Response** This extracts one iteration (instance) of the looping structure. **Note** The Promote function is available only if you select a group or repeating record.

Outbound Example

In this example, your application file definition contains:
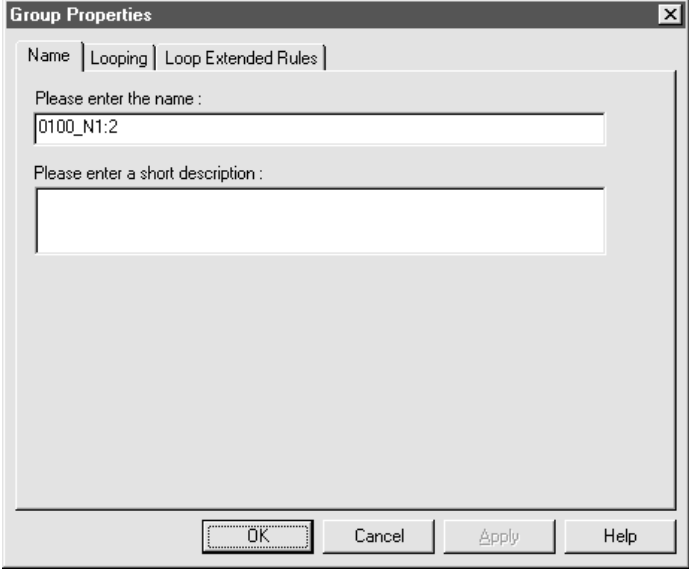
◗ a Ship To record that groups all the shipping information, and

◗ a Bill To record that groups all the billing information.

Your Ship To and Bill To records both occur only once in the application. The EDI standard file contains a N1 group (for ANSI) or a NAD group (for EDIFACT) that repeats 200 times. If the Entity Identifier Code of the N1/NAD is BT, that means the N1/NAD group contains billing information. If the Entity Identifier Code is ST, the N1/NAD group contains shipping information.

Example Procedure

This procedure is an example of how to map unique data to your application file definition.

Step	Action
1	Activate the N1/NAD loop and the appropriate segments and elements, if you have not already done so. If you already activated the N1/NAD group, just highlight it. **Reference** See How to Manually Activate EDI Standard Map Components for more information on activating.
2	Click **Promote** on the Main Toolbar to extract a single instance of the N1/NAD group. *(Continued on next page)*

(Contd) Step	Action
3	From the **Edit** menu, select **Properties**. **System Response** The system displays the Group Properties dialog box with the Name tab displayed. Group Properties Name | Looping | Loop Extended Rules Please enter the name : 0100_N1:2 Please enter a short description : OK Cancel Apply Help
4	In the description box, type a specialized definition to identify this group as containing billing information.
5	Click **OK**. **System Response** The system saves the change and exits the Group Properties dialog box.
6	Right-click the original N1/NAD group (that now repeats 199 times) again and from the Edit menu, select Properties. **System Response** The system displays the Group Properties dialog box with the Name tab displayed.
7	In the description box, type a specialized definition to identify this group as containing billing information. (Continued on next page)

(Contd) Step	Action
8	Click **OK**. **System Response** The system saves the change and exits the Group Properties dialog box.
9	Click the **Deactivate** button on the Main Toolbar and then click the original N1/NAD group (that now repeats 198 times), so the system does not process it.
10	Next, link the elements in the single occurrence of the N1/NAD that contains billing information to the fields in your Bill To application record. Then, link the elements in the single occurrence of the N1/NAD that contains shipping information to the fields in your Ship To application record.

How to Split Groups and Repeating Records

Introduction   The Split function enables you to break a group or repeating record into two loops.

When to use   Use this function when you need more than one instance of the same map component that still occurs multiple times.

Split vs. copy and paste
You can use the Copy and Paste functions (and change the number in the **Maximum usage** box) to accomplish the same task.
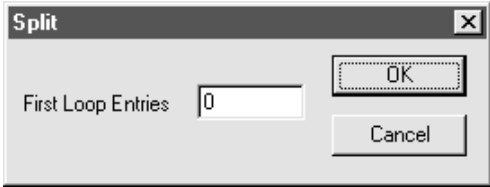
Note
The Split function is available only if you select a group or repeating record.

Procedure   Use this procedure to split a group or repeating record.

Step	Action
1	Highlight the group or repeating record from which you want to extract one iteration.
2	Click **Split** on the Main Toolbar. **System Response** The system displays the Split dialog box. *(Continued on next page)*

(Contd) Step	Action
3	In the **First Loop Entries** box, type the number that indicates the sequential number of iterations where you want the group or repeating record split. **Example** If the X loop repeats a maximum number of 5 times, and you type 2 in the **First Loop Entries** box, the resulting split generates one X loop that repeats a maximum of 2 times and a second X loop that repeats a maximum of 3 times. **Note** The number you enter in the **First Loop Entries** box must be greater than zero and less than the maximum number of iterations of the loop.
4	Click **OK**. **System Response** Exits the Split dialog box.

Outbound Example

In this example, your application file definition contains:

◗ an Allowance record that groups all the allowance information, and

◗ a Charge record that groups all the charge information.
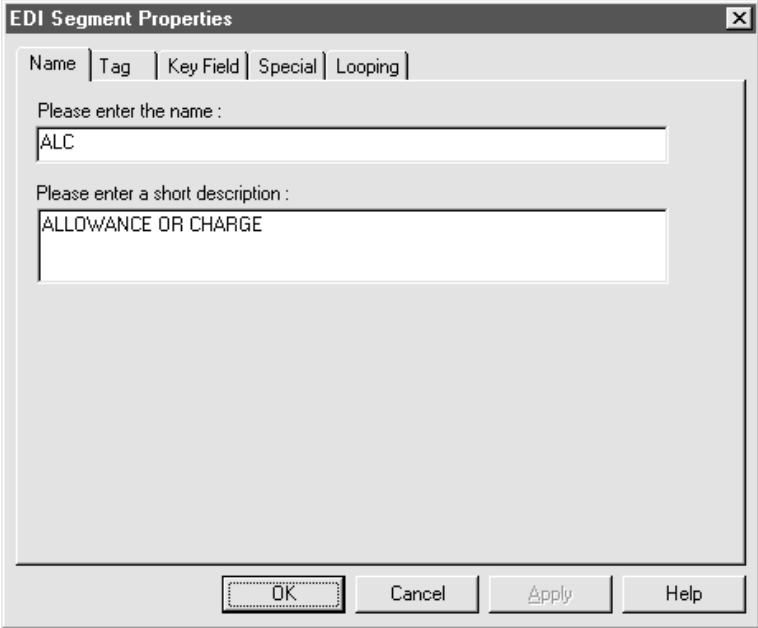
Your Allowance and Charge records can repeat up to five times in the application file definition. The EDI standard file contains a ALC repeating segment that repeats 10 times. If you split the ALC segment in two, you can map allowance information directly to your Allowance application record and map charge information directly to your Charge application record.

Example procedure
This procedure is an example of how to split the ALC segment.

Step	Action
1	Have you activated the ALC segment and the appropriate elements? ◗ If YES, highlight the ALC segment. ◗ If NO, activate them now. **Reference** See How to Manually Activate EDI Standard Map Components for more information on activating components.
2	Click **Split** on the Main Toolbar. **System Response** The system displays the Split dialog box.
3	In the **First Loop Entries** box, type 5. **Note** This is the number of sequential iterations where you want the repeating segment split.
4	Click **OK**. **System Response** Exits the Split dialog box. (Continued on next page)

(Contd) Step	Action
5	From the **Edit** menu, select **Properties**. **System Response** The system displays the EDI Segment Properties dialog box (Name tab).
6	In the description box, type a specialized definition to identify this segment as containing allowance information.
7	Click **OK**. **System Response** The system saves the change and exits the EDI Segment Properties dialog box.
8	Highlight the original ALC segment (that now repeats 5 times).
9	From the **Edit** menu, select **Properties**. **System Response** The system displays the EDI Segment Properties dialog box (Name tab).
10	In the description box, type a specialized definition to identify this segment as containing allowance information. (Continued on next page)

(Contd) Step	Action
11	Click **OK**. **System Response** The system saves the change and exits the EDI Segment Properties dialog box.
12	Next, link the elements in the ALC segment that contains allowance information to the fields in your Allowance application record. Then, link the elements in the ALC segment that contains charge information to the fields in your Charge application record.

Example: Creating a Nested Looping Structure
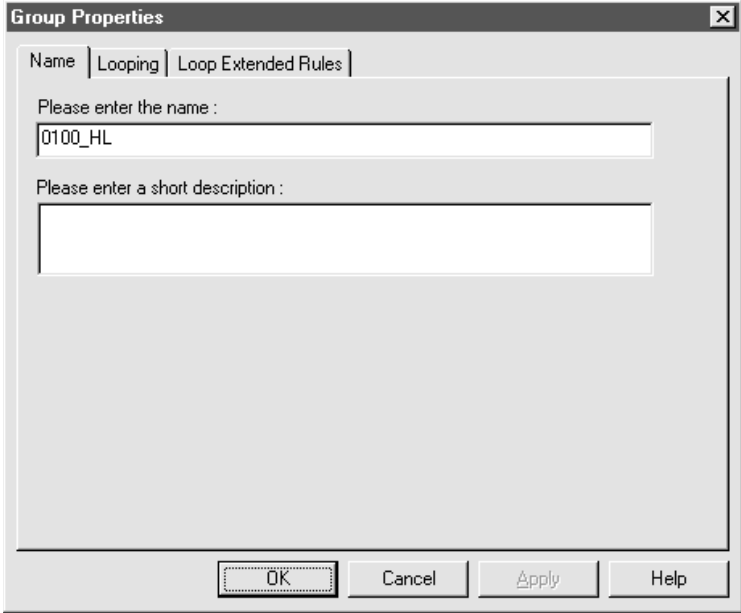
Introduction
In this example, you are defining an ANSI X12 Ship Notice/Manifest (856). When the system generates the 856 document, it contains one HL group. The HL group identifies dependencies between and the content of hierarchically-related groups of segments.

However, for your EDI standard file, you need to define additional levels of detail information. The details that you need to define include shipment, order, and line item information. To modify the structure of your EDI standard file to accommodate these requirements, you need to use the Copy and Paste functions to create a nested looping structure for the HL group.

Example procedure
This procedure is an example of how to create a nested looping structure.

Step	Action
1	Highlight the HL group and click **Copy** on the Main Toolbar.
2	From the **Edit** menu, select **Properties**. **System Response** The system displays the Group Properties dialog box.

(Contd) Step	Action
3	In the second box on the Name tab, type a specialized definition to identify this group as containing shipment information.
4	Click **OK**. **System Response** Saves the change and exits the Group Properties dialog box.
5	Select the map component in the HL that you want the nested HL group (order information) pasted *after* and click **Paste**. **System Response** The system adds a second HL group as a nested loop to the original HL loop.
6	From the **Edit** menu, select **Properties**. **System Response** Displays the Group Properties dialog box for the second HL group.
7	In the second box on the Name tab, type a specialized definition to identify this group as containing shipment information.
8	Click **OK**. **System Response** Saves the change and exits the Group Properties dialog box.
9	Double-click the second HL group (order information) to open it.
10	Select the map component in the HL that you want the nested HL group (item information) pasted *after*, and click **Paste**. **System Response** The system adds a third HL group (item information) as a nested loop of the second HL group.
11	From the **Edit** menu, select **Properties**. **System Response** Displays the Group Properties dialog box for the third HL group.
12	In the second box on the Name tab, type a specialized definition to identify this group as containing shipment information.

(Contd) Step	Action
13	Click **OK**. **System Response** Saves the change and exits the Group Properties dialog box.
14	Now you can activate the appropriate groups, segments, and elements in the new HL nested looping structure. **Reference** See <u>How to Manually Activate EDI Standard Map Components</u> for more information on activating.

How to Make the Two Sides of a Map Equal

Introduction

When you open a map, the system displays the input and output sides of a map in equal dimensions. After manipulating a map and moving the center bar that divides the display of the input and output sides, you may want to make the two sides equal again.

The Equalize function enables you to reinstate the two sides of the map with focus in equal dimensions. The Equalize function is optional.

Procedure

Use this procedure to make the two sides of a map equal in size.

Action
From the **View** menu, select **Equalize**. **System Response** The system moves the center dividing bar between the two sides of the map with focus, so that the two sides of the map are equally proportioned.

How to Map Binary Segments

Introduction

Certain EDI standards use binary segments. Gentran:Server enables you to create and configure map components to process binary data.

When you load a EDI standard that contains binary segments from the Standards CD supplied with Gentran:Server, your system automatically creates the required elements.

If you need to manually create them, use the procedures in this topic.

Example

When you load the X12 275 transaction set into the output side of a map, you will notice that it contains a binary data segment with two elements--one for the length of the binary data and one for binary data. The default data type for the length element is "Number" and the maximum length is set to "15." The default data type for the binary data element is "String" and the default length is "760."

Required segments and elements

To handle binary data received for an inbound EDI segment, you must create or configure regular segments and elements on the input side of your map.

To generate an EDI segment containing binary data, you must create a binary data segment on the output side of your map. Then, you must create two elements in the binary segment for these data types:

◗ Bin_Len, which is for the length (in characters) of the binary segment

◗ Bin_Data, which is for the binary data

This table summarizes the input and output elements of your map for binary data

Input Data Type	Link Status	Output Data Type
Number	Not Linked	Bin_Len
String	Linked	Bin_Data

Setting up the Input side

Use this procedure to create the components on the input side of the map.

Step	Action
1	On the input side of the map, create or select a segment for the data.
2	Right-click the component and select **Properties** from the shortcut menu. **System Response** The system displays the Properties dialog box for the component.
3	Set the component's properties on the tabs of the dialog box. You must complete the Name tab and Looping tab.
4	Add a new element or field to the component you just created.
5	Right-click the element or field and select **Properties** from the shortcut menu. **System Response** The system displays the Properties dialog box for the component.
6	Set the component's properties on the tabs of the dialog box. On the Validation tab, you must specify **Number** as the data type.
7	Add a second new element or field to the segment or record.
8	Right-click the component and select **Properties** from the shortcut menu.
9	Set the component's properties on the tabs of the dialog box. On the Validation tab, you must specify **String** as the data type.

Setting up the output side

Use this procedure to create the components on the output side of the map.

Step	Action
1	On the output side of the map, create or select a segment or record for the binary data.
2	Right-click the component and select **Properties** from the shortcut menu. **System Response** The system displays the Properties dialog box for the component.
3	Set the component's properties on the tabs of the dialog box. On the Special tab, click the **Binary** check box.
4	Add a new element or field to the binary component or select the existing element or field.
5	Right-click the component and select **Properties** from the shortcut menu. **System Response** The system displays the Properties dialog box.
6	Set the component's properties on the tabs of the dialog box. On the Validation tab, you must specify **Bin_Len** as the data type.
7	Add a second new element or field to the segment or select the existing element or field.
8	Right-click the component and select **Properties** from the shortcut menu.
9	Set the component's properties on the tabs of the dialog box. On the Validation tab, you must specify **Bin_data** as the data type.
10	Link the components on the input side of the map to the components on the output side of the map. **Important** You must link the string field on the input side of the map (the field created to contain the file name of the binary data file) to the bin_data field on the output side of the map.
11	Complete the map and compile it to create the translation object.

How Gentran:Server Handles Inbound Binary Data

Introduction   This topic describes how Gentran:Server handles inbound binary data.

In this illustration, the system is configured to insert the binary data into the output file. You can also configure Gentran:Server to simply pass the file name string to the output file.

Inbound binary data   This diagram illustrates how Gentran:Server processes inbound binary data.

Inbound Binary Data
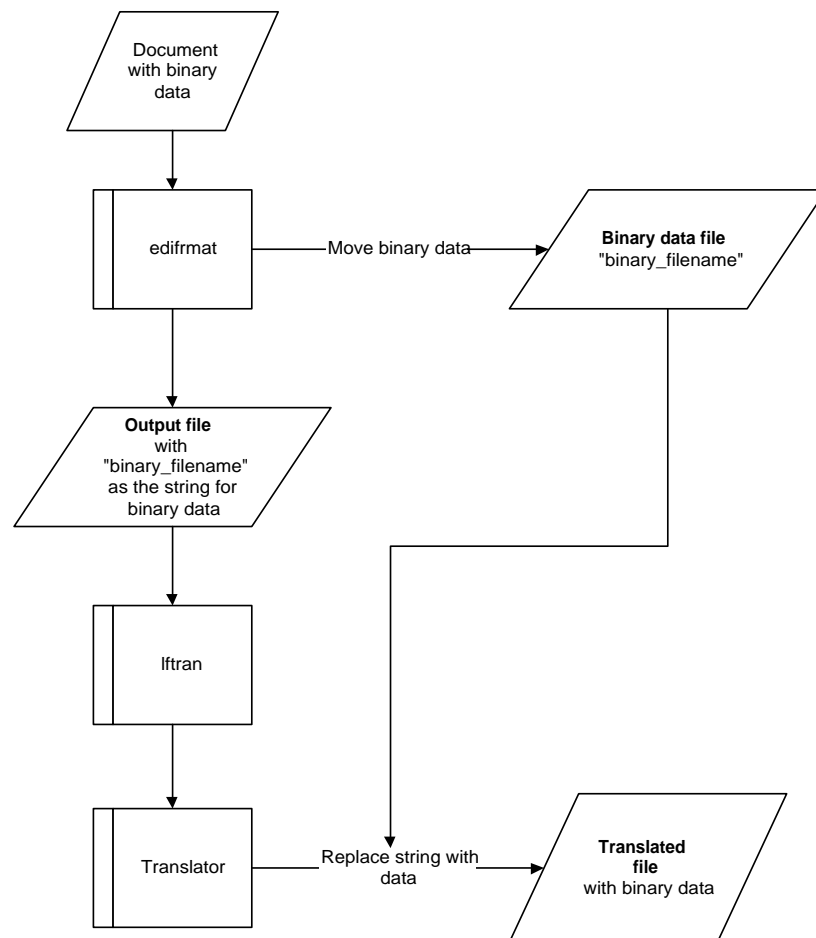
Stages in the inbound binary process

This table describes how the system processes binary data received as input.

Stage	Description
1	Gentran:Server recognizes that the input file contains binary data.
2	The **edifrmat** program extracts the binary data and places it in a temporary file. **Note** The system generates the name for this file and places the file in the wrktmp subdirectory of the directory specified for user files.
3	The **edifrmat** program generates the output file, replacing the binary data with the file name of the temporary file that contains the binary data.
4	The **lftran** program processes the data and passes it to the translator.
5	The translator: ◗ reads the string ◗ locates the file that contains the binary data ◗ inserts the size of the binary data into the Bin_Len element or field position in the output segment/record ◗ can insert the binary data into the Bin_Data element or field position in the output segment/record if you configure the system to do this

How Gentran:Server Handles Outbound Binary Data

Introduction   This topic describes how Gentran:Server handles outbound binary data.

Outbound binary data   This diagram illustrates how Gentran:Server processes outbound binary data.
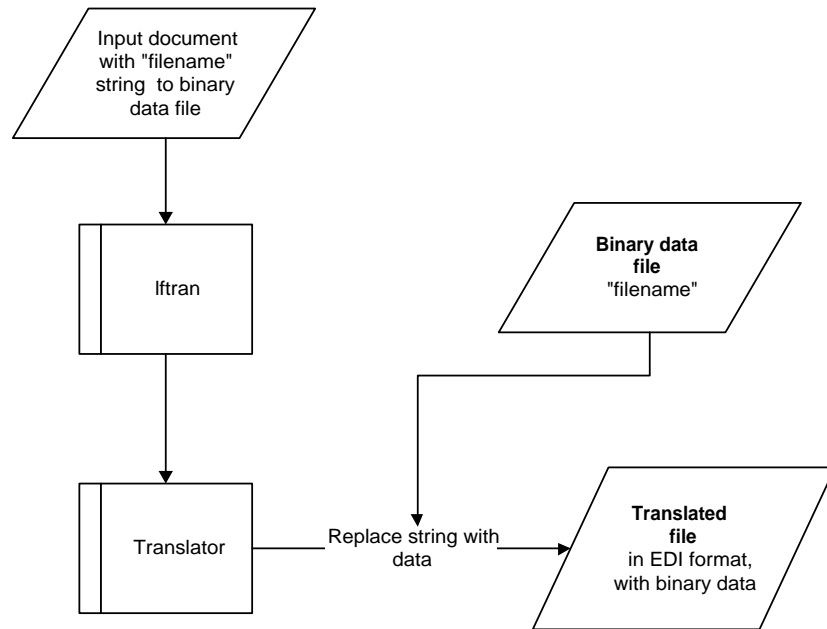
Outbound Binary Data

Stages in the outbound binary process

This table describes how the system processes outbound binary data.

Stage	Description
1	The input document contains a string with the name of the file that contains the binary data. **Notes** You can use a mapping instruction (extended rule) to insert the string into the document. The binary data is in a separate file from the other input data. It can reside in any directory. You must supply the full path to the file.
2	The **lftran** program processes the data and passes it to the translator.
3	The translator: ◗ begins translating the input file ◗ reads the string ◗ locates the file that contains the binary data ◗ inserts the size of the binary data into the Bin_Len element or field position in the output file ◗ inserts the binary data into the Bin_Data element or field position in the output file

Formatting Your Data

Overview

Introduction

When you define or modify a field, you must specify the type and format. The options that are available for the format of the field depend on which type you select:

◗ String

◗ Number

◗ Date/Time

String type

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

Number type

A number type field contains either an implied decimal or real number that can be mathematically manipulated.

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

Date/Time type

A Date/Time type field contains a date or time. If you specify that a field is a Date/Time type, you must specify exactly how the date or time must be formatted.

Syntax Tokens

Definition   A syntax token is a symbol or expression that defines ranges of characters and numbers that are allowed to be used for a string-type field.

Uses   Syntax tokens enable you define ranges of characters and numbers valid for a string-type field. You use syntax tokens in the **Format** box of the Field Properties dialog box. This enables you to define the type of characters that you want the system to use when compliance checking each field.

Default syntax tokens   The system creates default syntax tokens for each map. These default tokens are sufficient for most maps.

Single map   When you set up a token, it applies only to that map. If you want to use the token in another map, you need to set it up for the second map as well.

Character ranges   Gentran:Server uses the ANSI Character Set when determining the start-end range.

For Western European languages, consult the ANSI character chart (1252 Windows Latin 1). This chart also displays ranges so that you can enter appropriate ranges for the characters in your language.

If no chart is available, use the following guidelines:

◗   To include all the accented characters in the major languages of Western Europe, add the following ranges:

Start	End
0xC0	0xD6
0xD8	0xF6
0xF8	0xFC

◗ To include Œ and œ (Scandinavian users), add the following ranges.

Start	End
0x8C	0x8C
0x9C	0x9C
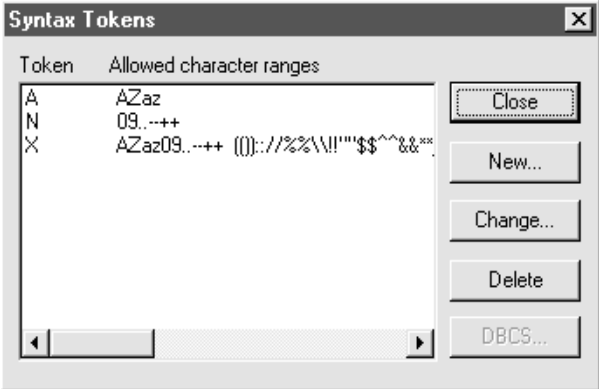
How to Create or Edit a Syntax Token
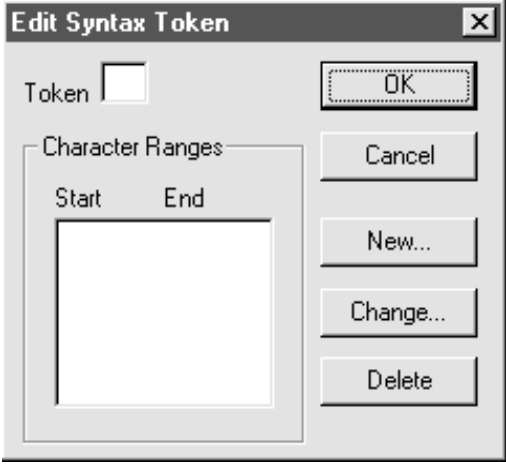
Introduction   This procedure explains how to create a syntax token for Western European languages or edit an existing syntax token.
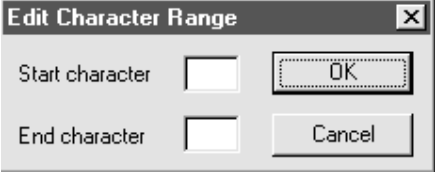
Procedure: creating a syntax token (Western European languages)   Use this procedure to create or edit a syntax token.

Step	Action
1	From the **Edit** menu, select **Syntax Tokens**. **System Response** The system displays the Syntax Tokens dialog box. **Notes** ◗ The Token column contains the value designated as the syntax token for each existing syntax token. The Token contains a range of characters that, when applied to an field, dictate the way that field must be formatted. If the field is not formatted as specified, the system generates a compliance error. ◗ The Allowed character ranges column contains the ranges of characters that are permitted for each existing syntax token. Each range consists of a pair of characters that define the start and end characters. (Continued on next page)
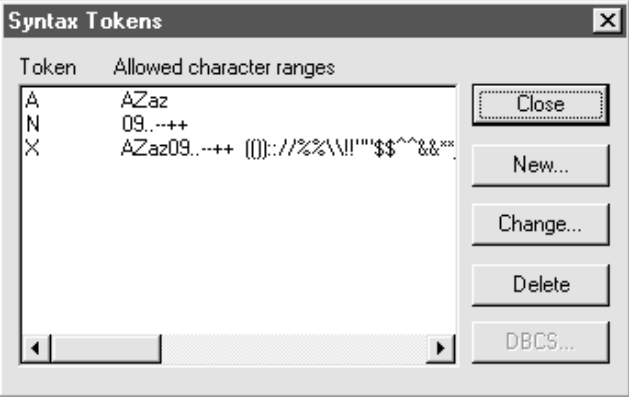
(Contd) Step	Action
2	Are you creating a new syntax token? ◗ If YES, click **New**. ◗ If NO, select the token you want to edit and click **Change**. **System Response** The system displays the Edit Syntax Token dialog box.
3	In the **Token** box, type the unique one-character alphanumeric value that the system recognizes as containing the allowed range of characters you designate. **Notes** ◗ The Token can only be one unique character, upper- or lower-case alphabetic or numeric 1 - 9. ◗ The Character Ranges list contains the character range or ranges that you define for this token. You can define more than one character range for each token. For example, you can define the token A as allowing the character range A through Z *and* the character range a through z. This indicates that token A only allows upper-case *and* lower-case alphabetic characters. (Continued on next page)

(Contd) Step	Action

4	Do you want to create a new character range?
	
	 ◗ If YES, click **New**.
	
	    System Response
	    The system displays the Edit Character Range dialog box.
	
	    [Edit Character Range dialog box showing Start character and End character fields, with OK and Cancel buttons]
	
	 ◗ If NO, continue with Step 8.

5	In the **Start** character box, type the character that begins the allowed token range. **Example** For example, if the character range you want to define is B through D, type B in the **Start** character box. **Notes** ◗ If you type a character, such as é, that is not accepted, you need to enter it in hex code. To enter hex characters, enter 0(zero)x or 0X, followed by the hex code. For example, the hex equivalent of é is 0xE9. ◗ The Start character and End character can only be one (1) character, upper- or lower-case alphabetic or numeric 1 - 9.
6	In the **End Character** box, type the character that terminates the allowed token range. Use the guidelines from Step 5.
7	Click **OK** to return to the Edit Syntax Tokens dialog box.
8	To add additional character ranges to the syntax token, repeat Steps 4 - 7 as many times as necessary.
9	Click **OK** to save the syntax token and return to the Syntax Tokens dialog box.
10	Click **Close** to exit the Syntax Tokens dialog box.

How to Delete a Syntax Token

Introduction   You can delete a syntax token from the system.

Procedure   Use this procedure to delete a syntax token.

Step	Action
1	From the **Edit** menu, select **Syntax Tokens**. **System Response** The system displays the Syntax Tokens dialog box.
2	Select the token that you want to delete and click **Delete**. **System Response** The selected entry is deleted without warning.

How to Delete a Character Range in a Syntax Token

Introduction   You can delete a specific character range from a syntax token.

Procedure   Use this procedure to delete a character range from a syntax token.

Step	Action
1	From the **Edit** menu, select **Syntax Tokens**. **System Response** The system displays the Syntax Tokens dialog box.
2	Select a syntax token and click **Change**. **System Response** The system displays the Edit Syntax Tokens dialog box.
3	Select the character range that you want to delete and click **Delete**.

How to Use Syntax Tokens with a String-Type Field

Introduction   If you specify that a field is a string type, you must designate the format by specifying a syntax token. A syntax token defines ranges of characters or numbers that are valid for a string-type field.

You need to create the syntax token before using it to define a string-type field.

References
See Syntax Tokens for information about these symbols.

See How to Create or Edit a Syntax Token for instructions on creating syntax tokens.

Procedure   Use this procedure to define a string-type field with a syntax token.

Step	Action
1	Double-click an existing field or create a new one. **Reference** See Defining a Fixed-Format Application File for more information. **System Response** The system displays the Field Properties dialog box.
2	Select the Validation tab. **System Response** The system displays the validation options.
3	From the **data-type** list, select **String**. **Note** This indicates that the field contains characters.
4	From the data format list, select a predefined syntax token to denote that this field must be formatted as the specified syntax token dictates.
5	Click **OK** to exit the Field Properties dialog box.

How to Use the Number Type

Introduction

A number type field contains either an implied decimal or real number that can be mathematically manipulated. If you specify that a field is a number type, you must designate the format by specifying a format of either N (implied decimal) or R (real) and the number of decimal places.

Real or numeric

An N formatted number is a number that has an implied decimal point (e.g., 2.01 is formatted as 201). An R formatted number has an explicit decimal point (e.g., 2.01 is formatted as 2.01), and truncates trailing zeroes. Whether you use the N or R format depends on the requirements of your application. Regardless of whether you use the N or R format, you must also indicate the number of decimal places in the field.

This table lists the available real and numeric options.

Option	Decimal point type	Number of places
R0	Explicit	No decimal places
R1	Explicit	Up to one decimal place
R2	Explicit	Up to two decimal places
R3	Explicit	Up to three decimal places
R4	Explicit	Up to four decimal places
R5	Explicit	Up to five decimal places
R6	Explicit	Up to six decimal places
R7	Explicit	Up to seven decimal places
R8	Explicit	Up to eight decimal places
R9	Explicit	Up to nine decimal places
N0	Implied	No decimal places
N1	Implied	Up to one decimal place
N2	Implied	Up to two decimal places
N3	Implied	Up to three decimal places

Option	Decimal point type	Number of places
N4	Implied	Up to four decimal places
N5	Implied	Up to five decimal places
N6	Implied	Up to six decimal places
N7	Implied	Up to seven decimal places
N8	Implied	Up to eight decimal places
N9	Implied	Up to nine decimal places

Note
If you select an implicit decimal (N format) for a field and the data in that field has less than the specified number of decimal places, the translator pads the data with zeroes to the left so that it still interprets the data within the specified format. For example, if you specify a format of N3 for a field, and the data in that field is 1, the translator interprets the data as .001.

Procedure   Use this procedure to use the number type.

Step	Action
1	Double-click an existing field or create a new one. **Reference** See Defining a Fixed-Format Application File for more information. **System Response** The system displays the Field Properties dialog box.
2	Select the Validation tab. **System Response** The system displays the validation options.
3	From the **data-type** list, select **Number**. **Note** This indicates that the field is a numeric or real number that can be mathematically manipulated.
4	From the data format list, select the appropriate real or numeric option.
5	Click **OK** to exit the Field Properties dialog box.

How to Use the Date/Time Type

Introduction   A Date/Time type field contains a date or time. If you specify that a field is a Date/ Time type, you must specify exactly how the date or time must be formatted.

Date/time formats   This table lists the valid date and time formats.

Format	Description
MM/DD/YY	Two-digit month, separator, two-digit day, separator, last two digits of year (example: 12/15/99 for December 15, 1999)
YYYYMMDD	Four-digit year, two-digit month, two-digit day (example: 19991215)
YYYY/MM/DD	Four-digit year, separator, two-digit month, separator, two-digit day (example: 1999/12/15)
DD/MM/YYYY	Two-digit day, separator, two-digit month, separator, four-digit year (example: 15/12/1999)
MM/DD/YYYY	Two-digit month, separator, two-digit day, separator, four-digit year (example: 12/15/1999)
HHMM	Two-digit hour, two-digit minutes (example: 0330 for thirty minutes past three o'clock)
HHMMSS	Two-digit hour, two-digit minutes, two-digits seconds (example: 033045 for thirty minutes and forty-five seconds past three o'clock)
HH:MM	Two-digit hour, separator, two-digit minutes (example: 03:30)
HH:MM:SS	Two-digit hour, separator, two-digit minutes, separator, two-digit seconds (example: 03:30:45)
DDMMYY	Two-digit day, two-digit month, last two-digits of year (example: 151299)
MMDDYY	Two-digit month, two-digit day, last two-digits of year (example: 121599)
DDMMYYYY	Two-digit day, two-digit month, four-digit year (example: 15121999)

(Contd) Format	Description
MMDDYYYY	Two-digit month, two-digit day, four-digit year (example: 12151999)
YYDDD	Last two-digits of year, three-digit Julian day (example: 99349 for the 349th day of 1999)
DDDYY	Three-digit Julian day, last two digits of year (example: 34999)
YYYYDDD	Four-digit year, three-digit Julian day (example: 1999349)
DDDYYYY	Three-digit Julian day, four-digit year (example: 3491999)
YY/DDD	Last two digits of year, separator, three-digit Julian day (example: 99/349)
DDD/YY	Three-digit Julian day, separator, last two digits of year (example: 349/99)
YYYY/DDD	Four-digit year, separator, three-digit Julian day (example: 1999/349)
DDD/YYYY	Three-digit Julian date, separator, four-digit year (example: 349/1999)
MONTH	The month (example: December)
DAY	The day of the week (example: Friday)

Procedure   Use this procedure to use the Date/Time type.

Step	Action
1	Double-click an existing field or create a new one. **Reference** See Defining a Fixed-Format Application File for more information. **System Response** The system displays the Field Properties dialog box.
2	Select the Validation tab. **System Response** The system displays the validation options. (Continued on next page)

(Contd) Step	Action
3	From the **data-type** list, select **Date/Time**. **Note** This indicates that the field is a date or time.
4	From the data format list, select the appropriate date or time option.
5	Click **OK** to exit the Field Properties dialog box.

Working with File Definitions

Overview

Introduction
After you define a file format for the input or output side of your map, you can save it as a separate file so that you can use it in another map. This new file is called a file definition.

File definition name
The system enables you to name the file definition. The file name extension for all file definitions is .DDF.

File format
The system saves file definitions in XML format.

Location
You should always save your file definitions in the drive and folder specified for File Definitions on the Gentran:Server Set Up Directories dialog box. Saving all file definitions in the same folder enables the Application Integration subsystem to list all the available file definitions when you choose to load one into a map.

How to Save a File Definition

Introduction   This topic explains how to save a file format definition in your map as a file definition (.DDF file).
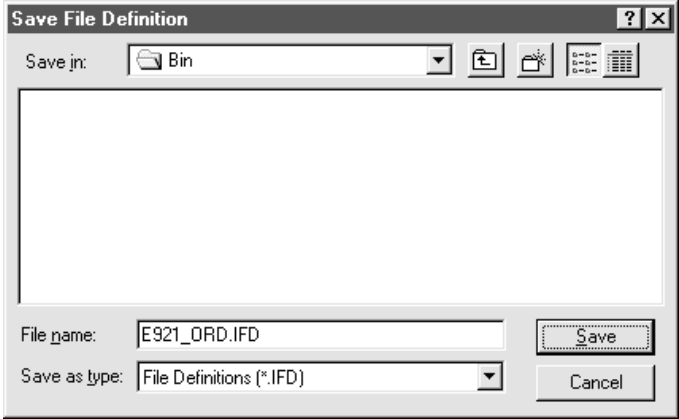
Reference
See Data Definition Format (DDF) for information on Data Definition Format files.

See How to Load a File Definition for information on loading a file definition to use in another map.

Procedure   Use this procedure to save an individual file definition.

Step	Action
1	Right-click the **File Format** icon (either the input or output side of the map) and select **Save File Definition** from the shortcut menu. **System Response** The system displays the Save File Definition dialog box. **Reference** See Data Definition Format (DDF) (.DDF) for information on Data Definition Format files.
2	Does the **Save in** list reflect the drive and folder specified for File Definitions on the Set Up Directories dialog box? ◗ If YES, continue with Step 3. ◗ If NO, select the appropriate folder and continue with Step 3. *(Continued on next page)*

(Contd) Step	Action
3	Type the file name in the **Filename** box or, from the list, select the file definition that you want to save.
4	Click **Save**. **System Response** The system saves the file definition.

How to Load a File Definition

Introduction   Gentran:Server enables you to load a file definition (.DDF) that you previously saved. This feature provides you with a quick way to build either side of your map. You can either load the file definition when you create a new map or replace the input or output side of an existing map with the file definition.

Reference
See How to Save a File Definition for information on saving a file definition.
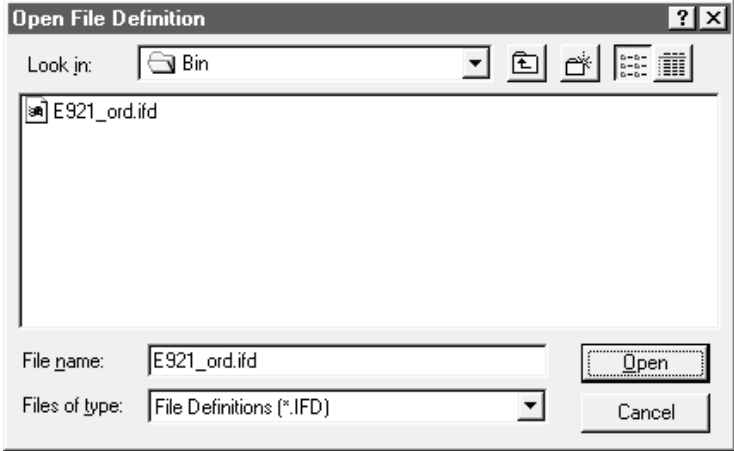
Caution   Loading a file definition replaces the selected side of the map. Please be certain that is your intent before performing this task.

Procedure   Use this procedure to load a file definition.

Step	Action
1	Right-click the **File Format** icon on the side of the map that you want to load and select **Open File Definition** from the shortcut menu. **System Response** <table><tr><td>**IF…**</td><td>**THEN the system displays…**</td></tr><tr><td>you already created that side of the map</td><td>a warning to let you know that the existing file format will be replaced.</td></tr><tr><td>that side of the map has not been defined</td><td>the Open File Definition dialog box</td></tr></table>
2	Did you receive a replacement warning message in Step 1? ◗ If YES, click **Yes** to continue to the Open File Definition dialog box and then continue with Step 3. ◗ If NO, continue with Step 3. (Continued on next page)

(Contd) Step	Action
	Open File Definition dialog box. **Open File Definition** Look in: ☐ Bin 📄 E921_ord.ifd File name: E921_ord.ifd Open Files of type: File Definitions (*.IFD) Cancel
3	Does the **Look in** list reflect the drive or folder where Gentran:Server is installed? ◗ If YES, continue with Step 4. ◗ If NO, select the appropriate folder and continue with Step 4.
4	From the **Files of type** list, select File Definitions (*.DDF). **System Response** The dialog box lists all the file definitions stored in the folder.
5	From the list, select the file definition that you want to load into the system or type the file name in the **Filename** box.
6	Click **Open**. **System Response** The system loads the selected file definition.

Completing a Map

Overview

Introduction	This section explains how to complete your map.
Compiling the map	To complete your map, you must first compile it to create a translation object.
Moving the translation object	If you have Gentran:Server for UNIX, you must move the translation object from your client PC to the UNIX host so that you can use it to translate documents.
Generating a mapping report	The Map Report option enables you to generate information about your map that can help you identify problems.
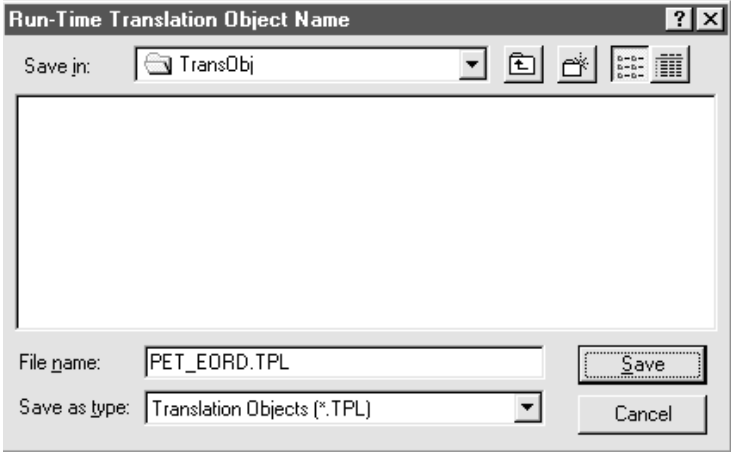
How to Compile a Map

Introduction The Compile function compiles the map and creates a translation object. The map that you create using Gentran:Server is a *source map*. When you compile the source map, the result is a *compiled translation object*. You use the Compile function after you complete and save the map.

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

Step	Action
1	From the **File** menu, select **Compile**.
	System Response
	The system displays the Run-Time Translation Object Name dialog box.
	(Continued on next page)

(Contd) Step	Action
2	In the **File name** box, type the name of the translation object. **Note** The name of the translation object is 64 characters. Use the default .TPL file extension. **Recommendation** Name the translation object the same file name (up to 64 characters in length) as you named the map. Preserving the same file name (with different file extensions) means that the relationship between the source map and the compiled translation object remains evident. **Example** If the source map name is MWT_850.MAP, the compiled translation object name is MWT_850.TPL. **Warning** Be very careful not to overlay the source map with the compiled translation object.
3	From the **Save in** list, change the drive or folder where the compiled translation object is stored, if necessary. **Warning** Do not store the compiled translation object in the directory that holds your source maps.
4	Click **Save**. **System Response** The system compiles the map and generates a translation object. The system displays the Compile Error dialog box.
5	Verify that no errors occurred and click **OK** to exit the dialog box. **Note** The date that the translation object was compiled on is automatically loaded into the **Compiled on** box on the Translation Object Details dialog box.
6	From the **File** menu, select **Save** to save the source map with the Compiled on date.

(Contd) Step	Action
7	Do you want to copy the translation object to the Host User Directory? ▶ If YES, GO TO the <u>How to Copy a Translation Object to the Host</u> topic. ▶ If NO, you are finished.

Reference

For more information about the Map Compiler process, see the <u>Map Compiler Processing</u> topic in the <u>Processes</u> chapter in the *Gentran:Server for UNIX and Workstation Technical Reference Guide.*

How to Mass Compile Multiple Maps

Introduction   In addition to compiling a single map, Gentran:Server allows you to mass compile several maps at a time. This is accomplished by invoking the Application Integration Mapper from the command line with the -c parameter. All the maps to be compiled must be located in the same directory.

Procedure   Use this procedure to compile all maps in a single directory.

Step	Action
1	Open a Command Prompt window.
2	Change to the directory where **mapper.exe** is located.
3	From the command line, invoke the mapper using the following syntax: **mapper.exe** -c <mapdirpath>*.map The path defined in <mapdirpath> must be the full path to the directory where the map files to be compiled are located. **System Response** The system compiles all *source map* (.map) files in the directory specified by <mapdirpath> into *translation object* (.tpl) files with the same name.

Examples   If you want to mass compile all the maps in a directory on drive d: named \gentran\mapsdir, type the following from the command line in Step 3:

mapper.exe -c d:\gentran\mapsdir*.map

All *source map* (.map) files in the *d:\gentran\mapsdir* directory are compiled into *translation object* (.tpl) files with the same name.

Other wildcards can also be used. If you want to mass compile all files with the .map extension whose name ends in z followed by exactly one other character, type the following from the command line:

mapper.exe -c d:\gentran\mapsdir*z?.map

How to Copy a Translation Object to the Host

Introduction   After you compile a new map to create a translation object, you can copy the translation object to the Host user directory for testing with the translator.

Procedure   Use this procedure to copy a translation object to the Host.
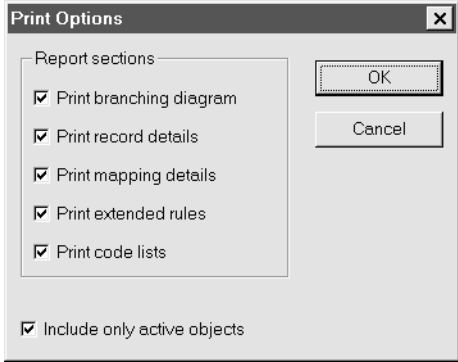
Step	Action
1	Open the Gentran:Server Main window.
2	Select **Copy TPL to User Directory** from the File menu. **System Response** Gentran:Server displays a dialog box for you to select the file you want to copy.
3	Select the translation object (TPL file) and click **Open.** **System Response** Gentran:Server copies the file to the User Directory on the Host.

How to Print a Mapping Report

Introduction   The Print functions enable you to print a mapping report for the current map to a printer or to a file.

Printing to a printer   Use this procedure to print a mapping report.

Step	Action
1	From the **File** menu, select **Print**. **System Response** The system displays the Print Options dialog box
2	Select the reports that you want by clicking the appropriate check boxes.
3	Click **OK.** **System Response** The system displays the Print dialog box.
4	Set the appropriate options and click **OK** if you do *not* need to change Setup options. **System Response** The system prints the mapping report.

Printing to a file Use this procedure to print a mapping report to a file.

Step	Action
1	From the **File** menu, select **Print to File**. **System Response** The system displays the Print Options dialog box. *[Print Options dialog box showing Report sections with checkboxes: Print branching diagram, Print record details, Print mapping details, Print extended rules, Print code lists, and Include only active objects; with OK and Cancel buttons]*
2	Select the reports that you want by clicking the appropriate check boxes.
3	Click **OK.** **System Response** The system displays the Print to File dialog box.
4	Specify the path and name of the file to which you want the report printed and then click **OK**. **System Response** The system prints the mapping report to the file.

Editing Translation Object Details

Overview

Introduction	You can edit the details of the translation object, including the description and version information.
	You can also instruct the translator to use the pad character and alignment settings of each string field when reading an application file, to determine how to trim pad characters from the string data.
	These options are available on the Translation Object Details dialog box.
Using rules	The Translation Object Details dialog box also enables you to specify whether you want the translator to execute standard and extended rules using older behavior (i.e. the **Gentran:Server for Windows NT 2.x Compatible Rule Execution** check box). This option is never used in Gentran:Server for UNIX or Gentran:Server Workstation and should not be checked.
	If an output record or segment is linked to an input field in a record or segment that contains data and is at the same hierarchical level as an input record, the system will execute the standard and extended rules on the output field. If no data is present on all the input fields that are linked to an output record then no rules will run on that record and the record will not be created in the output file.
Rule execution	The system executes standard and extended rules for a record or segment on the output side of the map only if at least one of its links populate the output map component with data.

Translation Object Details Dialog Box

Introduction   The Translation Object Details dialog box enables you to:

◗   Edit the description and version information for the translation object

◗   Instruct the translator to use the pad character and alignment settings of each string field when reading an application file.

Illustration   This illustration shows the Translation Object Details dialog box.
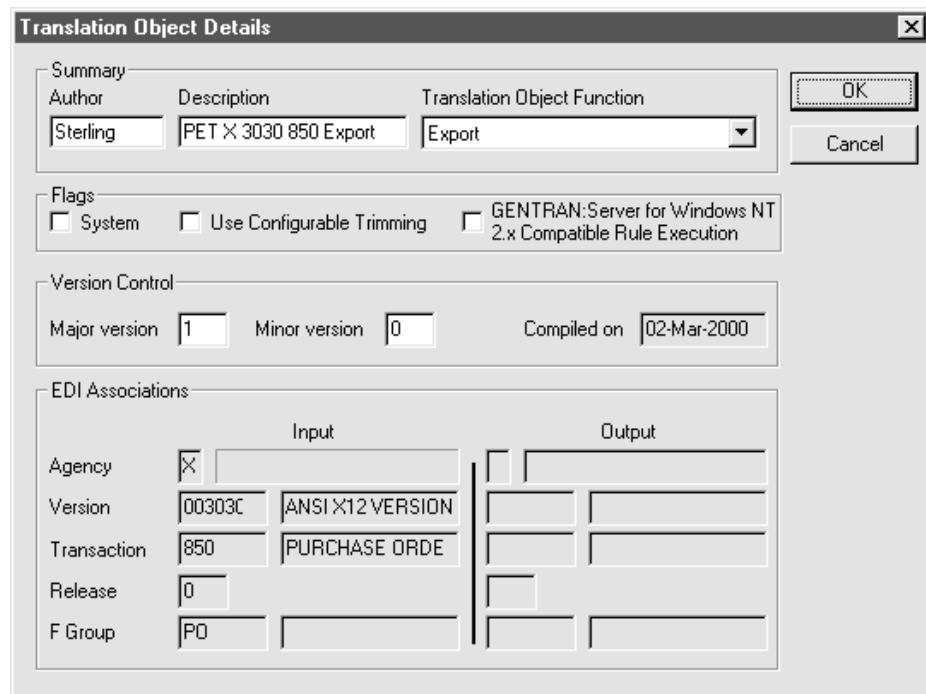
Parts and functions

The following table lists the parts of the Translation Object Details dialog box and their functions.

Part	Function
Author	Identifies the person or department that created the map. **Note** This box is mandatory.
Description	Contains a unique description of the map. **Note** This box is mandatory. The system uses this description to identify the map. We strongly recommend that you use the following identifying characteristics of the map: the partner that this map is used for, the standard, the version, the type of transaction this map uses, and the direction of the map (Application-to-standard, Standard-to-application, etc.). **Example** MWT X 3030 850 Standard-to-application is the description of a map used with partner MWT, for an ANSI X.12 version 003030 Purchase Order (850), that is sent Inbound (Standard-to-application).
Translation Object Function	Displays the type of map. **Note** You cannot change the value in this box.
System	This option is not used in Gentran:Server for UNIX and Gentran:Server Workstation. **Note** Never check this box. (Continued on next page)

(Contd) Part	Function
Use Configurable Trimming	Instructs the translator to use the pad character and alignment settings of each string field when reading an application file, to determine how to trim pad characters from the string data. **Example** If the alignment setting indicates that the data is aligned in the left of the field, then the system trims pad characters from the right of the field. If the alignment setting indicates that the data is aligned at the right of the field, then the system trims pad characters from the left of the field. Finally, if the alignment setting indicates that the field data is aligned in the center of the field, then the system trims pad characters from both the left and right of the field. If this option is cleared (which is the default) the system trims pad characters from both the left and right of each positional string field. This is the current system behavior. **Example** Use configurable trimming to preserve either trailing or leading spaces in your application data.
Gentran:Server for Windows NT 2.x Compatible Rule Execution	This option is not used in Gentran:Server for UNIX or Gentran:Server Workstation. **Note** Never check this box.
Major version/ Minor version	Enables you to assign a different version number to the translation object. **Note** The valid values for each box are 0 - 255. These two boxes are available for notation purposes only. The system does not use them. **Example** If you type 5 in the **Major** box and 45 in the **Minor** box, the Translation Object Version Number is 5.45. (Continued on next page)

(Contd) Part	Function
Compiled on	Displays the date when the translation object was compiled. **Note** This box is blank if the translation object has not been compiled.
EDI Associations	Displays EDI standard header information, such as the standard agency and standard version number. The boxes in this section are blank for translation objects that do not use an EDI standard. **Note** You cannot edit the boxes in this section
OK	Saves changes and exits the dialog box.
Cancel	Exits the dialog box without saving changes.

How to Change Translation Object Details

Procedure   Use this procedure to change translation object details.

Step	Action
1	From the Application Integration **View** menu, select **Details**. **System Response** The system displays the Translation Object Details Dialog Box.
2	Do you want to change the map description? ◗ If YES, type the new description in the **Description** box and continue with the next step. ◗ If NO, continue with the next step.
3	Do you want to change the map version? ◗ If YES, type the appropriate version numbers in the **Major** and **Minor** boxes and continue with the next step. ◗ If NO, continue with the next step.
4	Click **OK**. **System Response** The system saves your changes and exits the Translation Object Details dialog box.

Defining Trading Partnership Rules for Application Files

Overview

Introduction

In this chapter
This chapter explains how to define rules for a file definition that you want Gentran:Server to use to determine the Trading Partnership Code in a data file.

How your system uses these rules
Gentran:Server uses this method to identify the Trading Partnership Code when the file definition does not have a field marked as the Trading Partnership Code. The system also uses it when you run the **lftran** translation command with the A [ ] option.

Key terms
This table lists the key terms used in this chapter.

Term	Description
cross reference table	The table that associates identification strings with specific Trading Partnership Codes.
identification string	The value that results from applying a set of Trading Partnership rules to an application document.
Trading Partnership rule	An instruction that specifies the position of the information that you want Gentran:Server to extract from a particular field in an application document. A set of Trading Partnership rules derives an identification string from the document. Gentran:Server uses this string to determine the Trading Partnership Code

Configuration Process

Introduction   You can configure your system to determine the Trading Partnership Code from multiple fields in the first record of an application file. This topic describes the stages in the configuration process.

Stage table   This table describes the stages in the process.

Stage	Description
1	Create a set of Trading Partnership rules that will yield a unique identification string for a specific file definition (*.ddf). **Reference** See Defining Trading Partnership Rules for instructions.
2	Add the identification string to the TP Cross Reference table and link it to a Trading Partnership Code. **Reference** See Linking Rules to a Trading Partnership Code for instructions.

Defining Trading Partnership Rules

Overview

Introduction
The first stage in configuring your system to determine the Trading Partnership record from multiple fields is to create a set of Trading Partnership rules for a particular file definition.

In this stage, you:

◗ Select the file definition to which you want the rules to apply

◗ Build the rules.

Purpose of Trading Partnership rules
Gentran:Server uses the set of Trading Partnership rules to derive a unique identification string from an application document. Once the system has the identification string, it checks the cross reference table to find the Trading Partnership Code associated with the string.

In this section
This section describes:

◗ Trading Partnership rules

◗ How to start the rule-definition process

◗ How to add a new Trading Partnership rule to a set

◗ How to display a picture that shows the place of each Trading Partnership rule in the identification string

◗ How to edit or delete a Trading Partnership rule in a set.

Trading Partnership Rules

Definition

Trading Partnership rules are instructions that specify the field names and positions in an application file that you want to extract from a document to build a unique identification string.

A Trading Partnership rule specifies the position of the information that you want Gentran:Server to extract from a field in an application document. A rule stipulates the:

◗ Field name

◗ Position of the characters to be extracted

◗ Place in the identification string that the extracted characters will occupy

Guidelines for building rules

These are the guidelines for Trading Partnership rules:

◗ The total number of characters that a set of rules extract to make an identification string cannot exceed 40.

◗ You can associate multiple string combinations to the same Trading Partner Code as long as each combination is unique.

◗ The characters extracted to make strings are case sensitive. For example, AyZ is NOT equal to AYZ.

◗ Leading and trailing spaces are retained. They are placeholders in the string.

◗ Substrings cannot overlap.

TP Rule table

The system stores the rules you define in a DISAM table named apptptbl.dat/idx. Each record in this table contains the:

◗ Name of the file definition

◗ Record ID

◗ Field name and character-identification information for the rule.

Reference
See the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for the layout of the apptptbl.dat/.idx file.

Application TP Rules Dialog Box

Introduction   The Application TP Rules dialog box is used to choose the name of the file definition that describes the layout of the application documents to which the set of Trading Partnership rules apply.

Illustration   This illustration shows the Application TP Rules dialog box.

Fields and functions   This table describes the fields in the Application TP Rules dialog box and their functions.

Field	Function
Select Application/File Definition Filename	Identifies the name of the file definition that describes the records and fields in the application documents that Gentran:Server will process with the set of Trading Partnership rules.
Edit	Displays the Define TP Rules dialog box, which shows the Trading Partnership rules defined for the file definition.
Close	Closes the dialog box.

Add New TP Rule Dialog Box

Introduction The Add New TP Rule dialog box is used to create a new Trading Partnership rule.

Illustration This illustration shows the Add New TP Rule dialog box.

Fields and functions This table describes the fields in the Add New TP Rule dialog box and their functions.

Field	Function
Record ID	Displays the record ID value for the first record. Gentran:Server always uses the fields from the first record for Trading Partnership rules.
Field Name	Specifies the name of the field used for this Trading Partnership rule.
Start Position within Field	Designates the position of the beginning character you want to extract from the field.
From	Indicates the side of the field (left or right) you are counting from for the **Start Position within Field** position.

(Contd) Field	Function
Length	Specifies the number of characters you want to extract.
Marker	Designates the character that you want to represent the part of the string that these extracted characters will occupy.
String Position	Indicates the beginning character number in the string that the characters from this rule will occupy.
OK	Saves the Trading Partnership rule to the Application/TP table.
Cancel	Closes the dialog box without saving changes.
Help	Displays the Help text for this dialog box.

Define TP Rules Dialog Box

Introduction   The Define TP Rules dialog box is used to:

◗   Display an existing set of Trading Partnership rules for a file definition

◗   Access the Add New TP Rule dialog box

◗   Delete a Trading Partnership rule from the set

◗   Modify a Trading Partnership rule

◗   Access the View TP Rule dialog box to display a graphical representation of the Trading Partnership rules in the identification string.

Illustration   This illustration shows the Define TP Rules dialog box.
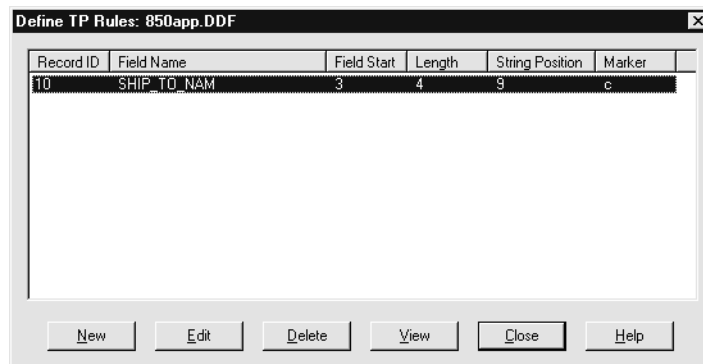
Fields and functions   This table describes the fields in the Define TP Rules dialog box and their functions.

Field	Function
Record ID	Displays the record ID value for the first record. Gentran:Server always uses the fields from the first record for Trading Partnership rules.
Field Name	Shows the name of the field used for this Trading Partnership rule.
Field Start	Shows the position of the beginning character to extract from the field. *(Continued on next page)*
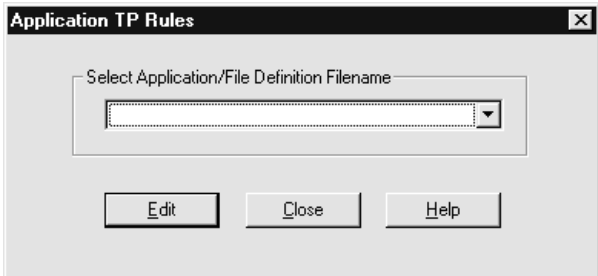
(Contd) Field	Function
Length	Shows the number of characters to extract from this field.
String Position	Shows the beginning character number in the string that the characters from this rule will occupy.
Marker	Shows the character that represents the part of the string that these extracted characters will occupy. This character is used in the View TP Rule dialog box.
New	Displays the Add New TP Rule dialog box, which is used to create a new Trading Partnership rule.
Edit	Displays the Edit TP Rule dialog box for the selected Trading Partnership rule.
Delete	Deletes the selected Trading Partnership rule.
View	Displays a graphical representation of the Trading Partnership rules that comprise a unique string.
Close	Closes the dialog box.
Help	Displays the Help text for this dialog box.

How to Start the Rule-Definition Process

Introduction   This topic describes how to start the rule-definition process by selecting the file definition and opening the Define TP Rules dialog box. You can use this procedure to add, edit, delete, or view a rule.

Procedure   Use this procedure to start the process of defining Trading Partnership rules.

Step	Action
1	Select **Application TP Rules** from the Application File Configuration menu on the Gentran:Server Main window. **System Response** Gentran:Server displays the Application TP Rules dialog box.
2	Click the drop-down arrow in the Select Application/File Definition Filename box to display a list of file definitions; then click the name of the file to which you want the rules to apply. **Note** Only those files checked into the directory for file definitions are displayed in the drop-down list. (Continued on next page)
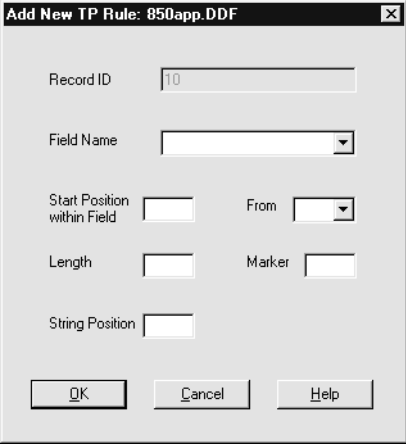
(Contd) Step	Action	
3	Click **Edit**.	
	System Response	
	IF Gentran:Server…	**THEN the system displays the…**
	Finds a record for the file you selected	Define TP Rules dialog box.
	Does not find a record for the file you selected	Add New TP Rule dialog box
4	GO TO one of the following:	
	How to Add a New Trading Partnership Rule	
	How to Display a Picture of the String	
	How to Edit a Trading Partnership Rule	
	How to Delete a Trading Partnership Rule	

How to Add a New Trading Partnership Rule

Introduction   To add a new Trading Partnership rule, you must enter information into the Add New TP Rule dialog box.
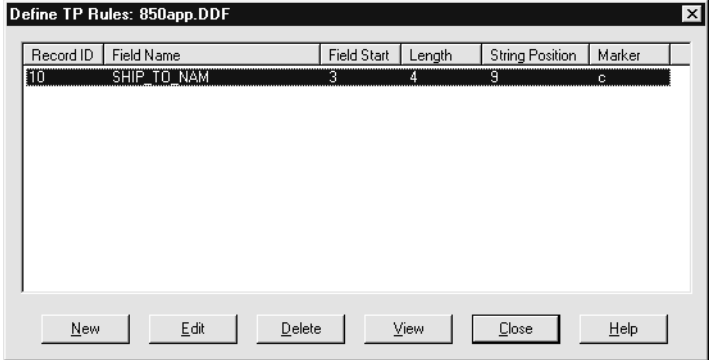
Procedure   Use this procedure to add a new Trading Partnership rule.

Step	Action
1	Start the rule definition process to display the Define TP Rule dialog box. **Reference** See <u>How to Start the Rule-Definition Process</u> for instructions.
2	Click **New** and continue with the next step. **System Response** Gentran:Server displays the Add New TP Rule dialog box.
3	From the drop-down list of the **Field Name** box, select the name of the field from which you want to extract characters for the identification string. **Note** The Record ID field is not available for editing because Gentran:Server always uses the fields in the first record of the application document for Trading Partnership information. (Continued on next page)

(Contd) Step	Action
4	Type the position of the beginning character you want to extract from the field into the **Start Position within Field** box. **Example** If you want to begin with the third character in the field, type 3 in the box.
5	In the **From** box, select left or right to indicate the side of the field you are counting from. **Example** If you want to begin with the third character in the field from the left, select **left** from the drop-down box in the **From** field.
6	Type the number of characters you want to extract into the **Length** field. **Example** If you want to extract four characters beginning with the third character in the field from the left, type 4 into the **Length** field.
7	In the **Marker** field, type the character that you want to represent the part of the string that these extracted characters will occupy. **Note** Gentran:Server uses this character in the View TP Rule dialog box, which graphically depicts the position of each Trading Partnership rule in the string.
8	In the **String Position** box, type the beginning character number in the string that the characters from this rule will occupy. **Example** If you want the four characters from this rule to occupy positions 9, 10, 11, and 12 in the string, type 9 in the **String Position** box. (Continued on next page)

(Contd) Step	Action
9	Click **OK** to save the rule. **System Response** Gentran:Server displays the Define TP Rules dialog box. The new rule is displayed.
10	Select another task.

IF you want to…	THEN…
Add another Trading Partnership rule	Repeat Steps 2 through 9.
View a picture that shows each rule's place in the unique string	GO TO How to Display a Picture of the String.
Modify or delete a Trading Partnership rule	GO TO How to Edit a Trading Partnership Rule.
Add the unique string to the TP cross reference table	GO TO Linking Rules to a Trading Partnership Code.

How to Display a Picture of the String

Introduction   This topic explains how to display a graphical representation of the Trading Partnership rules that comprise a unique identification string. In the picture, Gentran:Server uses the marker specified for each Trading Partnership rule to mark the rule's place in the string.

View TP Rule dialog box   This illustration shows an example of an identification string displayed in the View TP Rule dialog box. The set of markers show that this string is comprised of three Trading Partnership rules.
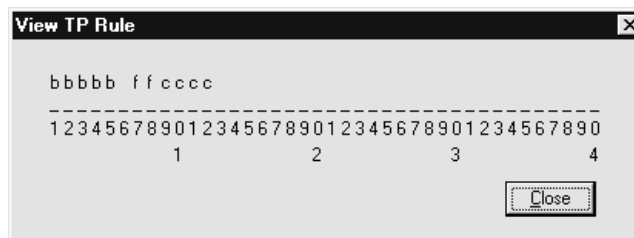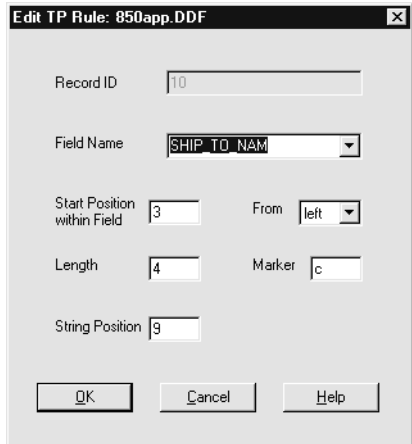
Procedure   Use this procedure to display a graphical representation of the unique string.

Step	Action
1	Open the Define TP Rules dialog box for the file definition. **Reference** See How to Start the Rule-Definition Process for instructions.
2	Click **View** on the Define TP Rules dialog box. **System Response** Gentran:Server displays the View TP Rule dialog box.
3	When you are finished, click **Close** to exit the picture and return to the Define TP Rules dialog box.

How to Edit a Trading Partnership Rule

Introduction   You can modify a Trading Partnership rule from the Define TP Rules dialog box.

Procedure   Use this procedure to modify an existing Trading Partnership rule.

Step	Action
1	Open the Define TP Rules dialog box for the file definition. **Reference** See How to Start the Rule-Definition Process for instructions.
2	Select the Trading Partnership rule that you want to modify.
3	Click **Edit**. **System Response** Gentran:Server displays the Edit TP Rule dialog box.
4	Replace the values you want to edit with new values.
5	Click **OK** to save your changes.

How to Delete a Trading Partnership Rule

Introduction You can delete a Trading Partnership rule from the Define TP Rules dialog box.

Procedure Use this procedure to delete a Trading Partnership rule.

Step	Action
1	Open the Define TP Rules dialog box for the file definition. **Reference** See How to Start the Rule-Definition Process for instructions.
2	Select the Trading Partnership rule that you want to delete.
3	Click **Delete**. **System Response** Gentran:Server displays the prompt: Do you want to delete this record?
4	Confirm that the rule is the one you want to delete.
5	Is this the rule that you want to delete? ◗ If YES, click **Yes** to delete the rule. ◗ If NO, click No to cancel the action and retain the rule. **System Response** Gentran:Server returns to the Define TP Rules dialog box. The rule you deleted has been removed.

Linking Rules to a Trading Partnership Code

Overview

Introduction   To enable Gentran:Server to determine a Trading Partnership Code from the identification string it extracts from an application file, you build a table that cross-references the string with a Trading Partnership Code.

Guidelines   These are the guidelines for linking strings to Trading Partnership Codes:

◗   Each identification string built from applying Trading Partnership rules must be unique. You cannot have the same combination of strings point to different Trading Partnership Codes.

◗   You can link two or more sets of strings to one Trading Partnership Code as long as each string is unique.

◗   Identification strings are case-sensitive. "AbC" does NOT equal "ABC."

◗   Leading and trailing spaces are place holders in the string.

Application TP Cross Reference Dialog Box

Introduction   The Application TP Cross Reference dialog box is used to choose the name of the file definition that describes the layout of the application documents to which the set of Trading Partnership rules apply.

Illustration   This illustration shows the Application TP Cross Reference dialog box.

Fields and functions   This table describes the fields in the Application TP Cross Reference dialog box and their functions.

Field	Function
Select Application/File Definition Filename	Identifies the name of the file definition that describes the records and fields in the application documents that Gentran:Server will process with the set of Trading Partnership rules.
Edit	Displays the Define TP Cross Reference dialog box, which shows the string value and corresponding TP Code.
Close	Closes the dialog box.

Define TP Cross Reference Dialog Box

Introduction   The Define TP Cross Reference dialog box is used to

◗ Display the cross references in the Application TP Cross Reference table

◗ Access the Add New String and TP Code Cross Reference dialog box

◗ Modify a cross reference

◗ Delete a cross reference from the table

Illustration   This illustration shows the Define TP Cross Reference dialog box.
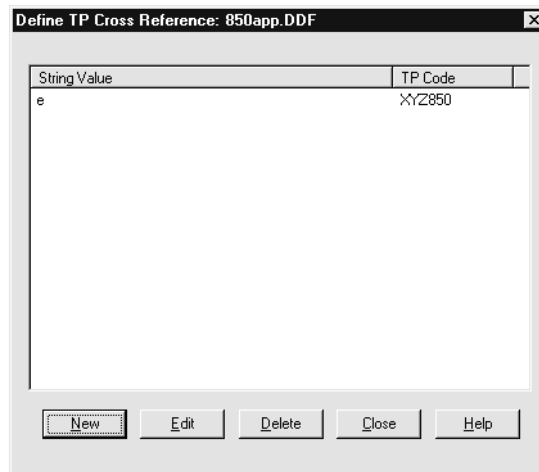
Fields and functions   This table describes the fields in the Define TP Cross Reference dialog box and their functions.

Field	Function
String Value	Indicates the identification string that results from applying a set of Trading Partnership rules to an application document.
TP Code	Indicates the Trading Partnership Code that the system uses when it processes application documents that yield the identification string in the String Value field.

(Contd) Field	Function
New	Displays the Add New String and TP Code Cross Reference dialog box, which is used to create a new cross reference.
Edit	Displays the Edit String and TP Code Cross Reference dialog box for the selected cross reference.
Delete	Deletes the selected cross reference.
Close	Closes the dialog box.
Help	Displays the Help text for this dialog box.

Add New String and TP Code Cross Reference Dialog Box

Introduction The Add New String and TP Code Cross Reference dialog box is used to create a new cross reference for a string and Trading Partnership Code and add it to the Application TP Cross Reference table.

Illustration This illustration shows the Add New String and TP Code Cross Reference dialog box.
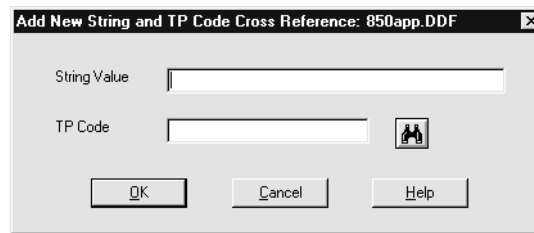
Fields and functions This table describes the fields in the Add New String and TP Code Cross Reference dialog box and their functions.

Field	Function
String Value	Specifies the identification string that results from applying a set of Trading Partnership rules to an application document.
TP Code	Specifies the Trading Partnership Code that you want the system to use when it processes application documents that yield the specified identification string.
OK	Saves the String Value and TP Code set to the Application/TP cross reference table.
Cancel	Closes the dialog box without saving changes.
Help	Displays the Help text for this dialog box.

The Application TP Cross Reference Table

Introduction

Gentran:Server stores the information from the TP Cross Reference dialog box in the Application TP Cross Reference table.

Application TP Cross Reference table

The Application TP Cross Reference table links the string values built from the Trading Partnership rules to Trading Partnership Codes.

The Application TP Cross Reference table is a DISAM file named appxref.dat/idx.
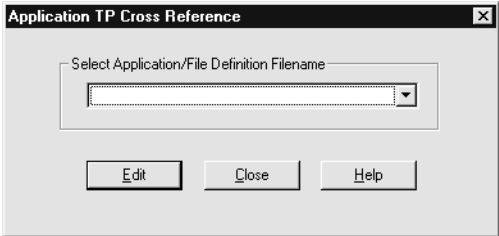
Reference
See the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for the layout of the appxref table.
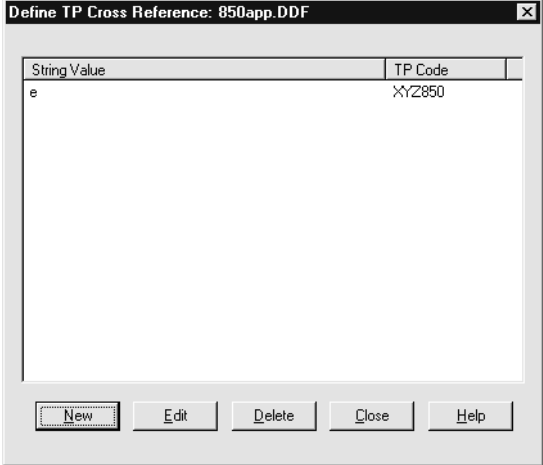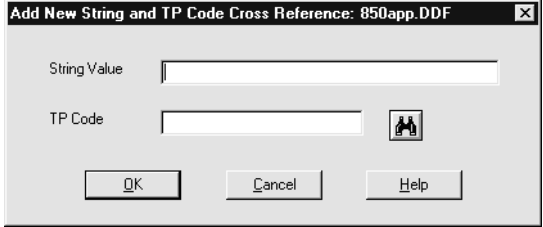
How to Build the Cross Reference Table

Introduction   To add an entry to the Application TP Cross Reference table, you enter the unique string value and the Trading Partnership Code that you want to associate with the string into the Add New String and TP Code Cross Reference dialog box.

Procedure   Use this procedure to add entries to the Application TP Cross Reference table.

Step	Action
1	Select **Application TP Cross Reference** from the Application File Configuration menu of the Gentran:Server Main window.
	System Response Gentran:Server displays the Application TP Cross Reference dialog box.
	Application TP Cross Reference ┌─Select Application/File Definition Filename─────────┐ │ [▼] │ └─────────────────────────────────────┘ [Edit] [Close] [Help]
2	Click the drop-down arrow in the Select Application/File Definition Filename box to display a list of file definitions; then click the name of the file to which the Trading Partnership rules apply.
	Note Only files checked into the directory that holds your file definitions are displayed in the drop-down list.

(Contd) Step	Action
3	Click **Edit**. **System Response** Gentran:Server displays the Define TP Cross Reference dialog box. *Define TP Cross Reference: 850app.DDF* String Value — e TP Code — XYZ850 New Edit Delete Close Help
4	Click **New**. **System Response** Gentran:Server displays the Add New String and TP Cross Reference dialog box. *Add New String and TP Code Cross Reference: 850app.DDF* String Value TP Code OK Cancel Help
5	In the **String Value** box, type the value that will result when Gentran:Server uses the Trading Partnership rules to extract a string value from an application file. (Continued on next page)

(Contd) Step	Action
6	In the **TP Code** box, type the Trading Partnership Code that you want to associate with the string value. **Tip** To display the Trading Partner Search dialog box, click the search icon (binoculars) next to the TP Code box.
7	Click **OK** to add the cross reference to the Application TP Cross Reference table. **System Response** Gentran:Server displays the new entry in the Define TP Cross Reference dialog box.

How to Sort the Cross References

Introduction

On the Define TP Cross Reference dialog box, you can sort the cross reference entries by the string value or the Trading Partnership Code, in either ascending order or descending order.

Sorting toggle switches

This illustration shows the location of the sorting toggle switches.
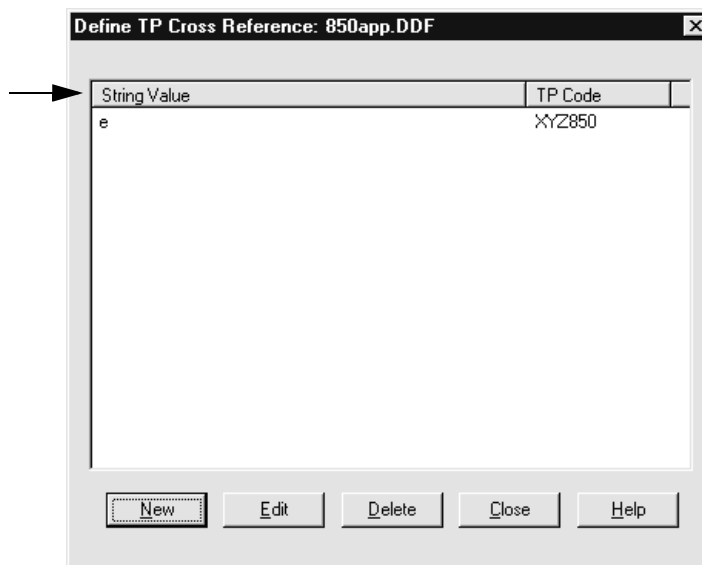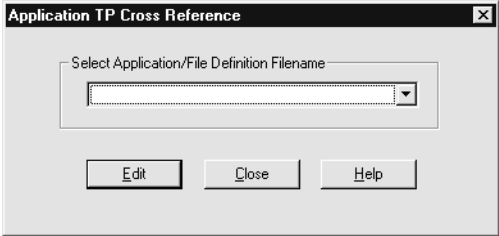
Sorting instructions
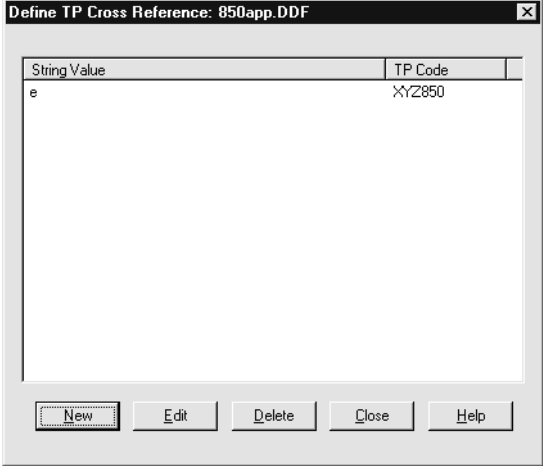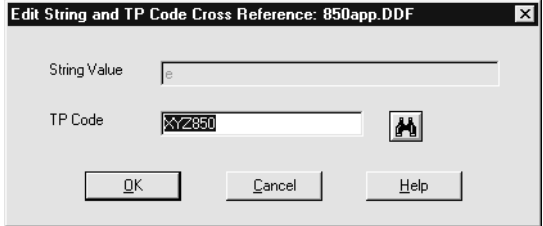
To change the sorting order of the cross-reference entries, click the String Value or Trading Partnership Code toggle switch once.

How to Edit a Cross Reference

Introduction   To edit an entry in the Application TP Cross Reference table, you open the cross reference from the Define TP Cross Reference dialog box and then change the Trading Partnership Code that is associated with the string.

Procedure   Use this procedure to edit a cross references.
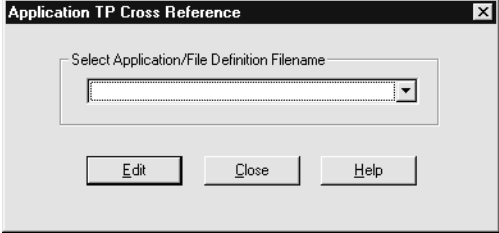
Step	Action
1	Select **Application TP Cross Reference** from the Application File Configuration menu of the Gentran:Server Main window. **System Response** Gentran:Server displays the Application TP Cross Reference Filename dialog box.
2	Click the drop-down arrow in the Select Application/File Definition Filename box and then click the name of the file definition associated with the cross reference you want to edit. (Continued on next page)
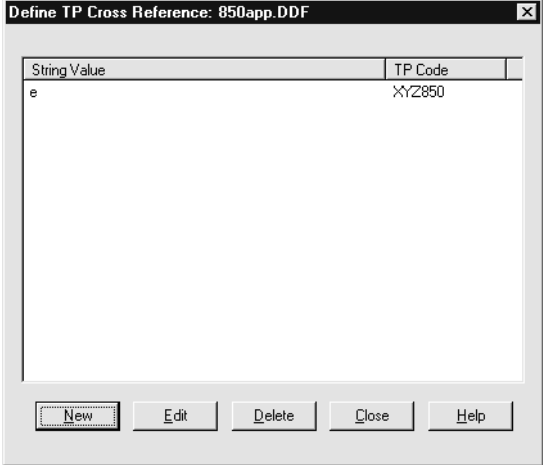
(Contd) Step	Action
3	Click **Edit**. **System Response** Gentran:Server displays the Define TP Cross Reference dialog box.
4	Select the cross reference that you want to edit and then click **Edit**. **System Response** Gentran:Server displays the Edit String and TP Code Cross Reference dialog box.
5	In the **TP Code** box, type another Trading Partnership Code or use the search button to choose another Trading Partnership Code.
6	Click **OK** to save your changes. **System Response** Gentran:Server updates the Application TP Cross Reference table (appxref) and displays the modified entry in the Define TP Cross Reference dialog box.

How to Delete a Cross Reference Entry

Procedure   Use this procedure to delete an entry from the Application TP Cross Reference table.

Step	Action
1	Select **Application TP Cross Reference** from the Application File Configuration menu of the Gentran:Server Main window. **System Response** Gentran:Server displays the Application TP Cross Reference dialog box.
2	Click the drop-down arrow in the Select Application/File Definition Filename box; then click the name of the file definition associated with the cross reference. *(Continued on next page)*

(Contd) Step	Action
3	Click **Edit**. **System Response** Gentran:Server displays the Define TP Cross Reference dialog box.
4	Select the cross reference that you want to delete.
5	Click **Delete**. **System Response** Gentran:Server displays a confirmation prompt.
6	Is this the cross reference that you want to delete? ◗ If YES, click **Yes** to delete the cross reference. ◗ If NO, click **No** to cancel the action and retain the cross reference. **System Response** Gentran:Server deletes the cross reference from the Application TP Cross Reference table (appxref) and from the Define TP Cross Reference dialog box.

How the System Uses the TP Rules

Process   This table explains how the system uses the Trading Partnership rules with an application file.

Stage	Description
1	Gentran:Server locates the file definition that defines the layout of the application document.
2	Gentran:Server determines that the file definition does not have a field defined as the Trading Partnership Code.
3	Gentran:Server reads the application document and uses the Trading Partnership rules to determine the string value.
4	Gentran:Server compares the string value to entries in the Application TP Cross Reference table to find the 15-character Trading Partnership Code associated with the string value.
5	Gentran:Server locates the Trading Partnership record and uses the information it contains to process the application document.

Using Standard Rules

Overview

Introduction

In this chapter   This chapter defines standard rules and explains how to use them.

Key terms   This table lists the key terms used in this chapter.

Term	Description
code list	A roster of code list tables that have been created or loaded in Gentran:Server.
code list table	A table that contains a list of valid codes for an element in an EDI standard.
standard rule	An Application Integration feature that enables you to apply basic or standard mapping operations to fields.

Standard Rules

Definition

A standard rule is a feature that enables you to apply basic or standard mapping operations to fields. Standard rules give you access to functions that are more complex than simple linking, but less involved than extended rules.

Types of standard rules

This table describes the various types of standard rules.

Type	Description
Select	Selects information from another source to one or more fields in the data.
Update	Updates the value of a field that was already read from the input file.
System Variable	Sets a variable that maps the current date and time to the selected element or field in your map.
Constant	Moves a literal constant value to a specified element or field to indicate a qualifying relationship with another element or field.
Use Accumulator	Performs numeric operations or calculations on the selected element or field. Enables you to map accumulated totals to count the occurrences of a specific field.
Loop Count	Counts the number of times a loop is repeated. Enables you to track the current loop or the outer loop in nested loops.
Use Code	Validates an element or field against a code list table to determine if the map component contains one of the values in the table.

One per field

Each of the standard rules are mutually exclusive. Therefore, you can use only one on a particular field.

Related topics

Simple links enable you to match an element or field on the input side of your map with an element or field on the output side.

Extended rules enable you to use a Gentran:Server proprietary programming language to perform virtually any mapping operation you require in the translation process.

References

◗   See <u>How to Compile a Map</u> for information on linking.

◗   See <u>Using Extended Rules</u> for information about extended rules.

Standard Rule Tab

Introduction

You set standard rules on the Standard Rule tab of the Field Properties dialog box.

Rule selection box

This illustration shows the rule selection box on the Standard Rule tab. The text and options on the remainder of the tab depend on the type of standard rule you select from the drop-down list.

References
For descriptions of the parts and functions, see:

[Standard Rule Tab (Select)](#)

[Standard Rule Tab (Update)](#)

[Standard Rule Tab (System Variable)](#)

[Standard Rule Tab (Use Constant)](#)

[Standard Rule Tab (Loop Count)](#)

[Standard Rule Tab (Use Accumulator)](#)

[Standard Rule Tab (Use Code List)](#)

Using the Select Standard Rule

Overview

Introduction

The Select function enables you to select entries from a parent group. 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.

Standard Rule Tab (Select)

Illustration   This diagram illustrates the Standard Rule tab with the Select function selected.

Parts and functions   This table describes the parts of the Select standard rule and their functions.

Part	Function
Please choose the table and key (or group) to use when selecting data	Specifies the table and key the system uses to select data. This table lists the valid values.
Raise compliance error if matching data is not found	Specifies whether you want the system to generate an error if the Select does not find a valid selection. **Note** The default is cleared (do not generate an error if the Select is not successful). (Continued on next page)

(Contd) Part	Function
Map from	Specifies the field from the specified table entry from which you want to map the contents. **Note** Entries are displayed in the map from list *only* after you select a table and key.
Map to	Specifies the field to which you want to map the contents of the map from box. **Notes** ◗ Each field from the side of the map with which you are currently working (input or output) is displayed in this list. ◗ A total of eight fields can be mapped using one Select rule. ◗ Entries are displayed in the map to list *only* after you select a table and key, and these entries are activated *only* after you highlight an entry in the map from box.

How to Map to a Synonym Table

Procedure   Use this procedure to map to a synonym table.

Step	Action
1	Double-click an existing field or element, or create a new one. **Reference** See <u>Defining a Fixed-Format Application File</u> or <u>Defining a Standard File Format or Variable-Length Application File</u> for more information on creating fields or elements. **System Response** The system displays the Field Properties or Element Properties dialog box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Select**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?"
4	Click **Yes** to continue.
5	Select **Synonym Table by In Value** from the table and key (or group) list.
6	Enter the appropriate sub-table name in the specified box.
7	Do you want the system to generate an error if the Select does not find a valid selection? ◗ If YES, check the **Raise a compliance error if matching data is not found** check box. ◗ If NO, ensure the **Raise a compliance error if matching data is not found** box is cleared. This is the default condition.
8	Select the field in the left field list box. This box lists the available Synonym Table fields. There will only be one field listed.
9	In the right field/element list box, select the field to which you want to map the contents of the selected Synonym Table field. (Continued on next page)

(Contd) Step	Action
10	Repeat Steps 8 and 9 as necessary to map more data.
11	Click **OK** to exit the dialog box.

Note
This standard rule is provided to support existing customers still using Visual Mapper synonym tables. For more information about the synonym tables used with the Visual Mapper, see the Visual Mapper Processes topic in the Processes chapter in the *Gentran:Server for UNIX and Workstation Technical Reference Guide*.

Using the Update Standard Rule

Overview

Introduction   The Update function enables you to update a specific field on the current side of the map with the contents of the field.

Standard Rule Tab (Update)

Illustration   This diagram illustrates the Standard Rule tab with the Update function selected.
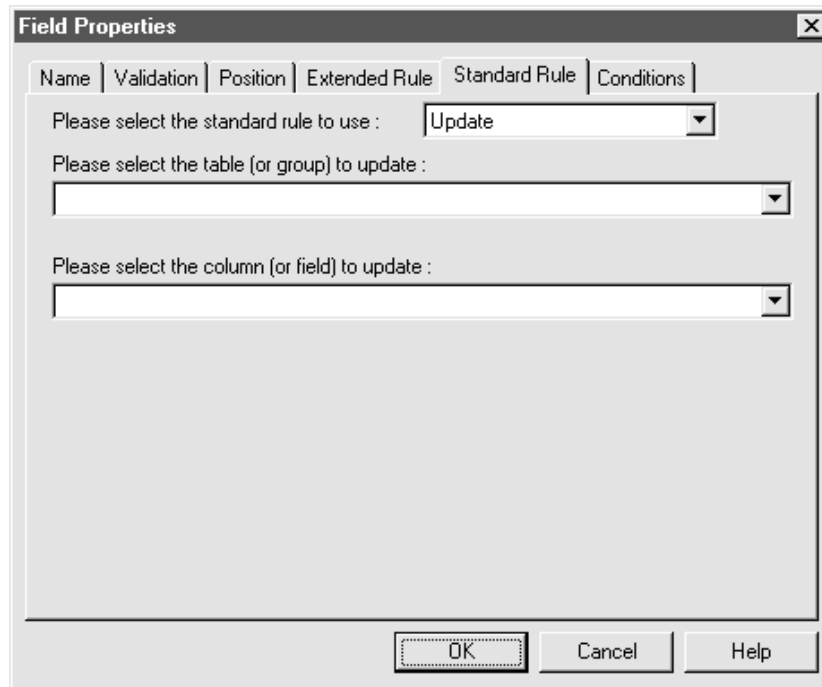
Parts and functions   This table describes the parts of the Update standard rule and their functions.

Part	Function
Please select the table (or group) to update	Contains the group that the system updates with the contents of the current field.
Please select the column (or field) to update	Contains the field that the system updates with the contents of the current field. **Note** The fields that are available in this list depend on which side of the map is selected.

Using the System Variable Standard Rule

Overview

Introduction   The Use System Variable function enables you to set a variable that maps the current date and time to the selected field. The selected map component must have a data type of Date/Time.

Standard Rule Tab (System Variable)

Illustration   This illustration shows the Standard Rule tab with the System Variable function selected.
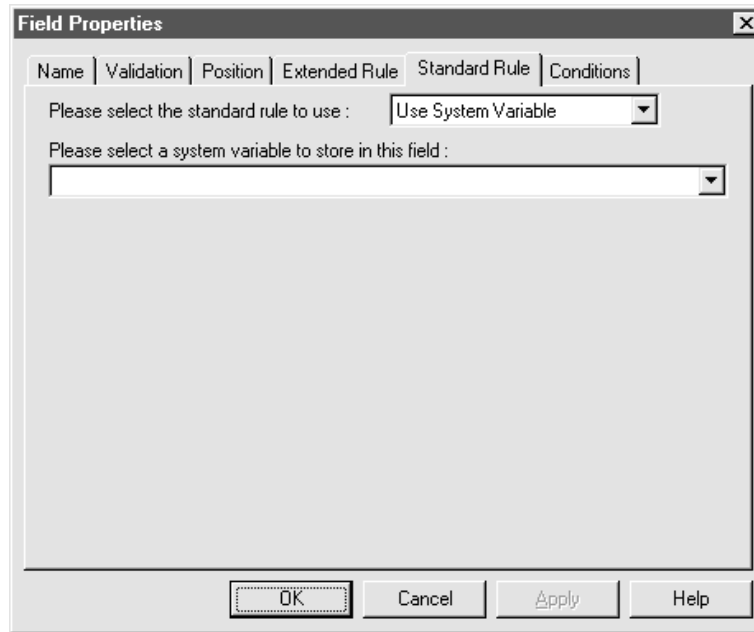
Parts and functions   This table describes the parts of the Standard Rule tab for the Use System Variable standard rule and their functions.

Part	Function
System variable box	Enables you to map current date and time variables to the field.

How to Use the System Date and Time

Procedure   Use this procedure to use the system date and time.

Step	Action
1	Double-click an existing field or create a new one. **Reference** See Defining a Fixed-Format Application File for more information on creating fields. **System Response** The system displays the Field Properties dialog box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Use System Variable**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
4	Click **Yes** to continue.
5	Select **Current date and time** from the system variable list to map the variables to the field.
6	Click **OK** to set up the system variable.

Using the Use Constant Standard Rule

Overview

Introduction   The Use Constant standard rule enables you to:

◗   Move a literal constant value to a specified field

◗   Generate qualifiers to indicate a qualifying relationship with another field.

Why use literal constants?   The system uses literal constants as repositories to store information that is used later in the map. You can create constants in maps to hold information that will be moved to an output field or used in a conditional statement.

You can also use literal constants to load common data into each application record on a map. Once you create the literal constant and name it, you can "hard code" (enter) a value that is loaded into the field that uses the constant.

To use the information stored in a constant, you need to use an standard rule.

Standard Rule Tab (Use Constant)

Illustration   This diagram illustrates the Standard Rule tab with Use Constant selected.
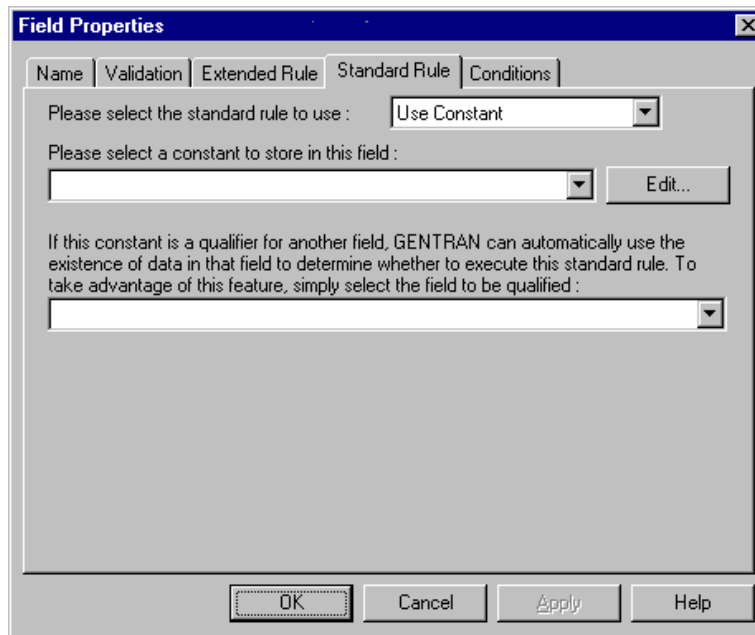
Parts and functions   This table lists the parts of the Standard Rule tab (with the Use Constant standard rule selected) and their functions.

Part	Function
constant	Lists the available constants. The selected constant is mapped to the current field. **Note** If the necessary constant is not in the constant list, you need to create it. Click **Edit** to access the Translation Object Constants dialog box. (Continued on next page)

(Contd) Part	Function
Edit	Accesses the <u>Translation Object Constants Dialog Box</u>, which enables you to create, edit, and delete literal constants.
qualifier	Lists the fields that you can use to qualify the selected map component, establishing a qualifying relationship between the two map components. **Note** If the system does not generate a qualifier field because data is not present, the system does not execute the standard rule that moves the constant to the current field.

Translation Object Constants Dialog Box

Illustration   This diagram illustrates the Translation Object Constants dialog box.
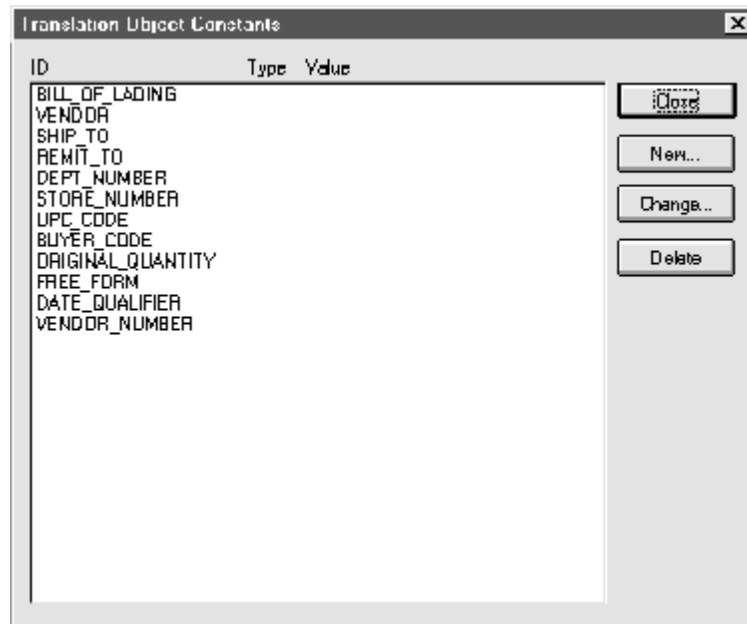
Parts and functions   This table lists the parts of the Translation Object Constants dialog box and their functions.

Part	Function
Constants list	Lists all literal constants currently defined in the system.
Close	Exits the Translation Object Constants dialog box.
New	Accesses the Edit Constant dialog box, which enables you to create a new constant.
Change	Accesses the Edit Constant Dialog Box, which enables you to edit the selected constant.
Delete	Removes the selected constant from the system.

Edit Constant Dialog Box

Illustration   This diagram illustrates the Edit Constant dialog box.
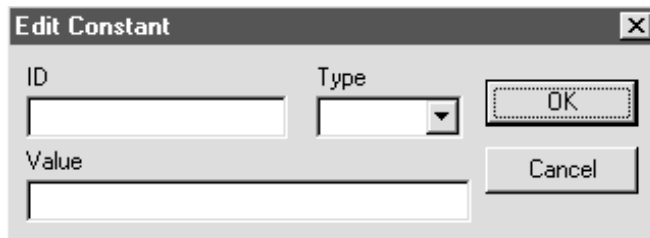
Parts and functions

This table lists the parts of the Edit Constant dialog box and their functions.

Part	Function
ID	Contains the literal constant identifier. **Note** This is usually a description of the field in which the constant is used. If you need to refer to the constant in an extended rule, you should use the data from this field.
Type	Specifies the category of this literal constant. Valid selections are: ◗ **Integer** -- Select for numeric constants that are a positive or negative natural (non-fraction) number or 0 (zero) ◗ **Real** -- Select for numeric constants that are a positive or negative integer with an explicit decimal point ◗ **String** -- Select for alphanumeric constants
Value	Specifies the actual constant expression. **Note** This is the value of the literal constant.

How to Define a Qualifying Relationship

Introduction

The system uses literal constants as repositories to store information that is used at a later point in the map. Typically, you use constants in an outbound map to generate a qualifier.
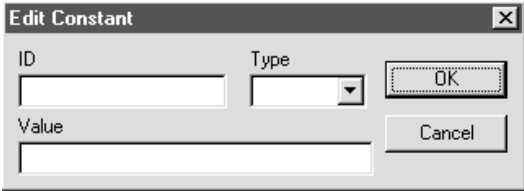
What is a qualifier?

A qualifier is an element that has a value expressed as a code that gives a specific meaning to the function of another element.

What is a qualifying relationship?

A qualifying relationship is the interaction between an element or field and its qualifier. The function of the element changes depending on which code the qualifier contains.

Example procedure

Use this procedure to define a qualifying relationship. In this example, you use a constant to define a qualifier for a Product/Service ID.

Step	Action
1	Double-click the Product/Service ID Qualifier that you want to use to further define (qualify) the Product/Service ID. **System Response** The system displays the Element Properties dialog box (Name tab).
2	Select the **Standard Rule** tab to access the standard rule options.
3	From the standard rule list, select **Use Constant**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
4	Click **Yes** to continue.
5	Click **Edit**. **System Response** The system displays the Translation Object Constants dialog box. (Continued on next page)

(Contd) Step	Action
6	Click **New**. **System Response** The system displays the Edit Constant dialog box. *[Edit Constant dialog box: ID, Type, Value fields with OK and Cancel buttons]*
7	In the **ID** box, type the literal constant identifier. **Note** This is usually a description of the field or element in which the constant is used. If you need to refer to the constant in an extended rule, you should use the data from this box.
8	From the Type list, select **String**. **Note** The Product/Service ID Qualifier is formatted as a string data type. This indicates the category of the literal constant.
9	In the **Value** box, type UI. **Note** This indicates that the Product/Service ID field must contain the UPC Code. This is the value of the literal constant.
10	Click **OK** to add the constant to the system. **System Response** The system displays the Translation Object Constants dialog box, which now includes the new constant.
11	Click **Close** to exit the Translation Object Constants dialog box. **System Response** The system displays the Standard Rules tab.
12	Select the constant from the drop-down list of constants. (Continued on next page)

(Contd) Step	Action
13	Select **Product/Service ID** from the qualifies list. This is the element that the Product/Service ID Qualifier qualifies. **Note** This list contains only the other *active* fields or elements in the same record or segment as the qualifying field.
14	Click **OK** on the Element Properties dialog box to establish the qualifying relationship between the two elements.

How to Define and Edit Constants

Procedure   Use this procedure to create or edit a constant so you can use it to store information.

Step	Action
1	From the **Edit** menu, select **Constants**. **System Response** The system displays the Translation Object Constants dialog box.
2	Do you want to create a new constant? ◗ If YES, click **New**. ◗ If NO (you want to edit an existing constant), select the constant and click **Change**. **System Response** The system displays the Edit Constant dialog box.
3	In the **ID** field, type the literal constant identifier.
4	From the **Type** list, select the category of this literal constant.
5	In the **Value** field, type the actual constant expression.
6	Click **OK** to add the constant to the system.
7	Click **Close** to exit the Translation Object Constants dialog box.

How to Use a Constant in a Map

Procedure   Use this procedure to use a constant in your map.

Step	Action
1	Double-click an existing field or create a new one. **Reference** See Defining a Fixed-Format Application File for more information on creating fields. **System Response** The system displays the Field Properties dialog box (Name tab).
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Use Constant**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?"
4	Click **Yes** to continue. **System Response** The system displays the constant options.
5	Do you need to create or edit a constant to use for the current field? ◗ If YES, click **Edit** to access the Translation Object Constants dialog box. **Reference** See How to Define and Edit Constants for more information on creating and editing constants. ◗ If NO, continue with the next step.
6	From the constants list, select the constant that you want to map to the current field. (Continued on next page)

(Contd) Step	Action
7	Do you want to establish a qualifying relationship between the current map component and another component? ◗ If YES, from the qualifier list, select the field that the system uses to determine whether to execute this standard rule when the qualifying component contains data. ◗ If NO, continue with the next step.
8	Click **OK** to save the standard rule and exit the Properties dialog box.

How to Delete Constants

Procedure   Use this procedure to delete a constant.

Step	Action
1	From the **Edit** menu, select **Constants**. **System Response** The system displays the Translation Object Constants dialog box.
2	Select the constant you want to delete.
3	Click **Delete**. **System Response** The system removes the constant without warning.

How to Generate Qualifiers

Introduction   A qualifier is a field that has a value expressed as a code that gives a specific meaning to the function of another field.

Procedure   Use this procedure to define qualifying relationships.

Step	Action
1	Double-click the field you want to use to further define (qualify) another field. **System Response** The system displays the Field Properties dialog box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Use Constant**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
4	Click **Yes** to continue.
5	From the qualifies list, select the field that this field qualifies. **Note** This list contains only the other *active* fields in the same record or segment as the qualifying field.
6	Click **OK**. **System Response** The system establishes the qualifying relationship between the two fields.

Using the Loop Count Standard Rule

Overview

Introduction

The Loop Count standard rule enables you to count the number of times a loop is repeated, if the field is part of a loop.

If the loop is a nested loop, you can track the current loop *or* the outer loop.

Example
If the Y loop is nested within the X loop, and the Y loop has cycled through 15 iterations and the X loop has cycled through 3 iterations, you can choose to count either the "15" (Y loop) or the "3" (X loop).

Standard Rule Tab (Loop Count)

Illustration   This diagram illustrates the Standard Rule tab with Loop Count selected.
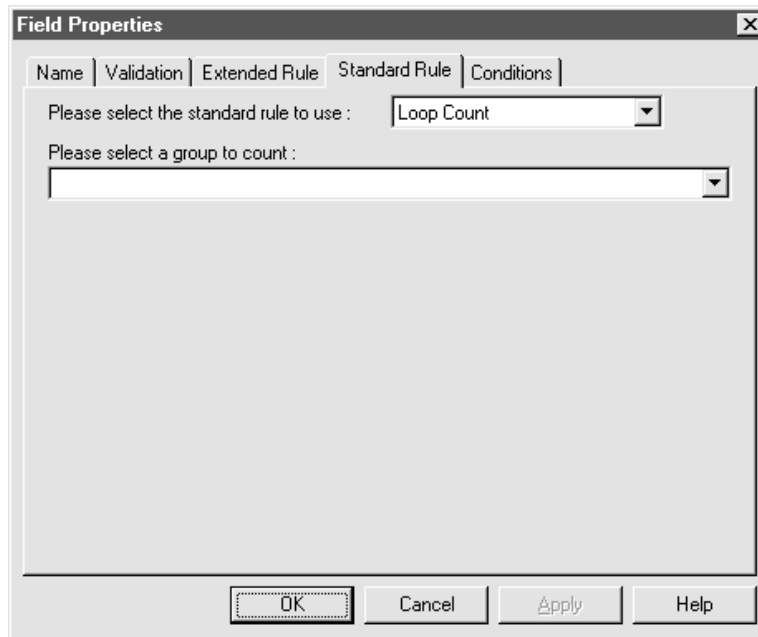
Parts and functions   This table lists the parts of the Standard Rule tab (with the Loop Count standard rule selected) and their functions.

Part	Function
group box	Specifies the loop that you want to count. **Note** If the loop is a nested loop, you can track the current loop *or* the outer loop. **Example** If the Y loop is nested within the X loop, and the Y loop has cycled through 15 iterations and the X loop has cycled through 3 iterations, you can choose to count either the "15" (Y loop) or the "3" (X loop).

How to Use the Loop Count Function

Procedure   Use this procedure to use the Loop Count function.

Step	Action
1	Double-click an existing field or create a new one. **Reference** See Defining a Fixed-Format Application File for more information on creating fields. **System Response** The system displays the Field Properties dialog box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Loop Count**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?"
4	Click **Yes** to continue.
5	Select the loop that you want to count.
6	Click **OK** to add the standard rule.

Using the Use Accumulator Standard Rule

Overview

Introduction The Use Accumulator standard rule gives you access to a set of numeric variables. You can manipulate these variables with numeric operations and then transfer the results to and from fields.

This function enables you to:

◗ Add, change, or delete calculations for the field.

◗ Use hash totals to accumulate numeric field values, such as quantity or price.

◗ Map the accumulated total into a control total field, using accumulators to count the occurrences of a specific field

◗ Map the accumulated total into a control total field, using accumulators to generate increasing or sequential record or line item numbers.

Standard Rule Tab (Use Accumulator)

Illustration   This diagram illustrates the Standard Rule tab with Use Accumulator selected.
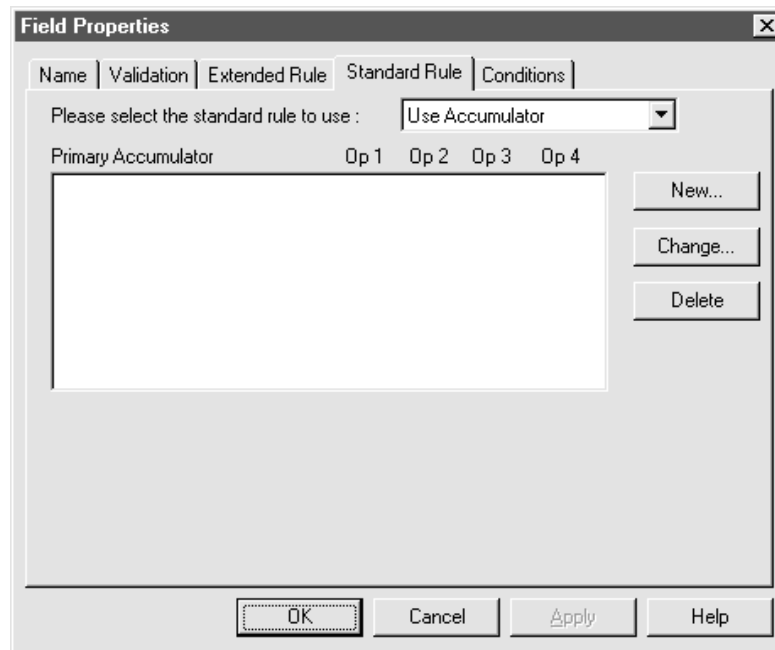
Parts and functions   This table lists the parts of the Standard Rule tab with the Use Accumulator standard rule selected and their functions.

Part	Function
Primary accumulator	Displays all existing calculations that were created for this field.
New	Accesses the Edit Accumulator Entry dialog box, which enables you to create a new calculation for this field. (Continued on next page)

Part	Function
Change	When a calculation in the Primary Accumulator list is selected, this button accesses the Edit Accumulator Entry dialog box, which you can use to edit the selected calculation.
Delete	When a calculation in the Primary Accumulator list is selected, this button deletes the selected calculation. **Warning** The system deletes the selected calculation without warning.

Edit Accumulator Entry Dialog Box

Illustration   This diagram illustrates the Edit Accumulator Entry dialog box.
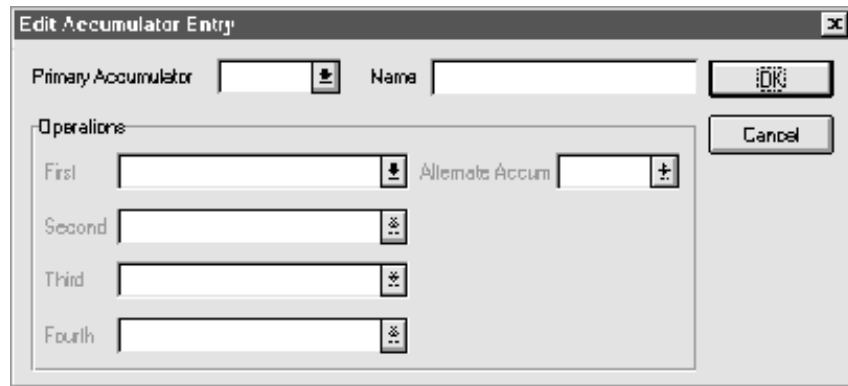
Parts and functions   This table lists the parts of the Edit Accumulator Entry dialog box and their functions.

Part	Function
Primary accumulator	Specifies the primary accumulator.
	Notes
	◗ Before the system performs any calculations on an accumulator, its content is 0 (zero). When you use an accumulator, the system adds a new accumulator to the bottom of this list.
	◗ Each map has only one set of accumulators. 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 is the result of the earlier calculation. The system performs any additional calculations you assign to that accumulator on the contents resulting from an earlier calculation.
	(Continued on next page)

(Contd) Part	Function
Name	Contains a descriptive alias that enables you to differentiate the use of accumulators you create.
First	Specifies the first operation that the system performs. **Notes** ◗ The **First** box is active only after you select a Primary Accumulator. ◗ Before the system performs any calculations on an accumulator, its content is 0 (zero). When you use an accumulator, the system adds a new accumulator to the bottom of this list. **Reference** See Accumulator operations for the valid accumulator operations.
Second	Specifies the second operation that the system performs after it completes the First operation. **Note** The Second box is active only after you select a First operation that does not involve the Alternate Accumulator. **Reference** See Accumulator operations for the valid accumulator operations.
Third	Specifies the third operation that the system performs after it completes the Second operation. **Note** The Third box is active only after you select a Second operation. **Reference** See Accumulator operations for the valid accumulator operations. (Continued on next page)

(Contd) Part	Function
Fourth	Specifies the fourth operation that the system performs after it completes the Third operation. **Note** The Fourth box is active only after you select a Third operation. **Reference** See <u>Accumulator operations</u> for the valid accumulator operations.
Alternate Accum	Specifies another accumulator that can be used in accumulator operations. **Reference** See <u>Accumulator operations</u> for the valid accumulator operations.

Accumulator operations

This table lists the available accumulator operations and their functions.

Part	Function
Increment primary	Adds 1 (one) to the contents of the Primary Accumulator, so that Primary = Primary + 1.
Decrement primary	Subtracts 1 (one) from the contents of the Primary Accumulator, so that Primary = Primary - 1.
Sum in primary	Adds the numeric value (takes the positive or negative sign of the numbers into account) of the field to the contents of the Primary Accumulator, so that Primary = (+/-)Primary + (+/-)Field.
Hash sum in primary	Adds the absolute value (does not take the positive or negative sign of the numbers into account) of the field to the contents of the Primary Accumulator, so that Primary + Field.
Load primary	Loads the contents of the field into the Primary Accumulator, so that Primary = Field.
Use primary	Loads the contents of the Primary Accumulator into the field, so that Field = Primary. *(Continued on next page)*

(Contd) Part	Function
Zero primary	Sets the value of the Primary Accumulator to zero, so that Primary = 0.
Multiply with primary	Multiplies the field with the contents of the Primary Accumulator, and stores the result in the Primary Accumulator, so that Primary = Primary * Field.
Divide by primary	Divides the field with the contents of the Primary Accumulator, and stores the result in the Primary Accumulator, so that Primary = Field / Primary.
Divide primary by field	Divides the contents of the Primary Accumulator with the field, and stores the result in the Primary Accumulator, so that Primary = Primary / Field.
Modulo with primary	Divides the contents of the field with the contents of the Primary Accumulator, and stores the *remainder* of that operation in the Primary Accumulator, so that Primary = Field % Primary.
Modulo with field	Divides the contents of the Primary Accumulator with the contents of the field, and stores the *remainder* of that operation in the Primary Accumulator, so that Primary = Primary % Field.
Negate primary	Makes the contents of the Primary Accumulator negative (i.e., Primary = Primary * -1). **Note** The only way to subtract the Primary Accumulator from the field is to Negate the Primary Accumulator and then use the Sum in primary operation to add the negative Primary Accumulator to the field.
Move primary to alternate	Copies the contents of the Primary Accumulator to the Alternate Accumulator. This overwrites the current contents of the **Alternate Accum** field, so that Alternate = Primary.
Add primary to alternate	Adds the contents of the Primary Accumulator to the contents of the Alternate Accumulator, and stores the result in the Primary Accumulator, so that Primary = Primary + Alternate.
Multiply primary by alternate	Multiplies the contents of the Primary Accumulator with the contents of the Alternate Accumulator, and stores the result in the Primary Accumulator, so that Primary = Primary * Alternate.

(Contd) Part	Function
Divide primary by alternate	Divides the contents of the Primary Accumulator with the contents of the Alternate Accumulator, and stores the result in the Primary Accumulator, so that Primary = Primary / Alternate.
Modulo primary with alternate	Divides the contents of the Primary Accumulator with the contents of the Alternate Accumulator, and stores the remainder of that operation in the Primary Accumulator, so that Primary = Primary % Alternate.

How to Count Line Items
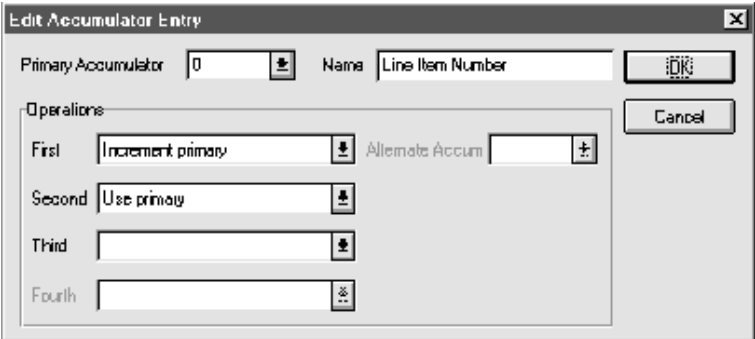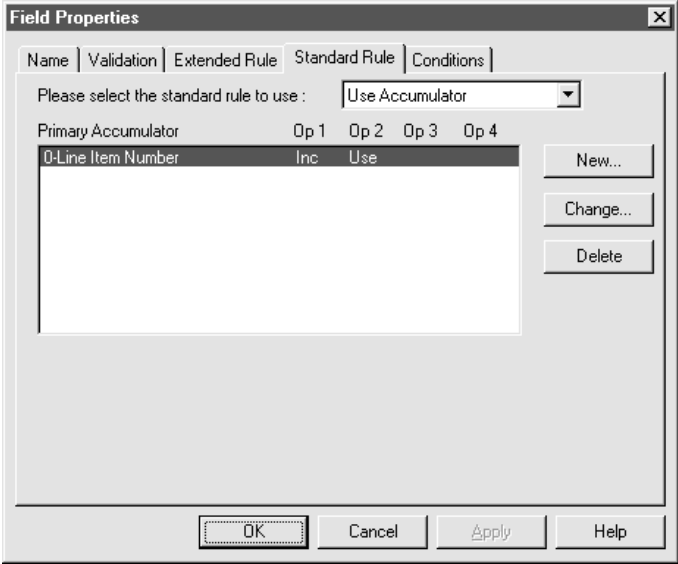
Introduction   This topic contains an example procedure for counting line items. For this example, assume that you want an incremental count of the number of line items, and you want to use that total value in the **Number of Line Items Total** field.

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

Step	Action
1	Double-click the field that you typically use to count the line items. **System Response** The system displays the Field Properties dialog box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Use Accumulator**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
4	Click **Yes** to continue.
5	Click **New**. **System Response** The system displays the Edit Accumulator Entry dialog box to create a new calculation for this field. *(Continued on next page)*

(Contd) Step	Action
6	From the **Primary Accumulator** list, select **0**. **System Response** The system activates the **First** operations box. **Note** Each map has only one set of accumulators. 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 is the result of the earlier calculation. The system performs any additional calculations you assign to that accumulator on the contents resulting from an earlier calculation.
7	In the **Name** box, type Line Item Number. **Note** This is a descriptive alias that enables you to differentiate the uses of the accumulators you create.
8	From the **First** list, select **Increment primary**. **Note** This is the first operation that the system performs. This specifies that the system increments the value in this accumulator by one for each iteration of the current group. (Continued on next page)
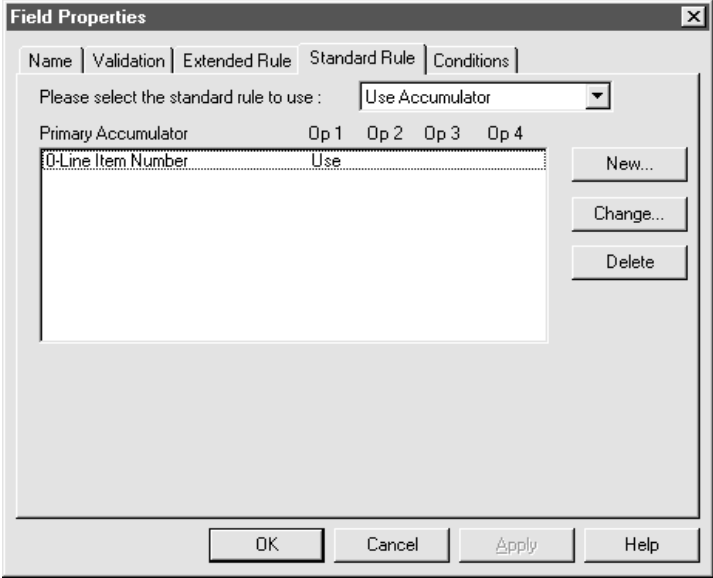
(Contd) Step	Action
9	From the **Second** list, select **Use primary**.
	Note This is the second operation that the system performs, after the First operation is completed. This specifies that the system loads the current value of the accumulator into the field.
	System Response This diagram illustrates how the Edit Accumulator Entry dialog box should look.
10	Click **OK** to add the accumulator.
	System Response This diagram illustrates how the Standard Rule tab should look.

(Contd) Step	Action
11	Click **OK** on the Field Properties dialog box to add the standard rule to the **Line Items** field.
12	Double-click the field that typically contains the total number of line items. **System Response** The system displays the Field Properties dialog box.
13	Select the Standard Rule tab to access standard rule options.
14	From the standard rule list, select **Use Accumulator**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
15	Click **Yes** to continue.
16	Click **New**. **System Response** The system displays the Edit Accumulator Entry dialog box to create a new calculation for this field.
17	From the **Primary Accumulator** list, select primary accumulator **0**. **System Response** The system activates the **First** operations box. **Note** Each map has only one set of accumulators. 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 is the result of the earlier calculation. The system performs any additional calculations you assign to that accumulator on the contents resulting from an earlier calculation.

(Contd) Step	Action
18	From the **First** list, select **Use primary**. **Note** This specifies that the system loads the current value of the accumulator into the Number of Line Items Total field. **System Response** This diagram illustrates how the Edit Accumulator dialog box should look. (Continued on next page)
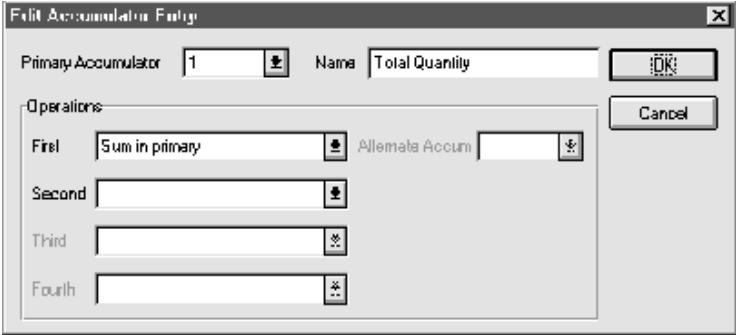
(Contd) Step	Action
19	Click **OK** to add the accumulator. **System Response** This diagram illustrates how the Standard Rule tab should look.
20	Click **OK** on the Field Properties dialog box to add the standard rule to the **Number of Line Items Total** field. **Note** The Number of Line Items Total field now contains the total number of line items in the purchase order.
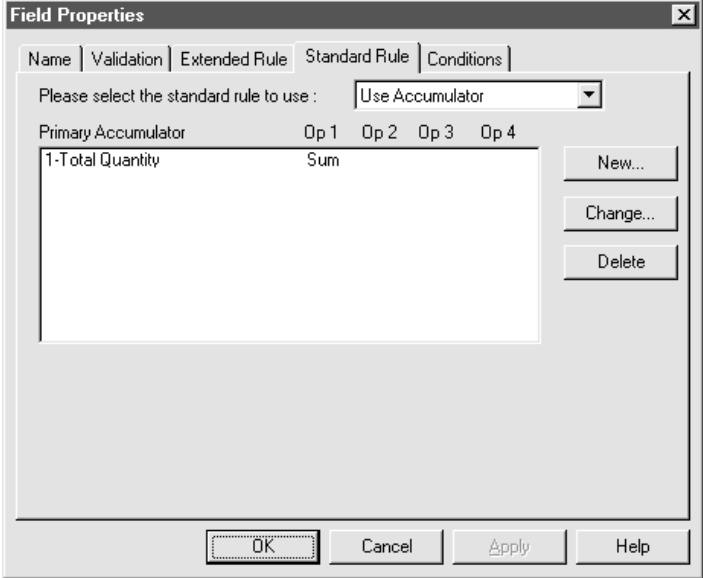
How to Calculate Hash Totals

Introduction   This table contains an example procedure. For this example, assume that you want to count the quantity ordered for each line item and load the total quantity in the Hash Total field.
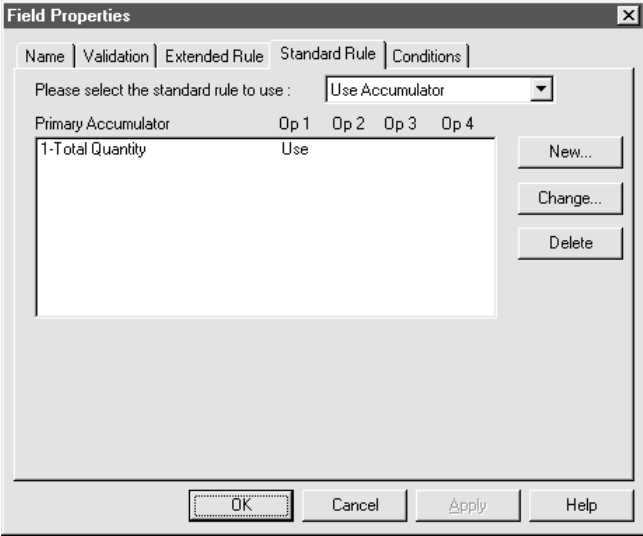
Procedure   Use this procedure to count the quantity ordered and generate a hash total for a purchase order.

Step	Action
1	Double-click the field that you typically use to count the line items. **System Response** The system displays the Field Properties dialog box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Use Accumulator**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
4	Click **Yes** to continue.
5	Click **New**. **System Response** The system displays the Edit Accumulator Entry dialog box to create a new calculation for this field.
6	From the **Primary Accumulator** list, select **1**. **System Response** The system activates the **First** operations box.
7	In the **Name** box, type Total Quantity. (Continued on next page)

(Contd) Step	Action
8	From the **First** list, select **Sum in primary**. **Note** This operation specifies that the system adds the numeric value (taking the positive or negative sign of the numbers into account) of the field to the contents of the Primary Accumulator for each iteration of the current group. **System Response** This diagram illustrates how the Edit Accumulator dialog box should look. (Continued on next page)

(Contd) Step	Action
9	Click **OK** to add the accumulator. **System Response** This diagram illustrates how the Standard Rule tab should look.
10	Click **OK** on the Field Properties dialog box to add the standard rule to the Line Items field.
11	Double-click the field that typically contains the total quantity of the purchase order. **System Response** The system displays the Field Properties dialog box.
12	Select the Standard Rule tab to access standard rule options.
13	From the standard rule list, select **Use Accumulator**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
14	Click **Yes** to continue. (Continued on next page)

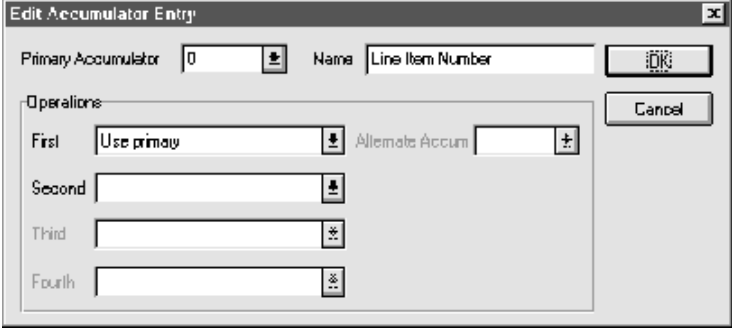| (Contd) Step	Action
15	Click **New**. **System Response** The system displays the Edit Accumulator Entry dialog box to create a new calculation for this field.
16	From the **Primary Accumulator** list, select **primary accumulator 1**. **System Response** The system activates the **First** operations box. **Note** This accumulator currently contains the total quantity.
17	From the **First** list, select **Use primary**. **Note** This operation specifies that the system loads the current value of the accumulator into the **Hash Total** field. **System Response** This diagram illustrates how the Edit Accumulator dialog box should look.

(Contd) Step	Action
18	Click **OK** to add the accumulator. **System Response** This diagram illustrates the Standard Rule tab.
19	Click **OK** on the Field Properties dialog box to add the standard rule to the **Hash Total** field. **Note** The Hash Total field now contains the total quantity of the purchase order.

How to Reset and Calculate a Value Total

Introduction
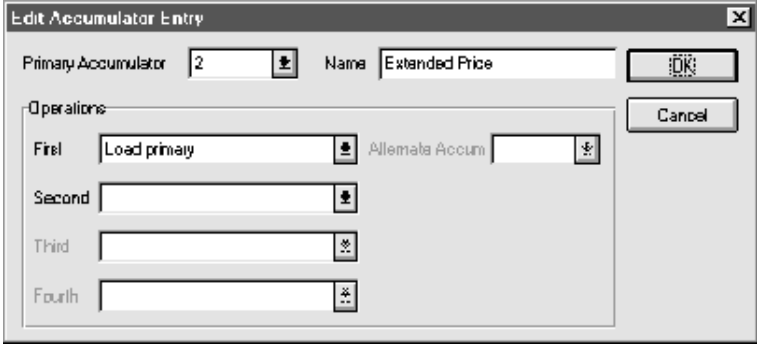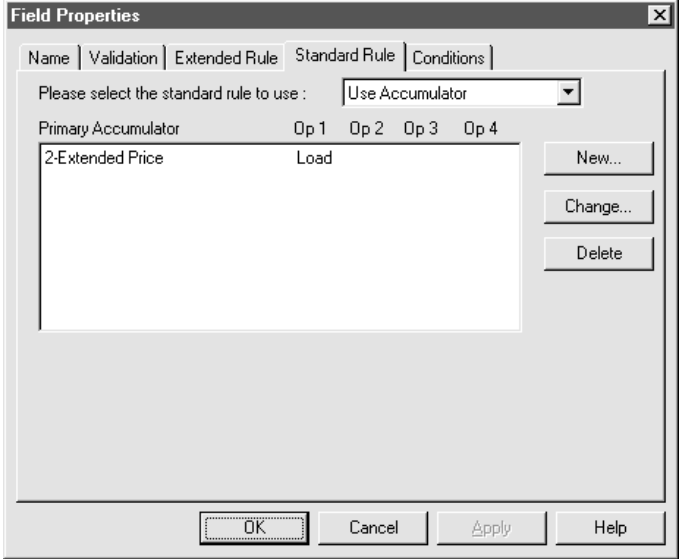
This topic contains an example procedure. For this example, assume that you want to multiply the quantity invoiced for each line item by the unit price to obtain the extended price. Then, you want to generate a running total of extended price and load the final total into the Total Invoice Amount field.

Procedure: multiplying quantity invoiced by unit price

Use this procedure to multiply the quantity invoiced for each line item by the unit price to obtain the extended price for an invoice.
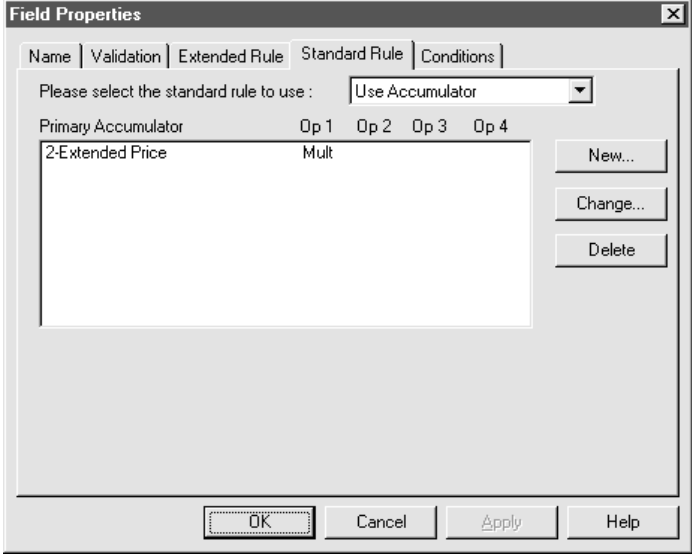
Step	Action
1	Double-click the field that you typically use to count the quantity invoiced. **System Response** The system displays the Field Properties dialog box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Use Accumulator**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
4	Click **Yes** to continue.
5	Click **New**. **System Response** The system displays the Edit Accumulator Entry dialog box to create a new calculation for this field.
6	From the **Primary Accumulator** list, select **2**. **System Response** The system activates the **First** operations box.
7	In the **Name** box, type Extended Price. (Continued on next page)

(Contd) Step	Action
8	From the **First** list, select **Load primary**. **Note** This operation specifies that the system loads the contents of the field into the Primary Accumulator for each iteration of the current group. **System Response** This diagram illustrates how the Edit Accumulator dialog box should look.
9	Click **OK** to add the accumulator. **System Response** This diagram illustrates the Standard Rule tab. (Continued on next page)

(Contd) Step	Action
10	Click **OK** on the Field Properties dialog box to add the standard rule to the **Quantity Invoiced** field.
11	Double-click the field that contains the unit price for each line item. **System Response** The system displays the Field Properties dialog box.
12	Select the Standard Rule tab to access standard rule options.
13	From the standard rule list, select **Use Accumulator**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
14	Click **Yes** to continue.
15	Click **New**. **System Response** The system displays the Edit Accumulator Entry dialog box to create a new calculation for this field.
16	From the **Primary Accumulator** list, select **primary accumulator 2**. **System Response** The system activates the **First** operations box. (Continued on next page)
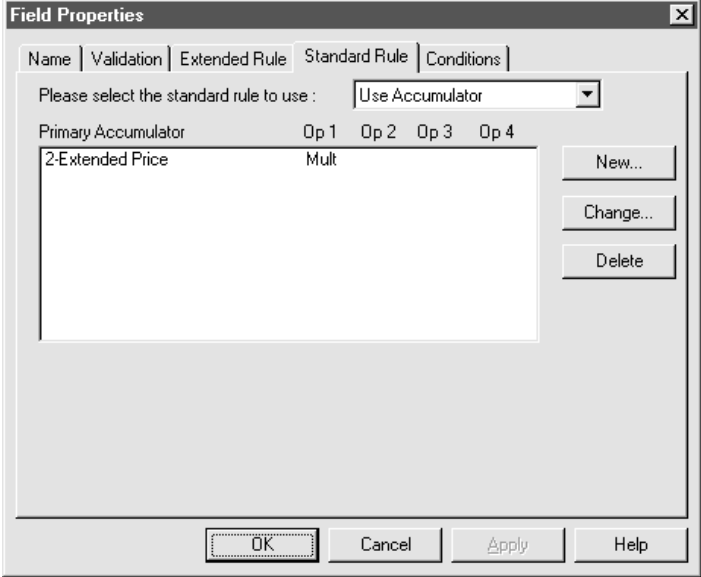
(Contd) Step	Action
17	From the **First** list, select **Multiply with primary**. **Note** This operation specifies that the system multiplies the value of the Unit Price field with the contents of the primary accumulator, and store the result in the primary accumulator for each iteration of the current group. **System Response** This diagram illustrates how the Edit Accumulator dialog box should look. (Continued on next page)

(Contd) Step	Action
18	Click **OK** to add the accumulator. **System Response** This diagram illustrates how the Standard Rule tab should look.
19	Click **OK** on the Field Properties dialog box to add the standard rule to the **Unit Price** field. **Tip** If there is an extended price field in your file, you could load the total from the extended price calculation into that field. To do this, you need to use an accumulator on that extended price field that specifies **Use primary** for accumulator 2.
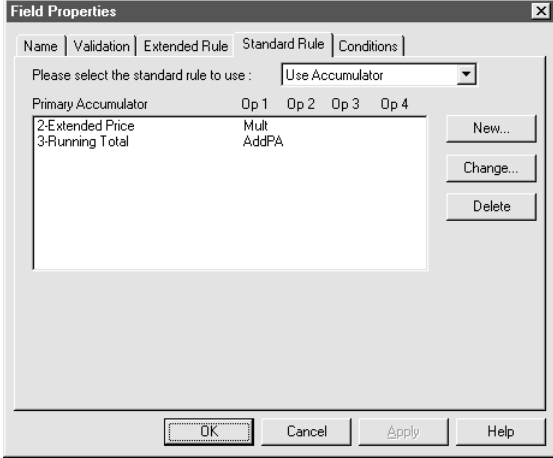
Procedure: generating a running total of extended price

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

Step	Action
1	Double-click the field that contains the unit price for each line item. **System Response** You already established one accumulator that is displayed in the list on the Standard Rule tab.
2	Click **New**. **System Response** The system displays the Edit Accumulator Entry dialog box to create a new calculation for this field.
3	From the **Primary Accumulator** list, select **3**. **System Response** The system activates the **First** operations box.
4	In the **Name** box, type Running Total. (Continued on next page)

(Contd) Step	Action
5	From the **First** list, select **Add primary** to alternate. **Note** This operation specifies that the system adds the contents of the primary accumulator to the contents of the alternate accumulator, and stores the result in the primary accumulator for each iteration of the current group.
6	From the **Alternate Accum** list, select **2**. **Note** This operation instructs the system to add the value of accumulator 2 (which contains the extended price for a line item) to the value of accumulator 3. The system stores the sum in accumulator 3, resulting in a running total of the extended price with each iteration of the current group. **System Response** This diagram illustrates how the Edit Accumulator dialog box should look. (Continued on next page)

(Contd) Step	Action
7	Click **OK** to add the accumulator. **System Response** This diagram illustrates how the Standard Rule tab should look.
8	Click **OK** on the Field Properties dialog box to add the standard rule to the **Unit Price** field.
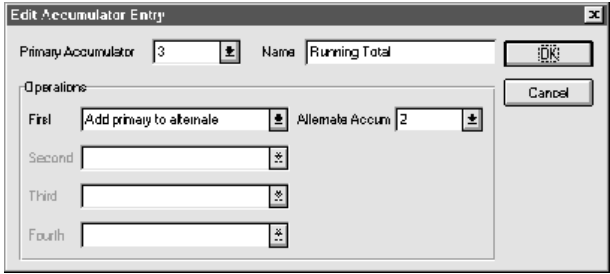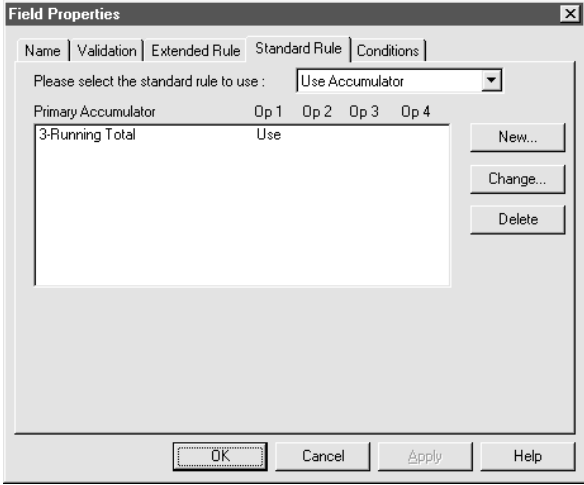
Procedure: loading a running total of extended price

Use this procedure to load the running total of the extended price into the Total Invoice Amount field.

Step	Action
1	Double-click the **Unit Price** field. **System Response** The system displays the Field Properties dialog box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Use Accumulator**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
4	Click **Yes** to continue. (Continued on next page)

(Contd) Step	Action
5	Click **New**. **System Response** The system displays the Edit Accumulator Entry dialog box to create a new calculation for this field.
6	From the **Primary Accumulator** list, select **3**. **System Response** The system activates the **First** operations box.
7	From the **First** list, select **Use primary**. **Note** This operation specifies that the system loads the contents of the primary accumulator into the **Total Invoice Amount** field. **System Response** This diagram illustrates how the Edit Accumulator dialog box should look. (Continued on next page)

(Contd) Step	Action
8	From the **Alternate Accum** list, select **2**. **Note** This operation instructs the system to add the value of accumulator 2 (which contains the extended price for a line item) to the value of accumulator 3. The system stores the sum in accumulator 3, resulting in a running total of the extended price with each iteration of the current group. **System Response** This diagram illustrates how the Edit Accumulator dialog box should look.
9	Click **OK** to add the accumulator. **System Response** This diagram illustrates how the Standard Rule tab should look.
10	Click **OK** on the Field Properties dialog box to add the standard rule to the **Total Invoice Amount** field.

Using the Use Code Standard Rule

Overview

Introduction

The Use Code standard rule enables you to compare the value of an element or field to values in a predefined code table. You can apply a Use Code standard rule to an element or field to:

◗ Check the element's value against a code table.

◗ Select from the codes in a specified code table.

◗ Store a code's description in another element or field.

You can also specify whether or not the system generates an error if a validation against the code table fails.

Code tables

The Gentran:Server Application subsystem enables you to create code tables to use with the current map.

Note
While you can use Application Integration code tables for only the current map, you can create a code table in Gentran:Server that you can use globally for all maps.

Standard Code List tables

EDI standards use Code List Tables as repositories for lists of codes defined for elements.

Each EDI standard provides a code list table for each element that can be further defined with a code. Gentran:Server allows you to load code list tables from the standard. You can either load all the codes in the table, or you can select only one or more codes from the table.

Once you load a code list table, you can use a Use Code standard rule with an element or field to either look up a value from a code table or validate the contents of an element or field against the values in the code table.

Code List tables are versatile tools. You can import and export them and copy and paste them between maps.

Qualifying another element with a Use Code rule

You can use the Use Codes standard rule to further qualify another element.

Example

If the XX element contains address information, you can further qualify that element by choosing the code SU from the 0222 table. In the 0222 table, the code SU is described as a supplier's address. Therefore, by using this code with the XX element, you are indicating that the XX element is not just address information, but address information for the supplier.

Standard Rule Tab (Use Code List)

Illustration   This diagram illustrates the Standard Rule tab with Use Code rule selected.
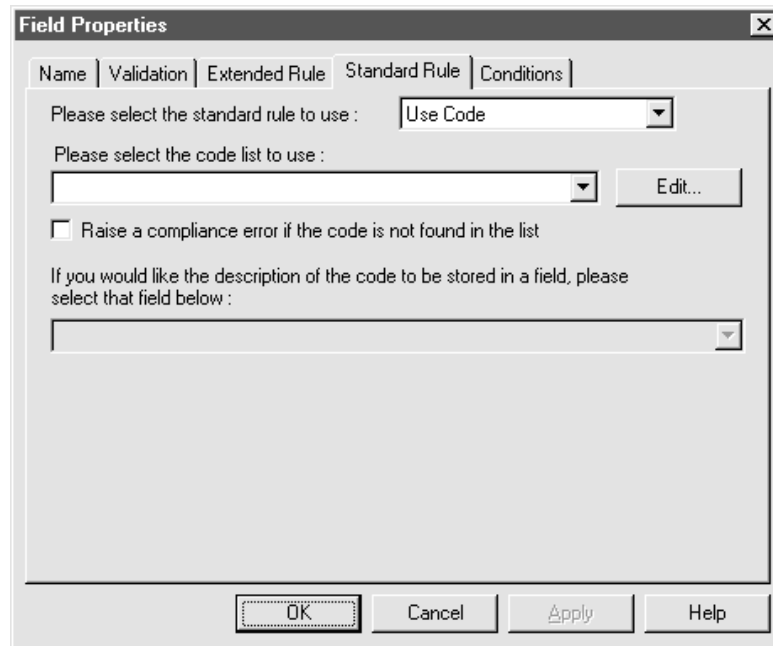
Parts and functions   This table lists the parts of the Standard Rule tab (with the Use Code standard rule selected) and their functions.

Part	Function
Code list	Displays a list of all the code tables that have been added or loaded to your system. **Note** If the necessary code table is not listed, click **Edit** to load or create a code table.
Edit	Accesses the Edit Code List dialog box, which enables you to create and edit code list tables. (Continued on next page)

(Contd) Part	Function
Raise compliance error	Indicates that, for compliance reasons, the field *must* contain one of the codes from the specified table. Nothing else is valid for that field. **Example** If a field is defined as containing only YES or NO, you can set up an exclusive code table that contains only YES and NO. Then, if you receive a MAYBE in that field, the system flags it as an error.
Code description	Contains the name of the field that holds the description of the code. **Example** If the code is SU, it is much more useful to view the description of the code (Supplier's Address). If you selected field XX from the store description list, the description for the code is mapped to field XX.

Code List Dialog Box

Illustration   This diagram illustrates the Code Lists dialog box.
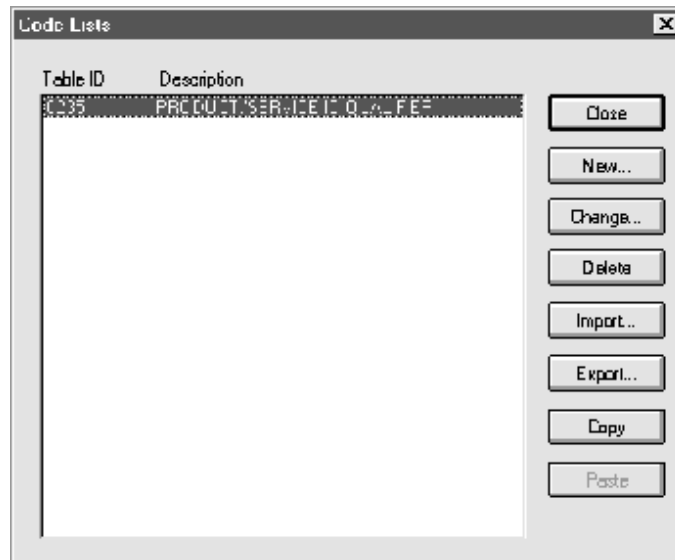
Parts and functions   This table lists the parts of the Code Lists dialog box and their functions.

Part	Function
Table ID	Specifies the table identifier.
Description	Describes the table.
Close	Closes the dialog box without saving changes.
New	Accesses the Edit Code List dialog box, which enables you to create a new code list.
Change	Accesses the Edit Code List dialog box, which enables you to edit the selected code list.
Delete	Deletes the selected code list
Import	Accesses the Open dialog box, which allows you to import a code list.
Export	Accesses the Save As dialog box, which allows you to export the selected code list.

Part	Function
Copy	Copies the selected code list.
Paste	Pastes a previously-copied code list in a map.

Edit Code List Dialog Box

Illustration   This diagram illustrates the Edit Code List dialog box.
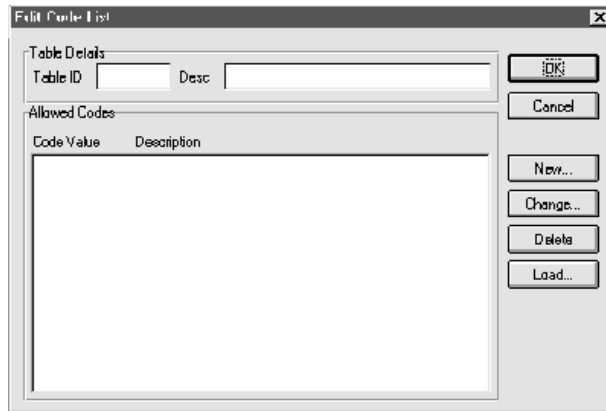
Parts and functions   This table lists the parts of the Edit Code List dialog box and their functions.

Part	Function
Table ID	Contains the name of the field for which this code list table is used.
Desc	Contains the description of the field for which this code list table is used.
Allowed Codes	Specifies the codes that are allowed for this table.
New	Accesses the Edit Code List Entry dialog box, which allows you to create a new code.
Change	Accesses the Edit Code List Entry dialog box, which allows you to edit the selected code.
Delete	Accesses the Edit Code List Entry dialog box, which allows you to delete the selected code.
Load	Accesses the Load Code List dialog box, which allows you to select, from a standard code table, specific codes that you want to load or select the entire list of codes.

Edit Code List Entry Dialog Box

Illustration   This diagram illustrates the Edit Code List Entry dialog box.

Parts and functions   This table lists the parts of the Edit Code List Entry dialog box and their functions.

Part	Function
Value	Specifies the value of the code.
Description	Contains the description of the code value. **Note** The description is used if you specify a field (in the Store Fields list on the Field Properties dialog box) to which you want the code description mapped.

How to Validate Data Against a Code List Table

Introduction   This topic contains an example procedure. For this example, assume that you need to map the customer product code from the Product/Service ID to the customer product code field (CUSTPROCODE) on the application side of the map. To ensure that the data that is mapped from the Product/Service ID element is your trading partner's customer product code, you need to establish and use a code table with the Product Service ID *Qualifier*.

Procedure   Use this procedure to validate the data in the Product/Service ID field against a code list table.

Step	Action
1	Load the code list table from the standard for the Product/Service ID Qualifier element. **Reference** See How to Load a Code List Table from the Standard for more information on loading a code table. **Note** The standards provide code list tables only for elements that use them. For example, in the TD4 segment, the TD401 (Special Handling Code) and TD403 (Hazardous Material Class Code) have code tables.
2	Define a code list table for the Product/Service ID Qualifier element that contains only the code value BP (Buyer's Part Number). **Reference** See How to Define and Modify a Code List for more information on defining a code table.
3	Double-click the element for which you need to validate data against a code table (Product/Service ID Qualifier). **System Response** The system displays the Element Properties dialog box.
4	Select the **Standard Rule** tab to access standard rule options. *(Continued on next page)*

(Contd) Step	Action
5	From the standard rule list, select **Use Code**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
6	Click **Yes** to continue.
7	From the code list, select the code list table that the data in this field is validated against (0235).
8	If you need to specify that, for compliance reasons, the element *must* contain one of the codes from the specified table (nothing else is valid for that field), select the compliance error check box.
9	Click **OK** to add this standard rule to the field.

How to Define and Modify a Code List

Procedure   Use this procedure to define or modify a code list table.

Step	Action
1	From the **Edit** menu, select **Code Lists**. **System Response** The system displays the Code Lists dialog box.
2	Do you want to create a new code list? ◗ If YES, click **New**. ◗ If NO (you want to edit a code list), select a code list and click **Change**. **System Response** The system displays the Edit Code List dialog box.
3	In the **Table ID** box, type the name of the field for which this code list table is used.
4	In the **Description** box, type the description of the field for which this code list table is used.
5	Do you want to create a new code? ◗ If YES, click **New**. ◗ If NO (you want to edit a code), select a code and click **Change**. **System Response** The system displays the Edit Code List Entry dialog box.
6	In the **Value** box, type the actual value of the code.
7	In the **Description** box, type a description of the code value.
8	Click **OK** to save the code list entry.
9	Repeat Steps 5 through 8 to add more code list entries to the code list table.
10	Click **Close** to save and exit the Edit Code List dialog box.
11	Click **Close** to exit the Code Lists dialog box.

How to Delete a Code List or Code List Entry

Procedure: deleting a code list

Use this procedure to delete a code list table.

Step	Action
1	From the **Edit** menu, select **Code Lists**. **System Response** The system displays the Code Lists dialog box.
2	Select the code list you want to delete.
3	Click **Delete** to delete the code list table. **System Response** The selected table is deleted without warning.

Procedure: deleting a code list entry

Use this procedure to delete a code list entry.

Step	Action
1	From the **Edit** menu, select **Code Lists**. **System Response** The system displays the Code Lists dialog box.
2	Select the code list from which you want to delete an entry and click **Change**. **System Response** The system displays the Edit Code List Entry dialog box.
3	Select the entry and click **Delete** to remove the entry. **System Response** The selected entry is deleted without warning.
4	Click **OK** to save the code list table.

How to Import a Code List

Introduction   The Code List Import function enables you to import code lists created for another map and share code lists with other users of Gentran:Server.

Procedure   Use this procedure to import a code list table.

Step	Action
1	From the **Edit** menu, select **Code Lists**. **System Response** The system displays the Code Lists dialog box.
2	Click **Import**. **System Response** The system displays the Open dialog box.
3	Select the location of the code list file. **Note** The default location is the folder where your Application Integration system is installed. The default file extension for code lists is .CDE. (Continued on next page)

(Contd) Step	Action
4	Select the code list file from the list and click **Open**. **System Response** The system imports the code list and returns to the Code Lists dialog box. The imported code list is available for your use.
5	Click **OK** to exit the Code Lists dialog box.

How to Export a Code List

Introduction   The Code List Export function enables you to export code lists to file. This enables you to define a code list for one map and use that code list in another map. This function also enables you to share code lists with other users of Gentran:Server.

Procedure   Use this procedure to export a code list table.

Step	Action
1	From the **Edit** menu, select **Code Lists**. **System Response** The system displays the Code Lists dialog box.
2	Select a code list and click **Export**. **System Response** The system displays the Save As dialog box.
3	Do you want to change the name? ◗ If YES, type the name of the export file in the File name box. ◗ If NO, continue with the next step. **Note** The file name defaults to the table ID with a .CDE file extension. The default location is the folder where your Application Integration system is installed. *(Continued on next page)*

(Contd) Step	Action
4	Click **Save**. **System Response** The system exports the code list and returns to the Code Lists dialog box.
5	Click **Close** to exit the Code Lists dialog box.

How to Load a Code List Table from the Standard

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

Step	Action
1	Double-click the element for which you need to use a code table. **System Response** The system displays the Element Properties dialog box. **Note** The standards provide code list tables only for elements that use them. For example, in the TD4 segment, the TD401 (Special Handling Code) and TD403 (Hazardous Material Class Code) have code tables that the standard provides.
2	Select the **Standard Rule** tab to access standard rule options.
3	From the standard rule list, select **Use Code**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
4	Click **Yes** to continue.
5	Click **Edit**. **System Response** The system displays the Edit Code List dialog box.
6	Click **Load**. **System Response** The system prompts you to select a data source name (DSN) from which to access the EDI standard.
7	Select the appropriate DSN and click **OK**. **System Response** The system displays the Load Code List dialog box. (Continued on next page)

(Contd) Step	Action
8	Do you want to load the entire list of codes? ◗ If YES, click **Add All** to select all the codes and move them to the Codes Selected list. GO TO Step 10. ◗ If NO, (you want to load specific codes) GO TO Step 9. **Recommendation** Add just the codes that you and your trading partners are able to create or accept. Adding all the codes in the code table (using the **Add All**) creates a much larger translation object than if you use only selected codes.
9	From the **Codes in Standard** list, highlight each code that you want to load and then click **Add** to move it to the Codes Selected list.
10	Click **OK** to load the code list.
11	Click **Close** to exit the Code Lists dialog box.

How to Copy and Paste Code Lists

Introduction   The Code List Copy and Paste function enables you to copy code lists to from one map to another.

Procedure   Use this procedure to copy and paste a code list table.

Step	Action
1	From the **Edit** menu, select **Code Lists**. **System Response** The system displays the Code Lists dialog box.
2	Select a code list and click **Copy**. **System Response** The system copies the code list to the clipboard.
3	Click **Close** to exit the Code Lists dialog box.
4	Is the map open? ▸ If YES, continue with Step 5. ▸ If NO, open the map in which you want to use the code list.
5	From the **Edit** menu, select **Code Lists**. **System Response** The system displays the Code Lists dialog box.
6	Click **Paste**. **System Response** The system adds the copied code list to this map.
7	Click **Close** to exit the Code Lists dialog box.

How to Validate Data Against Code List Tables

Procedure   Use this procedure to validate data against a code list table.

Step	Action
1	Double-click the field for which you need to validate data against a code table. **System Response** The system displays the Field Properties dialog box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Use Code**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?".
4	Click **Yes** to continue.
5	From the code list, select the code list table that you want the data in this field validated against. **Note** If this list is empty, you must create a code table.
6	Do you want to specify that the field must contain one of the codes from the specified table? ◗ If YES, select the **Raise a compliance error…** check box. ◗ If NO, continue with Step 7.
7	Click **OK** to add this standard rule to the field.

How to Map Code Item Descriptions

Procedure   Use this procedure to map a code item description.

Step	Action
1	Double-click the field for which you need to map a code description. **System Response** The system displays the Field Properties dialog box.
2	Select the Standard Rule tab to access standard rule options.
3	From the standard rule list, select **Use Code**. **System Response** The system displays this prompt: "Are you sure you want to change the Standard Rule for this field?"
4	Click **Yes** to continue.
5	From the code list, select the code list table that the data in this field is validated against. **Note** If this list is empty, you need to either create a code table or load an existing code table into the code table list.
6	Do you want to specify that the field must contain one of the codes from the specified table? ◗ If YES, select the **Raise a compliance error…** check box. ◗ If NO, continue with Step 6.
7	From the store description list, select the field to which you want the description of the code item mapped.
8	Click **OK** to add this standard rule to the field.

Using Extended Rules

Contents

Overview

Introduction

In this chapter   This chapter explains:

◗   What extended rules are and when they are processed

◗   How to use extended rules

◗   How to define extended rules

◗   Extended rule syntax

It also contains an alphabetic language reference guide to Gentran:Server proprietary programming language keywords and commands used in extended rules.

Key terms   This table lists the key terms used in this chapter.

Term	Description
command	A keyword that performs actions on variables. Commands are responsible for moving data.
expression	A logical unit of information that the system evaluates.
extended rule	An Application Integration feature that enables you to apply complex mapping operations to fields.
global scope	A variable in an extended rule that can be accessed from any other extended rule in the map.
keyword	A fixed defined use of a word that indicates how the programming language should be interpreted.
operator	The simplest operation in an expression.
pre-session rule	An extended rule that the system processes before it processes the translation object.
post-session rule	An extended rule that the system processes after it processes the translation object.

(Contd) Term	Description
scope	The range of an extended rule. The scope determines which variables in the rule are accessible from other extended rules. The scope depends on the current state of the map.
symbol	A variable, constant, map component, array, or other item that is the focus of an operation.
variable	An item that you define in the declarations section of an extended rule to store values. Variables consist of a name and a data type.

Introducing Extended Rules

Extended Rule

Definition
An extended rule is an Application Integration feature that enables you to apply complex mapping operations to fields. Extended rules give you access to functions that are more involved than standard rules.

You define extended rules with the Gentran:Server proprietary programming language. This is a full programming language that gives you access to the entire Gentran:Server internal storage area.

When to use extended rules
Use extended rules when you need to define more complex translations than are available through the link function (simple mapping) and standard rules.

You can define an extended rule as:

◗ A session rule (pre-session or post-session)

◗ A rule for a map component

What comprises an extended rule?
An extended rule can consist of two sections, a *declarations* section followed by a *statements* section.

Every extended rule must have a statements section. In the *statements* section, you define the actions that you want the rule to execute.

If you use additional variables, you must have a declarations section for the rule. You use this section to declare any variables that you use in a rule that are not already defined as part of the input or output specification of the map.

Reference
See Declarations Section for more information about the declarations section.

See Statements Section for more information about the statements section.

Declarations Section

Introduction   The declarations section is used to declare the names and types of any variables you use either in the current rule or in any other rule that is within the scope of the current rule. The variables that you define in the declarations section store values.

Variables   Variables consist of a name and a data type.

Rules for variable names
These are the rules for naming variables:

◗   Variable names can include alphanumeric characters and the colon and underscore characters.

◗   The first character in a variable's name can not be numeric.

◗   All variable names are case-sensitive.

Supported data types

These are the data types that extended rules support:

◗   Integer – is a whole number with no decimal component.

◗   Real – is a whole number that may have a decimal component.

◗   String – contains one or more printable characters.

◗   Datetime – contains a date or time.

◗   Array – defines a table of multiple occurrences of a single data type.

Terminating a declaration   You must terminate a declaration with a semicolon (;). To improve readability, you can include a blank line in between the declaration and statement sections.

Variable declaration examples

This list contains examples of declarations of variables of each data type.

◗   Declare **i** as an integer
integer i;

◗   Declare **r** as a real number
real r;

◗   Declare **d** as a date or time
datetime d;

◗   Declare **s** as a 20-character string
string[20] s;

◗   Declare **a** as an array of 10 integers
integer a[10];

◗   Declare **p** as an array of 50 10-character strings
string[10] p[50];

Statements Section

Introduction	The statements section defines the actual work that an extended rule performs.

Statements and combinations	A statement is a single operation that consists of a combination of expressions, keywords, commands, operators, and symbols. A statement combination enables you to specify more complex operations.

What is an expression?	An expression is a logical unit that the system evaluates. The statements section consists of a sensible combination of keywords, operators, and symbols. **Examples** A = B A + B **Reference** See Extended Rule Syntax for a detailed explanation of these components.

Extended Rules Processing

Introduction

The system always processes field level extended rules after processing standard rules. After that, the point at which the system processes an extended rule depends on:

◗   Whether the rule is designated as pre- or post- session
◗   The scope of the extended rule
◗   The level of the map component that has the extended rule
◗   Whether the rule is attached to a component on the input side of the map or the output side

Pre-session and post-session rules

You can specify that an extended rule is pre- or post-session on the Session Rules dialog box.

The system processes pre-session extended rules before processing the translation object. The variables in the extended rules are accessible for every extended rule defined in the translation object.

The system processes post-session rules after processing the translation object.

Levels of extended rules

You can attach extended rules to three different levels of map components:

◗   EDI standard file and application file groups, using the Loop Level Extended Rules dialog box.
◗   Groups, sub-groups, repeating records, and repeating segments, using the Loop Level Extended Rules dialog box.
◗   Fields and elements, using the Field Level Extended Rule dialog box.

Scope

The *scope* of an extended rule determines which variables are accessible from within a given extended rule. The scope depends on the current state of the map.

This list defines the scope of an extended rule.

◗   Pre-session extended rules (defined on the Session Rules dialog box) are in scope for every rule in the translation object.
◗   On Begin extended rules (defined on the Loop Level Extended Rules dialog box) are in scope until the conclusion of its companion On End rule (also defined on the Loop Level Extended Rules dialog box).
◗   Field level extended rules are in scope only for the duration of the field or element.

Scope (contd)

The scope of an extended rule attached to the current map component depends on the type and state of the map component.

If a current group to which the rule is attached is subordinate to another group, then:

◗ The parent group is automatically in scope for the duration of the entire child group

◗ The current hierarchical structure is also in scope for the duration of the child by using an addressing method. This method is explained in the symbol syntax for <u>String constant</u>.

An extended rule that is attached to a field or element is in scope only for the duration of the field or element. The system always processes field-level extended rules after processing standard rules.

A variable is considered to be "in scope" if it is:

◗ declared in the current rule,

◗ in the On Begin rule of a group that contains the current map component, or

◗ in the Pre-Session rule.

Input side and Output side of a map

The translator builds the data storage area for a map based on the structure of the input (source) side of the map. Therefore, extended rules address the map based on the hierarchy of the input side.

On the output side of a map, extended rules have access only to the current record and the entire input side of the map. However, from the input side of a map, you have access to the entire file structure. The translator does not write to the output file until it has read the entire input file. So, you can change the values output from any rule on the input side of a map or any rule on the output side until the time that the translator writes the output file.

Tip

When you use extended rules, always address the input side of the map so the translator can locate the map component that the rule accesses.

Input rule processing

The translator processes the input side of the map first and then the output side. This table lists the sequence of how the translator processes rules on the Input side of a map.

Step	Action
1	Load the input definition.
2	Read the input file.
3	Determine if data is present for the first/next group. If it is present, run the group On_Begin rule, if the rule is present.
4	Load each field within the first/next record within the current group and then execute field level rules (for each field) in the following sequence: ◗ Standard rules ◗ Extended rules
5	At the end of the group, execute the On_End rule, if the rule is present.
6	Repeat steps 2 - 5 for each group in the input file.

Output rule processing

This table lists the sequence of how the translator processes rules on the output (destination) side of a map.

Step	Action
1	Verify whether or not data exists for the first/next record.
2	If the record is the first record of a group, run On_Begin rule, if the rule is present.
3	For each field in the record, execute field level rules (for each field) in the following sequence: ◗ Run standard rules ◗ Run extended rules
4	Format data according to specified field properties (see the Field Properties dialog box).
5	Write the record to the output file. (Continued on next page)

(Contd) Step	Action
6	At the end of the group, execute On_End rule, if the rule is present.
7	Repeat Steps 1 - 6 for each record in the output file.

Rule processing diagram

This diagram illustrates when loop level (On Begin and On End) extended rules and field level extended rules are processed in relation to the system process flow.
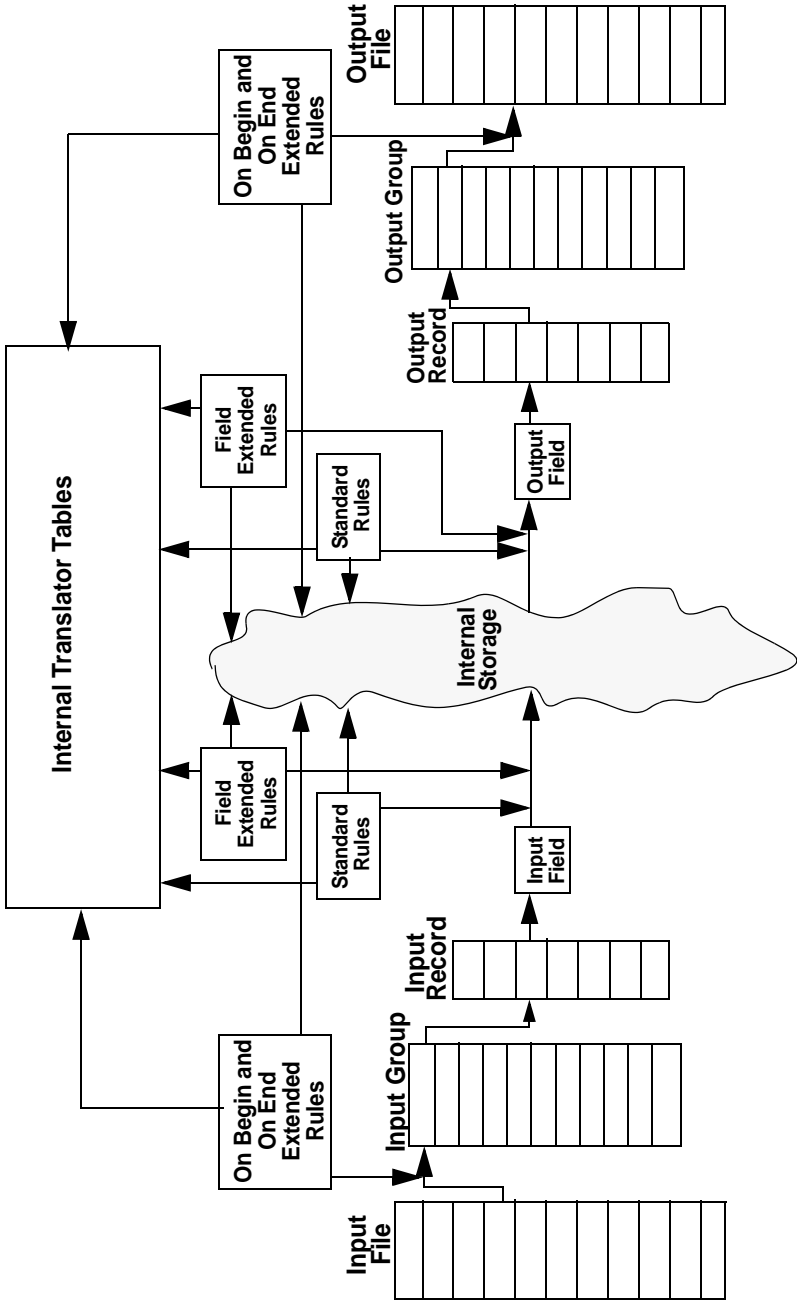
Defining Extended Rules

Overview

Introduction	Before you define an extended rule, you need to decide:

◗   what you want the scope of the rule to be

◗   when you want the rule to be executed

Levels	You can define extended rules to access components at these levels:

◗   the entire session (input and output sides of the map)

◗   looping map components (groups, segments, etc.)

◗   fields

The procedure you follow to define an extended rule varies slightly, depending on whether you are defining a session rule or a rule for a map component.

Session rules	A session rule applies to components on both the input and output sides of the map. There are two types of session rules:

◗   Pre-session

◗   Post-session

You can define both a pre-session and a post-session rule for a given session.

Pre-session
Pre-session rules define variables that must have global scope. Global scope means that the variable can be accessed from any other extended rule in the map. The system processes pre-session extended rules before processing the translation object, so variables in pre-session rules are in scope for every extended rule defined in the translation object.
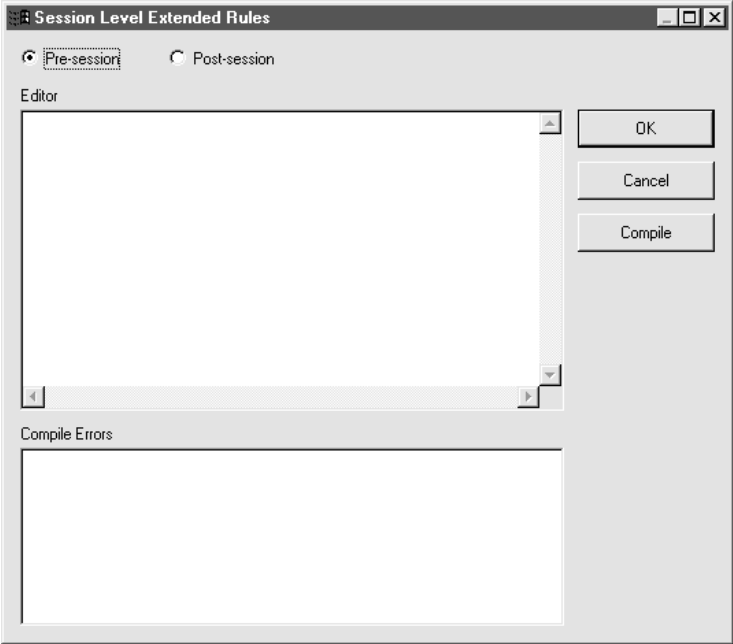
Post-session
The system executes post-session rules after processing the translation object. Thus, post-session rules do not have permanent scope.

How to Define a Session Rule

Introduction   This topic explains how to define a pre-session or a post-session extended rule.

Procedure   Use this procedure to define a session rule.

Step	Action
1	From the Edit menu, select **Session Rules**. **System Response** The system displays the Session Level Extended Rules dialog box.
2	Do you want to define a pre-session rule? ◗ If YES, click the Pre-session option. ◗ If NO (you want to define a Post-session rule), click the Post-session option.
3	In the Editor list, type the extended rule. (Continued on next page)

(Contd) Step	Action
4	Do you want to check the rule for errors? ◗ If YES, click **Compile** to compile the extended rule and then continue with Step 5. **System Response** The system displays any warnings or errors for the rule in the Compile Errors list. ◗ If NO, GO TO Step 6. **Note** The Compile function on the Session Level Extended Rules dialog box gives you immediate feedback about the accuracy of the rule you added, but not the rest of the map. The system does not fully compile the rule until you compile the entire map.
5	Correct any errors that the system flagged and click **Compile** again. **Note** Repeat this process until there are no errors.
6	Click **OK** to add the extended rule.
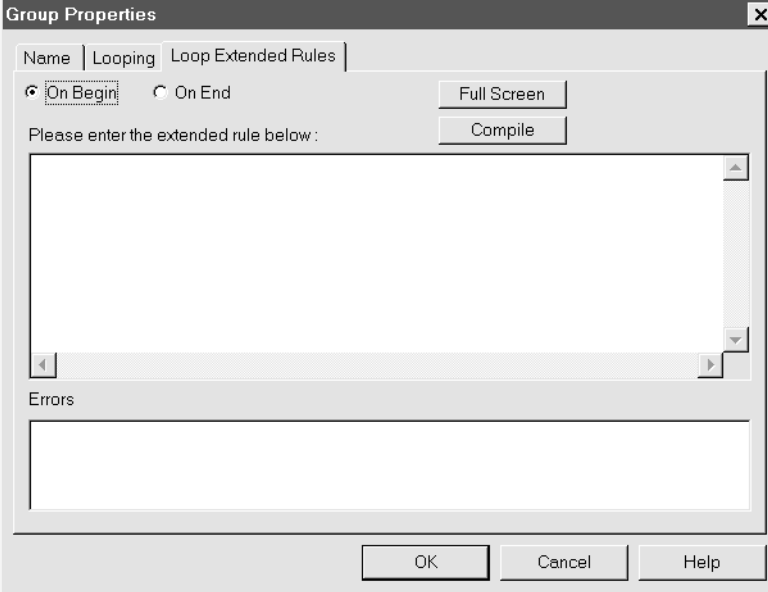
How to Define a Map Component Rule

Introduction   You can define extended rules for an EDI standard file, an application file, a group, a sub-group, a repeating record, a repeating segment, a field, or an element.

Procedure   Use this procedure to define an extended rule for a map component.

Step	Action
1	Right-click the map component to display the shortcut menu.
2	Use this table to determine your next action.

	IF the map component is…	**THEN select…**
	an EDI standard file, application file, group, sub-group, repeating record, or repeating segment	Extended Rules and continue with Step 3. **System Response** The system displays the Loop Level Extended Rules dialog box.
	a field or element	Extended Rule and GO TO Step 4. **System Response** The system displays the Properties dialog box for the component. (Continued on next page)

(Contd) Step	Action
3	Do you want the system to execute the extended rule before it processes the map component? ◗ If YES, select the **On Begi**n option. ◗ If NO (you want the rule to be executed when the system concludes its processing of that map component), select the **On End** option. **Note** You can define both an **On Begin** and an **On End** rule for a single map component.
4	Type the extended rule. **Reference** See Alphabetic Language Reference for rule syntax. (Continued on next page)

(Contd) Step	Action
5	Do you want to check the rule for errors?
	◗ If YES, click **Compile** to compile the extended rule and then continue with Step 6.
	System Response The system displays any warnings or errors in the **Errors** list.
	◗ If NO, GO TO Step 7.
	Note The Compile function gives you immediate feedback about the accuracy of your rule. The rule is compiled when you compile the entire translation object.
6	Correct any errors that the system flagged and click **Compile** again.
	Note Repeat this step until there are no errors.
7	Click **OK** to add the extended rule.

Extended Rule Syntax

Overview

Introduction

The statements section of an extended rule consists of a sensible combination of keywords, operators, and symbols. This section explains the correct syntax for each of these components.

Basic syntax rules

Use spaces and operators to separate keywords and symbols.

Do not string two keywords together sequentially without an operator.

Keywords and Commands

Introduction

A *keyword* is a fixed, defined use of a word that indicates how the programming language should be interpreted. There are two types of keywords:

◗   execution control

◗   command

Definition: execution control

The execution control keyword controls the flow of execution of the defined rule. These keywords are used to evaluate conditions and perform looping operations.

Definition: command

A command keyword performs actions on variables. Commands are responsible for moving data.

List of execution control keywords

This is a list of Gentran:Server execution control keywords.

◗   IF

◗   THEN

◗   ELSE

◗   BEGIN

◗   END

◗   WHILE

◗   DO

◗   CONTINUE

◗   BREAK

List of commands

This is a list of Gentran:Server commands.

◗   GET

◗   SET

◗   STRDATE

◗   CONCAT

List of commands (cont'd)

- ◗ LEN
- ◗ ATOI
- ◗ ATON
- ◗ CERROR
- ◗ EMPTY
- ◗ EXEC
- ◗ EXIST
- ◗ INDEX
- ◗ NTOA
- ◗ COUNT
- ◗ DELETE
- ◗ FSEEK
- ◗ FTELL
- ◗ PARAM
- ◗ READBLOCK
- ◗ UNREADBLOCK
- ◗ READBYTES
- ◗ SETPARAM
- ◗ LEFT
- ◗ MID
- ◗ RIGHT
- ◗ WRITEBLOCK
- ◗ WRITEBYTES

Note
You must terminate a statement with a semicolon (;).

Operators

Introduction   Operators define the simplest operation in an expression.

List of operators   This table lists the operators that you can use in extended rules and their functions.

Part	Function
+	addition, concatenation
-	subtraction
*	multiplication
/	division
=	assignment, equality
>	greater-than
<	less-than
>=	greater-than or equal to
<=	less-than or equal to
!=	not equal to
!	logical not
&	logical and
|	logical or
<<	date modification

Symbols

Introduction   Operations are performed on symbols.

The symbols that you can use in Gentran:Server extended rules are:

◗   variables

◗   constants

◗   map components/internal storage

◗   arrays

◗   accumulators.

Uses of symbols   You can use symbols to address existing map components. You can also create additional instances of map components, as long as the map component is originally defined in internal storage.

Example
You can use this function to create extra line items when one line item field is already defined in internal storage.

Symbol syntax   You must address each type of symbol in the proper syntax.

String constant
To address a string constant, you must enclose the constant value in quotes:
#fieldname = "HDR";
where HDR is the constant value

Addressing or creating a field in internal storage
To address a field or create a field in internal storage, within the scope of the current mapping action, use the syntax #FIELD_NAME
#field_1 = 2;
where 2 is a numeric constant value

Addressing or creating a field in a group
To address a field within a group or create a field within a group in internal storage within the scope of the current hierarchy, use the syntax $GROUP.#FIELD_NAME
$N1.#0234

Symbol syntax (contd)

Addressing or creating a group in internal storage

To fully address a group in the entire internal storage area, or to create a group in internal storage, use the syntax $LOOP[index1][index2][index3], where the index entries indicate the hierarchical structure of the loop and enable you to address specific instances of a group:

$Group_C[3][2].#Field_2

where you are specifying the second instance of Group_C within the third instance of Group_B:
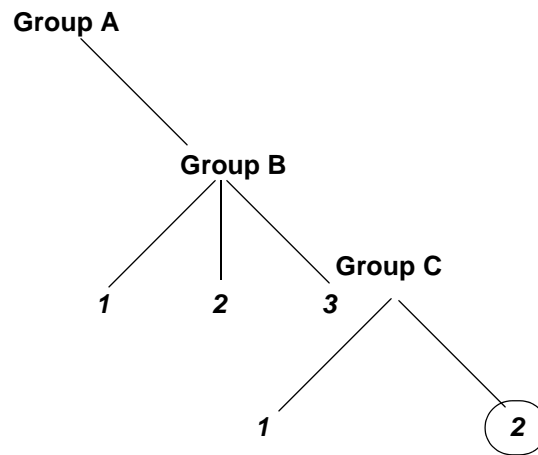
Addressing an array

To address an array (of any type), you address each element of the array individually. For example, if array_1 is an array (of integers) and is declared:

integer array_1[5]

With variables 0 through 4, each element of the array is addressed individually as follows:
array_1[0]
array_1[1]
array_1[2]
array_1[3]
array_1[4]

Accessing an accumulator

You can access an accumulator in the same manner as variables or internal storage. To address an accumulator, use the syntax accum(n) where "n" is the number (not the name) of the accumulator:
accum(2) = 5;

Example of a simple statement

This diagram illustrates an example of a simple statement.
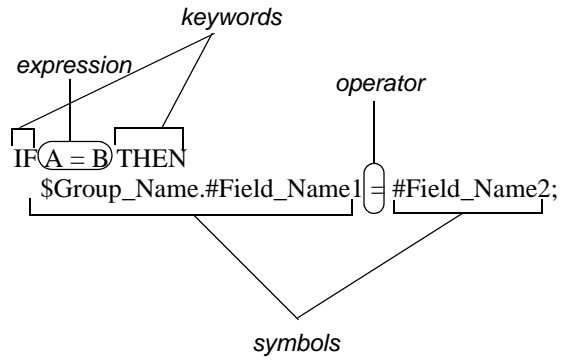
Common Statements and Examples

Overview

Introduction	This section lists examples of some of the common statements.

- ◗ assignment
- ◗ datetime expressions
- ◗ conditional logic
- ◗ string conditions and functions
- ◗ numerical functions
- ◗ raise compliance error function
- ◗ remove field value function
- ◗ existence of data function

Assignment

Introduction

The assignment statement is the most powerful and most often used extended rule statement. In the most simple form, it is written as:

variable=expression

However, you can use this statement in more flexible and complex ways, defined as follows:

numeric_variable=numeric_expression
numeric_field=numeric_expression
string_variable=string_expression
string_field=string_expression
datetime_variable=datetime_expression
datetime_field=datetime_expression

Numeric expression

A numeric expression can consist of numbers, numeric fields, numeric variables, and numeric functions combined with the standard arithmetic operators.

String expression

A string expression can consist of string constants, string fields, string variables, and string functions concatenated with the "+" operator.

Datetime expression

A datetime expression can consist of a datetime constant, datetime field or datetime variable.

Examples

Some examples of assignment expressions are as follows:

```
a = 5;
a = b + c;
s = "hello";
s = s + "world";
```

Datetime Expressions

Introduction   Datetime expressions consist of a datetime variable and (optionally) datetime modifiers. You can write datetime expressions using datetime constants if you are using the standard syntax:

```
year/month/day
hour:minute:second
year/month/day/hour:minute:second
```

You can also write datetime expressions with datetime fields, variables, or by using *date* and *time* functions.

Date syntax   Write date functions as follows. Specify the month (January - December) as 1-12.

```
datetime d;

d = date(1995,4,6);
d = date(1995,4,6,12,0);
d = date("%y/%m/%d", "95/4/6");
```

The **d = date("%y/%m/%d", "95/4/6");** format enables you to convert any string format type into a datetime format type by indicating a format mask (**"%y/%m/%d"**) along with the string (**"95/4/6"**) you want to convert. Use this function if you are using non-standard syntax and need to specify the syntax you are using.

<< operator   You can use the << operator to modify your datetime variable by adding time increments (e.g., days, weeks, years). For example:

```
datetime d;
d=d <<weeks(2);
//This adds 2 weeks to d.
```

Time syntax   Write time functions as follows:

```
d = time(12,0);
d = time(12,0,59);
```

You can use the << operator to modify your datetime variable by adding time increments (e.g., seconds, minutes, years). For example:

```
datetime d;
d=d <<seconds(1);
//This adds 1 second to d.
```

Get and set syntax

The *get* and *set* functions enable you to access (get) or modify (set) individual components of a datetime type. Use these functions as follows:

```
integer a;
datetime d;
a = get days (d);
a = get hours (d);
set hours(d,a);
set days (d,a);
```

Conditional Logic

Introduction

Gentran:Server uses conditional logic to test conditions. Then, depending on the results of the test, Gentran:Server may perform different operations.

Nesting

You can nest conditions to any level.

No terminating syntax

Do not end conditions with a semicolon (;). The semicolon is necessary for terminating statements, but not conditions.

IF...THEN... ELSE

You can use the IF/THEN keywords to execute one or more statements conditionally. The condition is usually a comparison, but it can be any expression that concludes with a numeric value. Gentran:Server interprets the value as either *true* or *false*. The system interprets a zero value as false and a nonzero value as true.

BEGIN/END keywords
If you include more than one statement in the body of an IF/THEN loop, you must surround the statements with the BEGIN/END keywords. If you use only a single statement, you can omit the BEGIN and END.

IF/THEN execution

Gentran:Server evaluates the IF/THEN condition.

◗ If the condition is true, the system executes all the statements that follow the THEN keyword.

◗ If the condition is false, the system does not execute the statements following THEN.

IF...THEN... ELSE

ELSE keyword execution

You can use the ELSE keyword in conjunction with IF/THEN to define several blocks of statements, one of which is executed.

Gentran:Server tests the first IF/THEN condition.

‣ If the condition is false, the system proceeds to test each sequential condition until it finds one that is true.

‣ The system executes the corresponding block of statements for the true condition.

‣ If none of the IF/THEN conditions are true, the system executes the statements following the ELSE keyword.

Syntax
```
IF condition THEN
BEGIN
    statement1;
    statement2;
END
ELSE
BEGIN
    statement3;
    statement4;
END
```

Example of conditional logic

Assume that you need to evaluate whether an N1 or NAD group contains billing or shipping information (this depends on the qualifier that a field in the group contains). Then you will map that information to the appropriate application fields.

For this example, you need to add an On End extended rule to the N1/NAD. The system executes the rule when the group terminates. This is an example of the syntax of the rule:
```
IF #0098 = "BT" THEN
BEGIN
    $Group_Name.#BILLTONAME = #0093;
    $Group_Name. #BILLTOADDR1 = #0166;
    $Group_Name.#BILLTOADDR2 = #0166:2;
    $Group_Name.#BILLTOCITY    = #0019;
    $Group_Name.#BILLTOSTATE = #0156;
    $Group_Name. #BILLTOPCODE = #0116;
END
IF #0098 = "ST" THEN
BEGIN
    $Group_Name.#SHIPTONAME = #0093;
    $Group_Name.#SHIPTOADDR1 = #0166;
    $Group_Name.#SHIPTOADDR2 = #0166:2;
    $Group_Name.#SHIPTOCITY    = #0019;
    $Group_Name.#SHIPTOSTATE = #0156;
    $Group_Name.#SHIPTOPCODE = #0116;
END
```

String Conditions and Functions

Introduction

You can use string conditions in IF/THEN and IF/THEN/ELSE statements to perform comparisons between strings. Here are some examples of the syntax:

```
IF s1 = s2 THEN
IF s1 < s2 THEN
IF s1 > s2 THEN
```

The following string functions are also available for you to use:

◗   left

◗   right

◗   mid

◗   strdate

◗   concat

◗   strstr

Left, right, mid syntax

The *left*, *right*, and *mid* functions enable you to extract substrings from a string.

◗   The left function extracts a specified number of characters from the left of the string variable or field and returns the result as a string.

◗   The right function extracts a specified number of characters from the right of the string variable and returns the result as a string.

◗   The mid functions extracts from a specified position in the string to the right, for a specified number of characters.

This is an example of how the statements are used.

```
string[10] s;
string[3] s1;
string[3] s2;
string[4] s3;
string[7] s4;

s = "abcdefghij";
s1 = left(s,3);
s2 = right(s,3);
s3 = mid(s,3,4);
```

Strdate syntax

The *strdate* function converts a datetime type into a string using a format that you specify. This function enables you to include static characters such as a slash (/), which gives you access to full date support.

Syntax
```
datetime d;
string[8] s;

strdate(d,"%y/%m/%d",s);
```

Concat syntax

The *concat* function concatenates a specified number of characters from one string onto the end of another string. This example demonstrates the syntax to concatenate five characters from string "s2" onto the end of string "s1":

```
string[10] s1,s2;
concat(s1,s2,5);
```

Strstr syntax

The *strstr* function finds a substring inside a string. This function returns the position of the first instance of the designated substring. If this function does not find the specified substring inside the string, it returns a value of -1.

Syntax
```
integer d;

d = strstr("hello", "el");
```

Numerical Functions

Introduction

The numerical functions enable you to convert one data type to another. These are the numerical functions that you can use:

◗ len

◗ atoi

◗ aton

◗ ntoa

Len syntax

The *len* function counts and returns the number of characters in a string.

Example
```
integer a;
a = len("hello");
```

Atoi, aton, ntoa syntax

The *atoi* function converts strings into integers.

The *aton* function converts strings into real numbers.

The *ntoa* function converts both integers and real numbers into strings.

Example
```
integer a;
real b;
string[8] s;
a = atoi("5");
b = aton("5.5");
ntoa(5.5, s);
```

Raise Compliance Error Function (CERROR)

Introduction   The *cerror* function raises a compliance error.

Syntax   `cerror(100,$ILD[0][1][1].#VATC);`

The above example raises compliance error 100 on the VATC element of the specified instance of the ILD group. You typically specify this function as a action to be performed if a condition is false. This function is valid on either side of a map.

Remove Field Value Function (EMPTY)

Introduction   The *empty* function sets the value of a field in internal storage to null. This function is not the same as setting the value of a field to a zero length string (" ") or to zero. This function is typically used to prevent an output value.

Syntax   `empty($ILD[0][1][1].#VATC);`

The above example sets the value of the specified instance of the VATC element to null.

Note that you cannot use EMPTY as part of an expression because it does not return a value.

Existence of Data Function (EXIST)

Introduction   The exist function returns a non-zero (true) value if there is data in a specified field in internal storage. If there is no data in the specified field, this function returns a zero (false) value. This function is typically used as a part of a condition.

Syntax   `IF exist($ILD[0][1][1].#VATC) THEN`

The above example returns a non-zero value if the condition is true (data is present in the specified instance of the VATC element). A zero value is returned if the condition is false (no data is present in the specified instance of the VATC element).

Count Function (COUNT)

Introduction   The *count* function returns the total number of times a group (or repeating record) has iterated.

Syntax   `count(GROUPNAME[3][*]);`

The above example returns the total iterations of the GROUPNAME group within the third iteration of the parent group.

Delete Function (DELETE)

Introduction   The *delete* function deletes the specified iteration of a repeating record or group.

Syntax   `delete(GROUPNAME[iteration]);`

File Pointer Functions (FSEEK, FTELL)

Introduction   The *fseek* function moves the file pointer to a new location which is a specified number of bytes (offset) from the beginning of the file, end of the file, or relative to the current position (origin). To invoke the fseek function against the input file, the value for current_file is "0." To invoke the fseek function against the output file, the value for current_file is "1." The fseek function is typically used in conjunction with only the ftell, readblock, and writeblock functions.

The *ftell* function obtains the current position of the file pointer and returns it as an integer. To invoke the ftell function against the input file, the value for current_file is "0." To invoke the ftell function against the output file, the value for current_file is "1." The fseek function is typically used in conjunction with only the fseek, readblock, and writeblock functions.

Syntax
```
fseek(current_file,offset,origin);
numeric_variable = ftell(current_file);
```

Block of Data Functions (READBLOCK, WRITEBLOCK)

Introduction   The *readblock* function reads a block of data (segment or record) from the input file and places it into the argument of a string variable. The *writeblock* function writes the data contained in the argument of a string variable to the output file.

The readblock and writeblock functions are used in conjunction with each other to pass a block of data from the input file to the output file without compliance checking or testing for proper EDI syntax. Together these functions provide a more efficient alternative of using wildcard segments.

Syntax
```
readblock(string_variable);
writeblock(string_variable);
```

User Exit Function (EXEC)

What is a user exit?

A user exit is an extended rule that enables the map to temporarily exit translation and then:

◗ Run a shell script, retrieve the results, return to translation, and use the results of the shell script in translation. In Gentran:Server for UNIX, the exec extended rule has this function.

Placing the extended rule

You can use the exec function in an extended rule for a map component at any hierarchical level, including field level extended rules, if appropriate.

Note

When you apply a user exit to a map component, subordinate map components may also be able to execute the same user exit.

Allowable data types

This table lists the extended rule data types that you can use with exec:

◗ INTEGER
◗ REAL
◗ STRING

Note

You cannot use date and time data unless you process it as a string.

Alphabetic Language Reference

Overview

Introduction   This section contains an alphabetic reference for the Gentran:Server proprietary programming language keywords and commands. This enables you to quickly and easily refer to information on specific keywords or commands.

Typeface conventions   This table lists the typeface conventions used in this section.

IF the typeface follows this example...	THEN it is used to indicate...
While	what you need to type.
condition	in syntax, information that you need to provide.
[1]	that, in syntax, items inside square brackets are optional.

Typeface conventions   The following programming guidelines are utilized in this section:

◗   Keywords and commands are shown in all lower case letters.

◗   Map components are shown in all upper case letters.

◗   A double slash (//) introduces comments.

◗   Lines that are too long to fit on one line in this section may be continued on the next line using a line-continuation character (Â).

atoi

Introduction

The **atoi** function is a numerical function that converts strings into integers. The numerical functions enable you to convert one data type to another.

Syntax

int = atoi(string);

where:int=integer variable
string=string variable

Example

```
integer a;
string[20] s;
s = "5";
a = atoi(s);
// "a" contains the value 5
```

aton

Introduction

The **aton** function is a numerical function that converts strings into real numbers. The numerical functions enable you to convert one data type to another.

Syntax

real = aton(string);

where:real=real number variable
string=string variable

Example

```
real b;
string[20] s;
s = "5.5";
a = aton(s);
// "a" contains the value 5.5
```

begin

Introduction

The **begin** and **end** keywords are used in conjunction with **if/then** loops. You can nest the conditions in an **if/then** loop to any level. If you include more than one statement in the body of an **if/then** loop, you must surround the statements with the **begin/end** keywords. If you use only a single statement, you can omit the **begin** and **end**.

Note
Do not end conditions with a semicolon (;). This terminating syntax is necessary for statements *only*.

Syntax

```
if condition then
begin
    statement1;
    statement2;
end
```

break

Introduction
The **break** function terminates the execution of the nearest enclosing **while** loop and passes control to the statement that follows the **end** keyword. The **break** keyword can be used in complex loops to terminate a loop before several statements have been executed.

Example

```
integer i
integer j;

i = j = 0;
while i<10 do
begin
    if (i = 8) then
        continue;
    if (i = 9) then
        break;
end
//As long as "i" has a value less than "10" the loop repeats. If "i"
//has
//a value of "i", the loop continues. If "i" has a value
//of "9" the loop terminates.
```

cerror

Introduction

The *cerror* function raises a compliance error. You can specify this function as a action to be performed if a condition is false.

Syntax

cerror(error_number,$GROUP_NAME[index][index][index].
Â#FIELD_NAME);

Example

```
cerror(100,$ILD[0][1][1].#VATC);
//This raises compliance error 100 on the VATC element of the
//specified instance of the ILD group
```

Compliance error codes

The compliance error codes are as follows:

Error Number	Error Generated
100	Mandatory Element
105	Conditional Element
110	Incorrect Element Format
111	Invalid Data
112	Invalid Time
113	Invalid String
114	Invalid Number
115	Incorrect Element Length
116	Element Too Short
117	Element Too Long
118	Invalid Overpunch Data (NCPDP)
119	Invalid Numeric Data
120	Too Many Components
	(Continued on next page)

(Contd) Error Number	Error Generated
121	Too Many Elements
125	Element Delimiter
130	Conditional Relation
135	Element Code
140	Implicit Rule Failure
145	Binary Substitution Failed
200	Mandatory Component
205	Conditional Component
210	Incorrect Component Format
215	Incorrect Component Length
220	Component Delimiter
225	Component Code
300	Mandatory Segment
305	Segment Sequence
310	Invalid Start End
315	Incorrect Block Format
320	Segment Terminator
330	Unexpected Segment
400	Session Error
405	Unknown Trading Partner
410	Invalid Control Number
415	Invalid Sub Count
420	Unknown Relationship
425	Unknown Standard
430	Unknown Version
435	Unknown Transaction
	(Continued on next page)

(Contd) Error Number	Error Generated
440	Duplicate Control Number
445	Duplicate Doc Name

**XML compliance
error codes**

The XML compliance error codes are as follows:

Error Number	Error Generated
600	XML Element Type Match
601	XML Element Root Type
602	XML Element Required
620	XML Attribute Undeclared
621	XML Attribute Not Unique
622	XML Attribute Required
623	XML Attribute Fixed Error
624	XML Attribute ID Not Unique
625	XML Attribute ID REF Match
630	XML Attribute Parse Name Error
631	XML Parse Equal Error
632	XML Parse O Quote Error
633	XML Parse C Quote Error
640	XML Attribute Name Error
641	XML Attribute Names Error
642	XML Attribute Nmtoken Error
643	XML Attribute Nmtokens Error
644	XML Attribute Entity Error
645	XML Attribute Entities Error

(Contd) Error Number	Error Generated
660	XML Reference Character Invalid
661	XML Reference Entity Recursion
662	XML Reference Entity Undeclared
663	XML Reference Invalid Usage
699	XML Unknown Error

Database compliance error codes

The Database compliance error codes are as follows:

Error Number	Error Generated
700	ODBC Data Source Open Error
701	ODBC Data Source Rollback
702	ODBC Data Source Commit Error
703	ODBC Data Source Rollback Error
710	ODBC Query Open Error
711	ODBC Command Error
712	ODBC Cursor Error
720	ODBC Output Table Error
721	ODBC Output Operation Error
730	ODBC Mandatory Element
731	ODBC Data Conversion Error
732	ODBC Incorrect Element Format
733	ODBC Implicit Rule Failure
799	ODBC Unknown Error

NCPDP compliance error codes

The NCPDP compliance error codes are as follows:

Error Number	Error Generated
800	NCPDP Batch File Header Segment - Missing
801	NCPDP Batch File Header Segment - Segment ID Error
802	NCPDP Batch File Header Segment - Transmission Type Error
803	NCPDP Batch File Header Segment - Sender ID Missing
804	NCPDP Batch File Header Segment - Batch ID Error
805	NCPDP Batch File Header Segment - File Type Error
806	NCPDP Batch File Header Segment - Receiver ID Missing
807	NCPDP Batch File Detail Data Record - Missing
808	NCPDP Batch File Detail Data Record - Segment ID Error
809	NCPDP Batch File Detail Data Record - Transaction Reference Missing
810	NCPDP Batch File Trailer Record - Missing
811	NCPDP Batch File Trailer Record - Segment ID Error
812	NCPDP Batch File Trailer Record - Batch ID Error
813	NCPDP Batch File Trailer Record - Record Count Error
814	NCPDP Batch File Trailer Record - Record Count Error Mismatch Error
815	NCPDP Data Record - Repeating Segment Error
816	NCPDP Data Record - Transaction Code Error
817	NCPDP Data Record - Transaction Count Error
818	NCPDP Data Record - Transaction Count Mismatch Error
819	NCPDP Data Record - Header Response Code Error
820	NCPDP Data Record - Transaction Response Code Error
830	NCPDP Null Character

(Continued on next page)

(Contd) Error Number	Error Generated
832	NCPDP Invalid Character
833	NCPDP Unexpected End-of-File
835	NCPDP Unexpected STX (start of text)
836	NCPDP Unexpected ETX (end of text)
837	NCPDP Missing Stream Separator 1
838	NCPDP Missing Stream Separator 2
839	NCPDP Unexpected Separator
840	NCPDP Unexpected Field Separator
841	NCPDP Unexpected Group Separator
842	NCPDP Missing Group Separator
845	NCPDP Positional Segment Tag Off
855	NCPDP Delimited Segment Empty
856	NCPDP Delimited Segment Missing First Field Separator
857	NCPDP Delimited Field ID Invalid
858	NCPDP Delimited Field ID Invalid Character
859	NCPDP Delimited Field ID Null Character
860	NCPDP Delimited Field ID Incomplete
861	NCPDP Delimited Field Data Missing
862	NCPDP Delimited Field Data Null Character
863	NCPDP Delimited Field Data Invalid
864	NCPDP Delimited Field Data Invalid Character
865	NCPDP Delimited Segment ID Invalid
866	NCPDP Delimited Segment ID Invalid Character
867	NCPDP Delimited Segment ID Null Character
868	NCPDP Delimited Segment ID Incomplete *(Continued on next page)*

(Contd) Error Number	Error Generated
869	NCPDP Delimited Field ID Size
885	NCPDP Batch - Nothing After STX
886	NCPDP Batch - Unexpected STX
887	NCPDP Batch - Missing ETX
888	NCPDP Batch - Data File Open
890	NCPDP Data Field String Separator
891	NCPDP Data Field String No Filename
892	NCPDP Data Field String Start Byte
893	NCPDP Data Field String End Byte
894	NCPDP Data Field String Incomplete
895	NCPDP Data Field String Extra Characters
896	NCPDP Data Field Size Incorrect
897	NCPDP Data Field Start Out of Bounds
898	NCPDP Data Field Start Beyond End
899	NCPDP Unknown Error

concat

Introduction

The **concat** function concatenates a specified number of characters from one string onto the end of another string.

Syntax

concat(string,string,num_char);

where:string=string variable
num_char=number of characters from the second string onto the
end of the first string

Example

```
string[10] s1,s2;
concat(s1,s2,5);
//Concatenate five characters from string "s2" onto the end of
//string "s1"
```

continue

Introduction   The **continue** function continues the execution of the innermost loop without
processing the statements in the loop that follow the **continue** statement.

Example
```
integer i
integer j;
i = j = 0;
while i<10 do
begin
     if (i = 8) then
          continue;
     if (i = 9) then
          break;
end
//As long as "i" has a value of "8" the innermost loop repeats. If
//"i"
//does not have a value of "8" the next statement (break statement)
//is processed.
```

count

Introduction   The **count** function counts and returns the number of iterations of a group.

Example   This is an example of the count function:

```
integer i;
i = count($N1[*]);
//The [*] is a wildcard that counts the number of iterations of the
//N1 group.
```

delete

Introduction   The **delete** function deletes a specified occurrence of a group.

Syntax   delete($GROUP_NAME[N]);

where [n] is the occurrence of the group that you want to delete.

Example
```
delete($ILD[2]);
//Deletes the second occurrence of the ILD group.
```

empty

Introduction	The **empty** function sets the value of a field in internal storage to null. This function is not the same as setting the value of a field to a zero length string (" ") or to zero.

Syntax	empty($GROUP_NAME[index][index][index]. #FIELD_NAME)

Example

```
empty($ILD[0][1][1].#VATC);
//Set the value of the specified instance of the VATC element
//to null
```

end

Introduction

The end and begin keywords are used in conjunction with **if/then** loops. You can nest the conditions in an **if/then** loop to any level. If you include more than one statement in the body of an **if/then** loop, you must surround the statements with the **begin/end** keywords. If you use only a single statement, you can omit the **begin** and **end**.

Note
Do not end conditions with a semicolon (;). This terminating syntax is necessary for statements *only.*

Syntax

```
if condition then
begin
    statement1;
    statement2;
end
```

exec

Introduction   The exec function invokes the execution of a shell script. The translator waits until the script finishes before continuing with translation.

After the script runs and returns a numeric return code, the exec function:

◗   Retrieves the return code

◗   Returns to translation

◗   Uses the return code in translation.

Note
You must set an integer value equal to the return value of the exec (...) call for the rule to compile.

Syntax   nReturn = exec(string)

where:
nReturn = return value
string = string variable or literal value that represents the shell script

Example
```
integer nReturn;
nReturn = 0;
nReturn = exec ("c:\addrunm.sh");
```

exist

Introduction **Exist** tests for empty fields (containing null). The **exist** function returns a non-zero (true) value if there is data in a specified field in internal storage. If there is no data in the specified field, this function returns a zero (false) value. This function is typically used as a part of a condition.

Syntax if exist($GROUP_NAME[index][index][index]. #FIELD_NAME) then

Example
```
if exist($ILD[0][1][1].#VATC) then
//Return a non-zero value if the condition is true (data is present
//in the
//specified instance of the VATC element). A zero value is returned
//if the condition is false (no data is present in the specified
//instance of
//the VATC element).
```

fseek

Introduction

The **fseek** function moves the file pointer to a new location, which is a specified number of bytes (offset) from the beginning of the file, end of the file, or relative to the current position (origin). To invoke the fseek function against the input file, the value for current_file is "0." To invoke the fseek function against the output file, the value for current_file is "1." The fseek function is typically used in conjunction with only the ftell, readblock, and writeblock functions.

Syntax

fseek(current_file,offset,origin);

where:

current_file	0 indicates the input file; 1 indicates the output file
offset	The position to which the file pointer is moved relative to the origin.
origin	Use **begin** to offset the file pointer from the beginning of the file, **end** to offset from the end of the file, or **current** to offset from the pointer's current position in the file.

Example

```
string[1024]temp_buffer;
Integer Position;
Position = ftell(0)
while readblock(temp_buffer) do
begin
   if left(tem_buffer,3) = "IEA" then
       begin
             fseek(0,Position,begin);
              break;
          end
   writeblock(tem_butffer);
    Position = ftell(0);
end
//Read segment from input file and place in temp_buffer. Look for
//"IEA" segment tag. If found, reset file pointer to where it was
//before the "IEA" segment was read. Write contents of temp_buffer
//to output file. Set "Position" = the current file pointer
//position.
```

ftell

Introduction   The **ftell** function obtains the current position of the file pointer and returns the position as an integer. To invoke the ftell function against the input file, the value for current_file is "0." To invoke the ftell function against the output file, the value for current_file is "1." The fseek function is typically used in conjunction with only the fseek, readblock, and writeblock functions.

Syntax   numeric_variable = ftell(current_file);

Example
```
string[1024]temp_buffer;
Integer Position;
Position = ftell(0)
while readblock(temp_buffer) do
begin
   if left(tem_buffer,3) = "IEA" then
       begin
             fseek(0,Position,begin);
              break;
          end
   writeblock(tem_butffer);
    Position = ftell(0);
end
//Read segment from input file and place in temp_buffer. Look for
//"IEA" segment tag. If found, reset file pointer to where it was
//before the "IEA" segment was read. Write contents of temp_buffer
//to output file. Set "Position" = the current file pointer
//position.
```

get

Introduction

The **get** function enables you to access individual components of a datetime variable.

Syntax

integer_variable = get datetime_component (datetime_variable);

where:integer_variable=integer variable
datetime_component=individual component of the datetime
variable
datetime_variable=datetime variable of which you want to
access a component part

Example

```
integer a;
integer b;
datetime d;
a = get days (d);
b = get hours (d);
//Accesses the days from the datetime variable "d" and loads into
//variable "a". Accesses the hours from the datetime variable "d"
//and loads into variable "b".
```

if...then...else

Introduction   You can use the **if**, **then**, and **else** keywords to apply conditional logic. Use conditional logic to have Gentran:Server test conditions and then, depending on the results of the test, perform different operations.

Note
Do not end conditions with a semicolon (;). This terminating syntax is necessary for statements *only*.

if/then   You can use the **if/then** keywords to execute one or more statements conditionally. The condition is typically a comparison, but it can be any expression that concludes with a numeric value. Gentran:Server interprets the value as either *true* or *false*. A zero value is false and a nonzero value is true.

Gentran:Server evaluates the **if/then** condition, and if it is true, the system executes all the statements that follow the **then** keyword. If the condition is false, none of the statements following **then** are executed.

else   You can use the **else** keyword in conjunction with **if/then** to define several blocks of statements, one of which is executed. Gentran:Server tests the first **if/then** condition. If the condition is false, the system proceeds to test each sequential condition until it finds one that is true. The system executes the corresponding block of statements for the true condition. If none of the **if/then** conditions are true, the system executes the statements that follow the **else** keyword.

Nesting conditions   You can nest conditions to any level.

Example
```
if condition then
begin
    statement1;
    statement2;
end
else
begin
    statement3;
    statement4;
end
```

index

Introduction   The **index** function enables you to determine which instance of a particular loop the translator is currently accessing.

Syntax   index(integer_variable);

where:integer_variable=integer variable that indicates the hierarchical level for which you want to determine the loop count

Example
```
index(1)
//This extended rule is located on the ILD group in the diagram
//below. This determines the current loop count for the first level
//(ODD) in the hierarchical structure.
```

INVOIC -M- INVOICE DETAILS
ODD -M- 999999
ILD -M- 999999

left

Introduction

The **left** function extracts a specified number of characters from the left side of a string variable or field and returns the result as a string.

Syntax

The syntax is as follows:

string_variable = left(string_variable,num_char)

where:num_char=integer variable

Example

```
string [25]name;
string [5]temp_variable;
name = "Acme Shipping Company"
temp_variable = left(name,4);
// "temp_variable" would contain "Acme"
```

len

Introduction   The **len** function is a numerical function that counts and returns the number of characters in a string. The numerical functions enable you to convert one data type to another.

Syntax   number_char = len(string);

where:num_char=integer variable

Example
```
integer a;
a = len("hello");
// "a" contains the value 5
```

messagebox

Introduction   The **messagebox** function enables you to display on the monitor a message box for which you have designated the format and content. You can specify the number and type of buttons on the message box, the message icon (for example, hand, question mark, exclamation point, or asterisk), and the message displayed. You can also issue a return value based on the chosen action, such as a Yes or No response from the user. This function is useful for debugging maps.

Syntax   messagebox("message",defined_number)

where:message=message string
defined_number=defined number of the desired buttons plus the
defined number of the desired icon (if used)

Example
```
if messagebox("Do you really want to delete this object?",36) = 6
begin
.
.
.
end

//Displays a message box with the given string as the message, Yes
//and No buttons (4) and a question mark icon (32). The number and
//type of buttons (4) plus the icon (32) equals the defined_number
//(36). If the user clicks the Yes button (return value "6"), the
//statements in the begin/end loop are executed.
```

**Defined_
numbers**   This table lists the defined_numbers for the button and icon types.

Defined_Number	Button or Icon Types
0	**OK** button only
1	**OK** and **Cancel** buttons
4	**Yes** and **No** buttons
16	Icon Hand
32	Icon Question Mark
	(Continued on next page)

Defined_Number	Button or Icon Types
48	Icon Exclamation Point
64	Icon Asterisk

Message box return values

This table lists the message box return values.

Return Value	Action Selected
1	**OK** selected
2	**Cancel** selected
6	**Yes** selected
7	**No** selected

mid

Introduction

The **mid** function extracts characters from a specified position in a string, either to the end of the string or for a specified number of characters, and returns the resultant string. This function is zero-based.

Syntax

string_variable = mid(string_variable,start_pos,num_char)

where:start_pos =integer variable
num_char=integer variable

Example

```
string [25]name;
string [10]temp_variable;
name = "Acme Shipping Company"
temp_variable = mid(name,6,8);
// "temp_variable" will contain "Shipping"
```

ntoa

Introduction The **ntoa** function is a numerical function that converts real numbers into strings. The numerical functions enable you to convert one data type to another.

Syntax string = ntoa(real,string);

where:real=real number variable
string=string variable

Example
```
real b;
string[8] s;
b = 5.5;
ntoa(5.5, s);
//The variable "s" contains the string "5.5".
```

param

Introduction

The **param** function is used to read the value of the PARAM(n) variable. This extended rule enables you to pass data strings between the core translator and the lftran translator program. You can run this rule from any scope (pre-session, post-session, group onbegin, field, and so on).

Use to get TP Code

In Gentran:Server for UNIX, param(1) is used exclusively to store the Trading Partner Code. The translator copies the TP Code to param(1) so the TP Code is available to be used in the map.

Reference
See the tp.dat/idx topic of the File Record Layouts chapter in the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for the layout of the trading partner record.

Syntax

param(integer_PARAM_number);

where: integer_PARAM_number = the number of the pre-defined variable

Example

```
//populate field #0351 with value of Param(1)
#0351 = param(1)

//field #0351 then contains the TP Code
```

readblock

Introduction   The **readblock** function reads a block of data (segment or record) from the input file and places it into the argument of a string variable.The readblock and writeblock functions are used in conjunction with each other to pass a block of data from the input file to the output file without compliance checking or testing for proper EDI syntax. Used together, these functions provide a more efficient alternative to using wildcard segments.

Syntax   readblock(string_variable);

Example
```
while readblock(temp_buffer) do
begin
   if left(tem_buffer,3) = "IEA" then
       begin
             fseek(0,Position,begin);
              break;
         end
   writeblock(tem_butffer);
    Position = ftell(0);
end
//Read segment from input file and place in temp_buffer. Look for
//"IEA" segment tag. If found, reset file pointer to where it was
//before the "IEA" segment was read. Write contents of temp_buffer
//to output file. Set "Position" = the current file pointer
//position.
```

readbytes

Introduction   The **readbytes** function reads a number of bytes from the input file. This function is used in conjunction with the writebytes function. Together, the readbytes and writebytes function provide an efficient method to pass data through a map if the data does not need to be compliance checked or altered in any way.

Readbytes is similar to the readblock function. However, readblock works with entire blocks only, such as an entire segment or record. Readbytes works with any quantity of data, whether it is smaller or larger than a block.

The readbytes function uses two parameters:

◗   The string variable into which the data being read will be stored

◗   The number of bytes to read

Syntax   readbytes(read_from_buffer, num_bytes);

where:read_from_buffer=string variable into which the data being read will be stored
num_bytes=integer value representing the number of bytes to read
from the input file.

Note
Readbytes returns the number of bytes it was actually able to read.

Example
```
string [1024] tempBuffer;
while readbytes(tempBuffer,1024) do
begin
writebytes(tempBuffer,1024);
End
//Read 1024 bytes from input file and place in string variable
//named tempBuffer.

writebytes("^0D^0A",2);
//Appends a CRLF to the end of the output file.
```

right

Introduction   The **right** function extracts a specified number of characters from the right side of a string variable or field.

Syntax   string_variable = right(string_variable,num_char)

where:num_char=integer variable

Example
```
string [25]name;
string [10]temp_variable;
name = "Acme Shipping Company"
temp_variable = right(name,7);
// "temp_variable" would contain "Company"
```

set

Introduction   The **set** function enables you to define individual components of a datetime variable.

Syntax   set datetime_component (datetime_variable,integer_variable);

where:datetime_component=individual component of the datetime variable
datetime_variable=datetime variable of which you want to access a component part
integer_variable=integer variable

Example
```
integer a;
integer b;
datetime d;
set days (d,a);
set hours (d,a);
//Defines the days of the datetime variable "d" from variable "a".
//Defines the hours of the datetime variable "d" from variable "b".
```

setparam

Introduction   The **setparam** function writes a value to a PARAM(n) variable. This enables you to pass data strings between the core translator and the outer translator program (lftran). You can run this rule from any scope (pre-session, post-session, group onbegin, field, and so on).

Use in audit record   In Gentran:Server for UNIX, PARAM(0) is used exclusively to store a mapped value in the audit record. After translation, the translator copies the content of PARAM(0) into the User Buffer field of the EDI audit record, edihist.dat/.idx. If the contents of PARAM(0) exceed 53 characters, the translator copies only the first 53 characters.

Reference
See the edistat.i/o, edihist.dat/idx topic of the File Record Layouts chapter in the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for the layout of the audit record.

Syntax   setparam(integer_PARAM_number, string_data);

where: interger_PARAM_number = the number of the pre-defined variable, string_data = the value of the data assigned to the parameter

Example   In the following example, #0324 is a field in the map. If the content of field #0324 is a PO number, such as PO152893, then parm 0 contains the string "The PO for this order is: PO152893."

```
string [64] strMsg;

// build the message to pass back to lftran
strMsg = "The PO Number for this order is: ";
strMsg = strMsg + #0324;

// populate user param 1 with this message
setparam(0,strMsg);

// when translation is complete lftran will
// extract the message we created from user
// param 0.
```

strdate

Introduction   The **strdate** function converts a datetime type into a string using a format that you specify. This function allows you to include static characters such as a slash (/), which gives you access to full date support.

Syntax   strdate(datetime,"format",string);

where:datetime=datetime variable (month specified as 0-11)
format=desired date format
string=string variable

Example
```
datetime d;
string[8] s;

strdate(d,"%y/%m%d",s);

//Converts a datetime variable into an eight character string in
//the format "year/month/day".
```

Format specifiers   This table lists the format specifiers.

Format Specifier	Description
%a	Abbreviated weekday name
%A	Full weekday name
%b	Abbreviated month name
%B	Full month name
%d	Day of the month as a decimal number (01 – 31)
%H	Hour in 24-hour format (00 – 23)
%I	Hour in 12-hour format (01– 12)
%j	Day of the year as a decimal number (001 – 366)

(Contd) Format Specifier	Description
%m	Month as a decimal number (01 – 12)
%M	Minute as a decimal number (00 – 59)
%S	Second as a decimal number (00 – 59)
%U	Week of the year as a decimal number, with Sunday as the first day of the week (00 – 51)
%w	Weekday as a decimal number (0 – 6, with Sunday as "0")
%W	Week of the year as a decimal number, with Monday as the first day of the week (00 – 51)
%y	Year without the century as a decimal number (00 – 99)
%Y	Year with the century as a decimal number
%%	Percent sign
%8	ISO-8601 date format YYYYMMDD'T'HHMMSS'.'sssZ **Note** Because %8 represents the entire date string, you cannot combine %8 with any other format specifier.

strstr

Introduction   The **strstr** function finds a substring inside a string. This function returns the position of the first instance of the designated substring within the specified string. If this function does not find the specified substring in the string, it returns a value of "-1". This function is zero-based.

Syntax   integer = strstr("string","substring");

where:integer=integer variable
string=string
substring=part of the string

Example
```
integer d;

d = strstr("mississippi","is");

//Finds the first instance of the substring "is" inside the string
//"mississippi" and returns the position of that first substring.
```

unreadblock

Introduction

The **unreadblock** function moves the input file-pointer back one block. (A block of data is equivalent to one EDI standard segment or one positional record.) Unreadblock is used with the readblock function to "unread" the block of data that readblock just processed. The unreadblock function unreads only the most recent block of data processed.

Note

Use unreadblock in conjunction with the readblock function only.

Unreadblock is an alternative to using the fseek and ftell functions, and is the preferred method of moving the file pointer back one block of data. Unreadblock allows the translator to correctly track the number of bytes read and number of segments read during the translation process by moving the file-pointer back and decrementing the segment and bytes counts accordingly.

Syntax

unreadblock();

Example

```
string[1024] tempBuffer;

while readblock(tempBuffer) do
begin
if left(tempBuffer,3) = "IEA" then
begin
unreadblock();
break;
end
writeblock(tempBuffer);
end

//Read segment from input file and place in tempBuffer variable.
//Look for the "IEA" segment tag and, if found, reset the file-
//pointer to where it was located before the "IEA" segment was
//read. Write contents of tempBuffer to output file.
```

while...do

Introduction

The **while...do** function executes a statement repeatedly until the specified termination condition evaluates to zero. The system tests the terminating condition *before* each iteration of the loop, so a **while** loop executes zero or more times depending on the value of the termination expression.

Note
Do not end conditions with a semicolon (;). This terminating syntax is necessary for statements *only*.

Example

```
integer i;

while i < 10 do
begin
      if (i = 8) then
          continue;
      if (i = 9) then
          break;
end
//While "i" is less than ten, execute the loop. If "i" is equal to
//or greater than ten, terminate the loop.
```

writeblock

Introduction

The **writeblock** function writes the data contained in the argument of a string variable to the output file.The readblock and writeblock functions are used in conjunction with each other to pass a block of data from the input file to the output file without compliance checking or testing for proper EDI syntax. Together these functions provide a more efficient alternative of using wildcard segments.

Syntax

writeblock(string_variable);

Example

```
while readblock(temp_buffer) do
begin
   if left(tem_buffer,3) = "IEA" then
      begin
            fseek(0,Position,begin);
             break;
         end
   writeblock(tem_butffer);
    Position = ftell(0);
end
//Read segment from input file and place in temp_buffer. Look for
//"IEA" segment tag. If found, reset file pointer to where it was
//before the "IEA" segment was read. Write contents of temp_buffer
//to output file. Set "Position" = the current file pointer
//position.
```

writebytes

Introduction   The **writebytes** function writes a specified number of bytes to the output file. You use this function in conjunction with the readbytes function. Used together, the readbytes and writebytes function provide an efficient method of passing data through a map if the data does not need to be compliance checked or altered.

Note
The system does not append a segment terminator to the end of the string value written to the output file.

Writebytes is similar to the writeblock function. However, writeblock works only with entire blocks, such as an entire segment or record. Writebytes works with any quantity of data, whether it is smaller or larger than a block.

The writebytes function uses two parameters:

◗   The string variable containing the data to be written to the output file

◗   The number of bytes to write

Syntax   writebytes(write_to_buffer, num_bytes);

where:write_to_buffer=string variable containing the data to be written to the output file
num_bytes=integer value representing the number of bytes to read
from the write_to_buffer variable and write to the
output file

Note
If hex values are used in constants, use \0x instead of ^.

Example
```
string [1024] tempBuffer;
while readbytes(tempBuffer,1024) do
begin
writebytes(tempBuffer,1024);
End
//Read 1024 bytes from input file and place in string variable
//named tempBuffer.

writebytes("^0D^0A",2);
//Appends a CRLF to the end of the output file.
```

Working with Trading Partnerships

Contents

Overview

Introduction

In this chapter   This chapter describes the procedures for maintaining your Trading Partnership files and records.

Key Terms   This table lists the key terms used in this chapter.

Term	Description
category	A user-definable record that enables you to group Trading Partnerships.
contact record	A record containing the name, address, and telephone numbers of an individual at your trading partner's business.
Group Organization record	A record containing all information specific to a single division or department in your trading partner's organization.
Interchange Organization record	A record containing all information specific to a single company.
Reconciliation ID record	A record containing a set of interchange and group IDs used with functional acknowledgments.
Standard Cross-Reference Table	The Gentran:Server feature that enables you to build a table of standard values. Gentran:Server can use these values to find the Trading Partner record for certain inbound documents.
TRADACOMS record	A record containing all of the supplementary Trading Partnership information for use with TRADACOMS standards.
	(Continued on next page)

(Contd) Term	Description
Trading Partner record	One of the records maintained in trading partner files: ◗ Interchange Organization record ◗ Group Organization record ◗ Trading Partnership record ◗ Contact record ◗ Reconciliation ID record ◗ Category record ◗ TRADACOMS record.
Trading Partnership	An arrangement with a specific trading partner to exchange information in a specific document type and using a particular standard version.
Trading Partnership code	A code you define that uniquely identifies a Trading Partnership record.
Trading Partnership record	A record containing information about one of the Trading Partnerships you have established.

Trading Partnership Records

Introduction

Trading Partnership records hold information about your trading partner. Gentran:Server uses the information you define in the Trading Partnership records every time it processes your data. During translation, Gentran:Server inserts the information defined in the Trading Partnership records into the stream of data that you are sending your trading partner. Specifically, Gentran:Server uses this information when building the Interchange envelope, the Group envelope, and the transaction set envelopes.

This section describes all Trading Partnership records and examines the relationships among them.

Understanding Trading Partnership records

Trading Partnership records are those records you maintain with the Trading Partnership Editor. The following records define a trading Partnership:

◗ Interchange Organization record

◗ Group Organization record

◗ Trading Partnership record

◗ Category record

◗ Reconciliation ID record

◗ Contact record

The system hierarchically defines the Trading Partner records. At the highest level is the Interchange Organization record. Below the interchange level is the Group Organization record. At the lowest level is the Trading Partnership record.

To translate data in Gentran:Server, you must define each of the following for every document that you want to process:

◗ One Interchange Organization record

◗ One Group Organization record

◗ One Trading Partnership record.

Depending on your organization's needs, you may also define the Category, Reconciliation ID, and Contact for the Trading Partnership.

Interchange Organization record	The Interchange Organization record contains information that is used when Gentran:Server creates the interchange envelope around your EDI data. Interchange Organization records must be unique for each interchange-level organization. In other words, you must have a separate Interchange Organization record for each interchange-level organization.

Group Organization record	The Group Organization record contains information that Gentran:Server uses when creating the group envelope around your EDI data. You may create multiple group records for a single interchange record. Group Organization records must be unique for each group-level organization. In other words, there must be a separate Group Organization record for each group-level organization.

Trading Partnership record	The Trading Partnership record contains basic Trading Partnership information such as the Trading Partnership Code, the compiled map (translation object) Gentran:Server will use when translating business documents for this partner, and whether the system will generate an acknowledgment. Trading Partnership records need not be unique for each Trading Partnership. In other words, two or more trading partnerships can use the same Trading Partnership record.

Category record	The Category record is a record you create that allows you to group trading partnerships according to similarities of interest. You may create up to five category types for use with your trading partnerships.

Reconciliation ID record	The Reconciliation ID record contains Reconciliation IDs for use in functional acknowledgments and CONTRLs. If the Interchange IDs and Group or Application IDs in your functional acknowledgments or CONTRLs are different than those used in the document being acknowledged, you must have a Reconciliation ID record in addition to a Trading Partnership record for each Trading Partnership. Reconciliation ID records must be unique for each Trading Partnership.

Contact record

You use contact records to store the name, address, and telephone number of a contact at your trading partner's place of business. You can create separate Contact records at the interchange, group, and trading partner level.

Contact records must be unique for each individual with whom you communicate in your trading partner's organization.
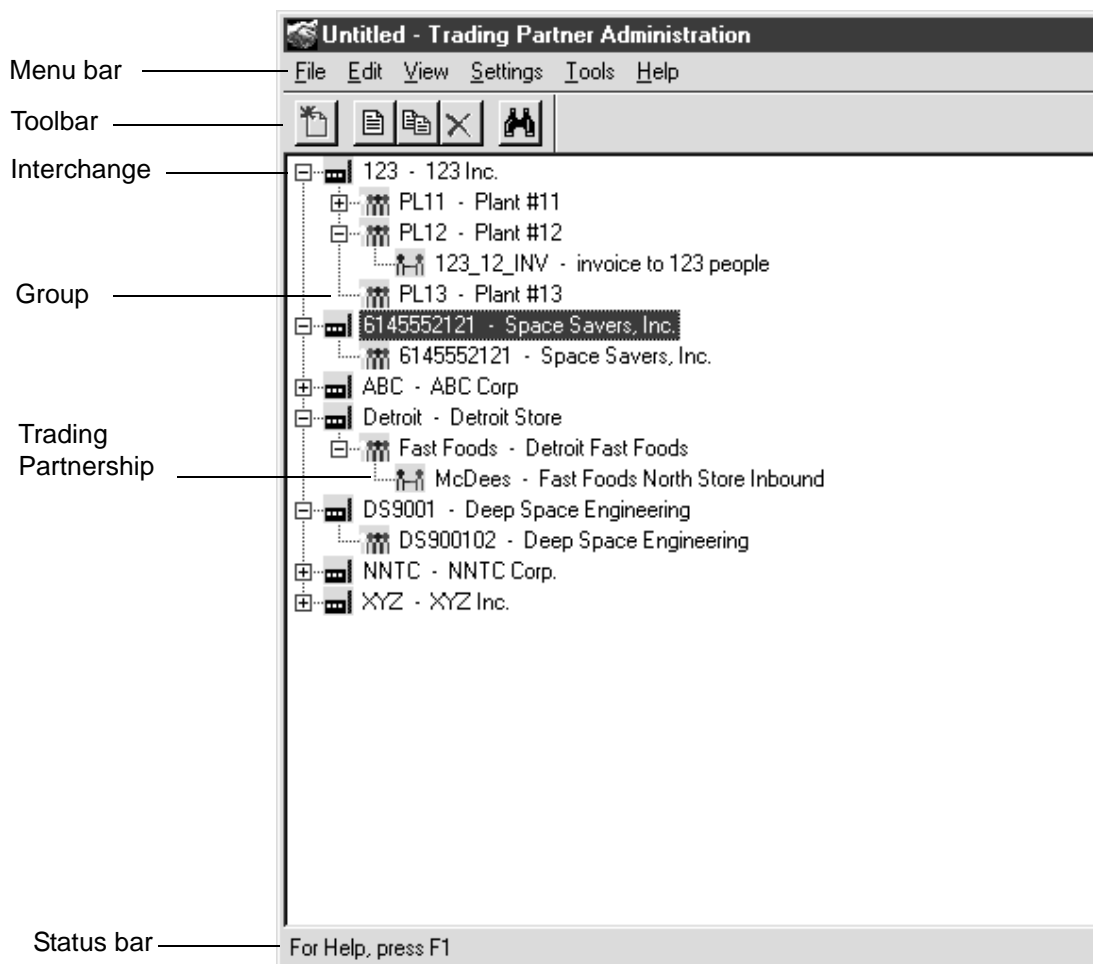
Trading Partnership Administration

The Trading Partner Administration Explorer

Introduction

You carry out most day-to-day activities involving your Trading Partnership records through the **Trading Partner Administration** explorer.

Trading Partner Administration explorer

This illustration shows an example of the **Trading Partner Administration** explorer for Gentran:Server Workstation.

Host and environment name

In the Gentran:Server for UNIX version, the title bar displays the host name and environment name.

Trading Partner Administration explorer fields and functions

This table describes the fields and functions of the **Trading Partner Administration** explorer window.

Field	Function
Menu bar	Opens menus to display commands.
Toolbar	Starts another function, such as the Save or Delete function.
Interchange	Identifies an Interchange Organization record, which contains all information specific to a single company
Group	Identifies a Group Organization record, which contains all information specific to a single division or department in your trading partner's organization.
Trading Partnership	Identifies a Trading Partnership record, which contains information about one of the trading partnerships you have established.
Status bar	Displays a status message, prompt, or toolbar button description.

Viewing options

The Trading Partnership Administration window has four viewing options:

◗ No tree
◗ Trading Partnership records only
◗ Organization records only
◗ Entire tree

Reference
See the How to Select a Trading Partner View topic in this chapter for information about selecting a viewing option.

Right mouse button

You can click the right mouse button (right-click) to display a shortcut menu to use with **Trading Partnership Administration**.

This illustration shows the shortcut menu that appears when you click the right mouse button.
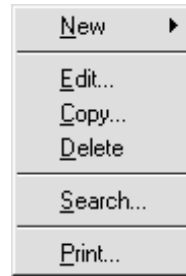
How to Access Trading Partnership Administration

Opening the Trading Partnership Administration window

Use this procedure to open the **Trading Partnership Administration** window.

Step	Action
1	Open Gentran:Server.
2	Click **Edit** on the **Main Menu**.
3	Click **Trading Partnership** on the **Edit** menu. **System Response**

IF…	THEN…
This is the first time anyone has opened Trading Partnership Administration	Gentran:Server displays the Preferences dialog box.
Your preferences for Trading Partnership Administration have been set	Gentran:Server displays the Trading Partnership Administration window.

Step	Action
4	Use this table to determine your next step.

IF Gentran:Server displays the…	THEN go to…
Preferences dialog box.	How to Set TP Administration Preferences in this chapter.
Trading Partnership Administration window.	How to Select a Trading Partner View in this chapter.

Note

You can also access the Trading Partnership Administration window by clicking the Trading Partnership Maintenance button on the menu bar of the Main Menu.

How to Select a Trading Partner View

Overview

This section explains the four views available in Trading Partnership Administration.

Introduction

You choose a view in Trading Partnership Administration based on these criteria:

◗ The number of Trading Partnership records you want to display.

◗ The number of records you maintain.

◗ Whether you have assigned Interchange and Group Organization codes to your Trading Partnership records.

Four viewing options

This table lists the available views and their functions.

View…	Description…
No Tree	An empty dialog box.
TP Only	A list of Trading Partnership records.
Org Only	A list of all Organization codes. Use when you have a large number of records.
Entire Tree	A complete tree view that includes a list of all Organization and Trading Partner records. Includes records that are not defined for any Organization code.

No Tree

When you select **No Tree** view, the system displays a blank screen.

Use this view for these tasks:

◗ Searching for and editing a single Trading Partnership Record.

◗ Creating new Interchange Organizations, Group Organizations, and Trading Partnership records.

◗ Performing all functions quickly.

Illustration   This illustration shows the Trading Partnership Administration using the **No Tree** view.
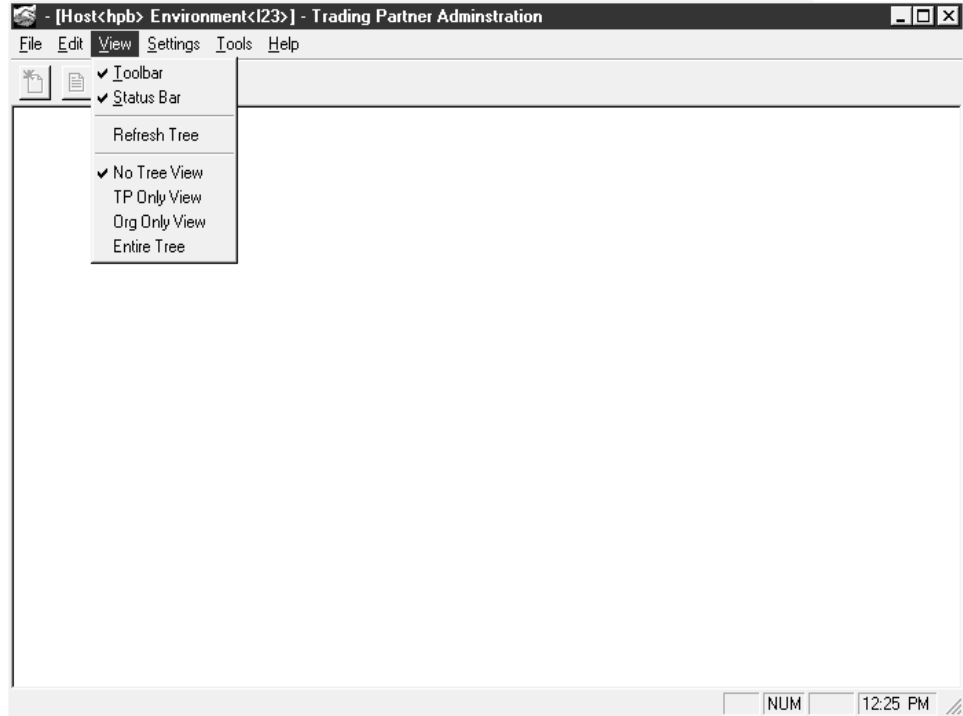
TP Only   When you select **TP Only** view, the system displays a list of all Trading Partnership records.

Note
The **TP Only** view displays only Trading Partnership records, not Interchange or Group Organization codes.

Illustration   This illustration shows the Trading Partnership Administration using the **TP Only** view.
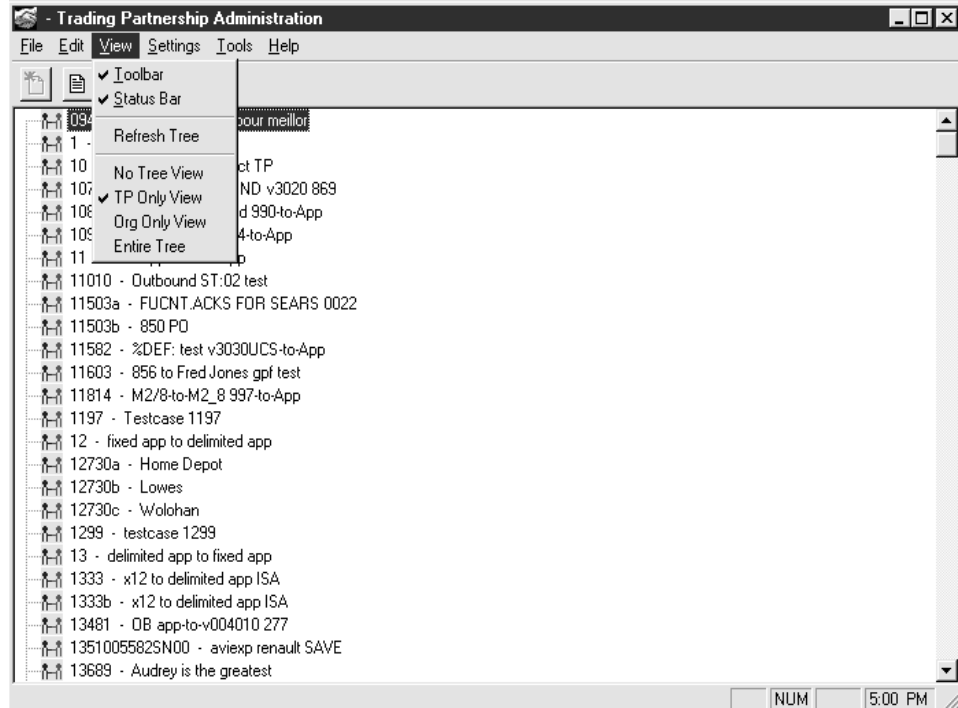
Org Only   When you select **Org Only** view, the system displays a tree containing all available Interchange Organization codes and those Group Organizations not associated with an Interchange Organization.

Note
Initially, this view does not display the Trading Partnership records.

Procedure   When you select an Interchange Organization code, Trading Partnership Administration expands the tree to display the Group Organizations.

Use this procedure to select a Trading Partnership Record.

Step	Action
1	Click on the appropriate Interchange Organization code to select it.
2	Select the appropriate Group Organization code.
3	Select the Trading Partnership record.

Note
You can use this view to display only those Trading Partnership records that are assigned to an Interchange Organization code.

Illustration   This illustration shows the Trading Partnership Administration using the **Org Only** view.
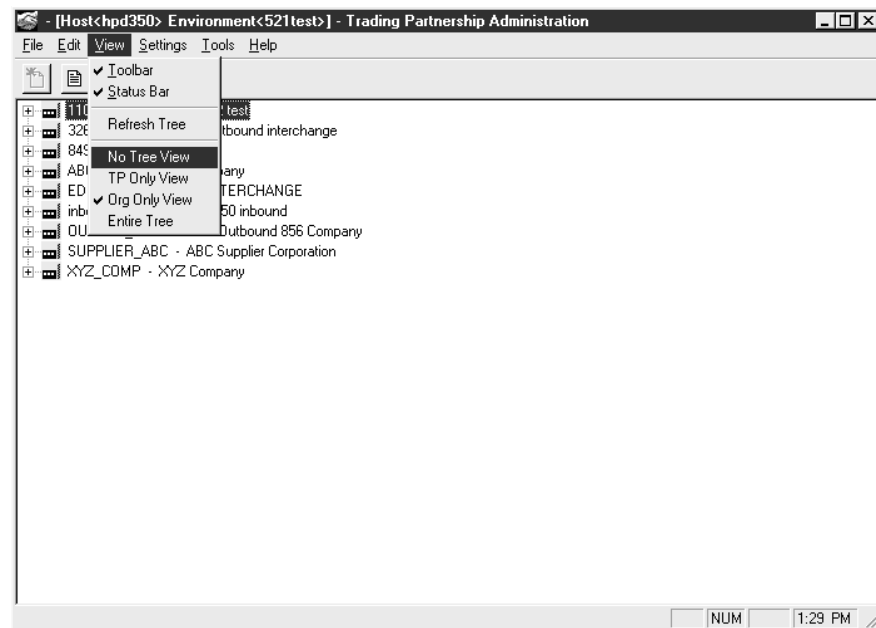
Entire Tree view

When you select the **Entire Tree** view, the system displays all Interchange and Group Organization codes and all Trading Partnership records.

Note

When you select an Interchange Organization code, the system displays the related Group Organization codes. When you select a Group Organization code, the system displays the related Trading Partnership records.

Illustration

This illustration shows the Trading Partnership Administration using the **Entire Tree** view.
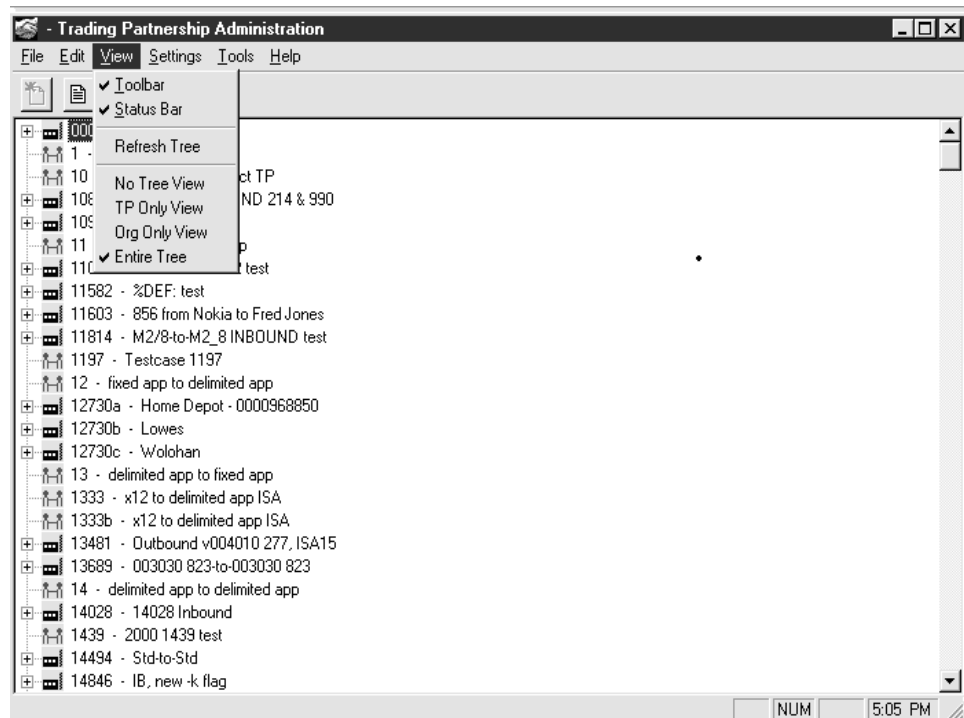
How to Set TP Administration Preferences

Overview
The default view of the Trading Partnership Administration window is initially set to **Build Entire Tree.**

Note
You can select another view as the default using the Trading Partnership Administration Preference dialog box by clicking **Preferences** on the **Settings** menu. However, if you are using Gentran:Server for UNIX, the Security Administrator must grant you the authority to change the setting.

Illustration
The illustration shows the Trading Partnership Administration Preferences dialog box.
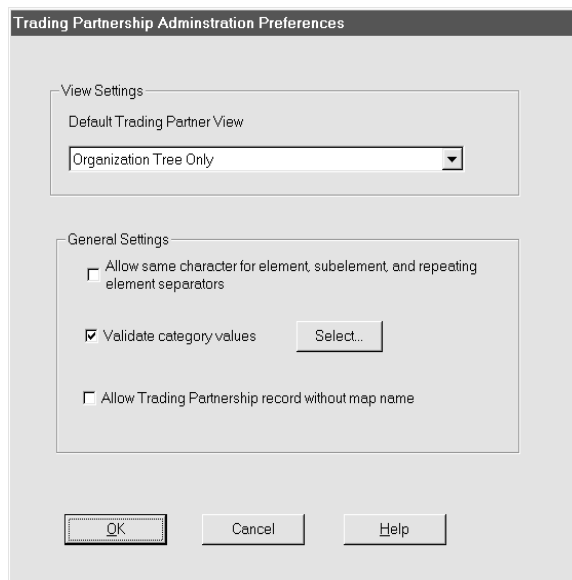
Setting the default view
To set another view as the default, change the value of the **Default Trading Partner view** text box by clicking the down arrow and selecting one of the four available views:
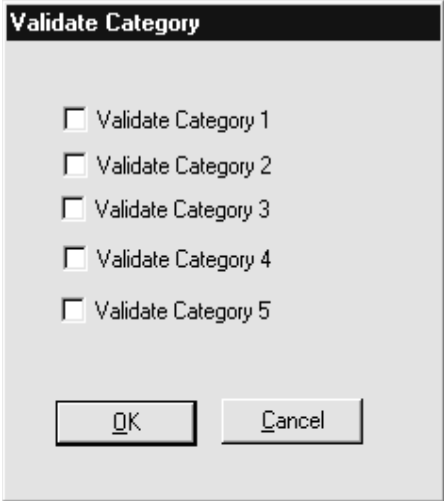
◗   No Tree

◗   Trading Partnership Tree Only

◗   Organization Tree Only

◗   Build Entire Tree

General Settings   You need to define the General Settings prior to creating and editing the Trading Partnership Record.
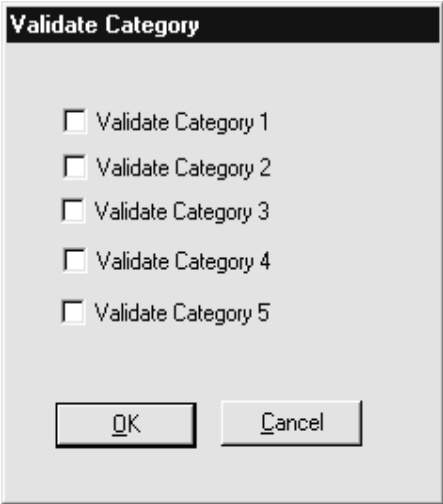
Field	Function
Allow same character for element, subelement, and repeating element separators	Disables the warning prompt displayed when the same character is used for element, subelement, and repeating element separators. Also allows the same values.
Validate category values	Validates the selected category against the database. **Note** Check the Validate category values and choose **SELECT** to display five categories. You can verify up to five categories in every Trading Partnership Record.
Allow Trading Partnership record without map name	Disables the warning displayed whenever you create a Trading Partnership record without assigning a map name to it, allowing you to create a Trading Partnership record without assigning a map name. **Note** If you try to create a Trading Partnership record without assigning it a map name and this option is not selected, the system displays an error message.

Selecting categories to validate

Follow this procedure to display and use the Validate Category dialog box.

Step	Action
1	Select the **Validate category values** option under General Settings.
2	Click **Select.** **System Response** The system displays the **Validate Category Window**.
3	Select the categories you want to validate.
4	Click **OK.** **System Response** The system returns to the Trading Partnership Administration Preferences dialog box.

How to Set Default Translation Selections

Introduction

You can set default (automatically assigned) translation selections for the Trading Partnership records you create. You set these choices for the type of translation (application-to-standard, application-to-application, etc.).
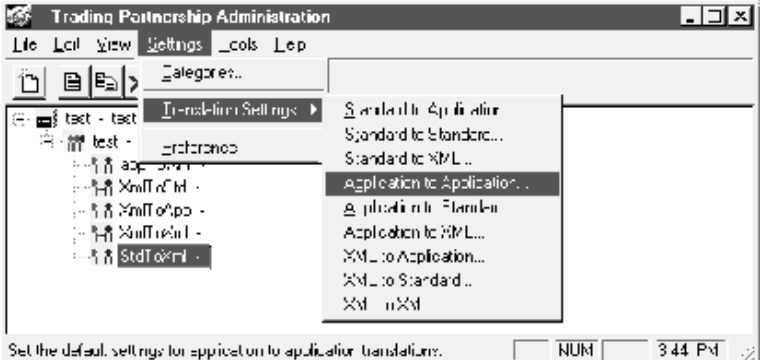
How the system uses the selections

Once you set default translation selections, Gentran:Server applies these selections to each subsequent Trading Partnership record you create that has the specified translation type. If you do not want a Trading Partnership to use the default settings, you can set them at the Trading Partnership level to override the default selections.
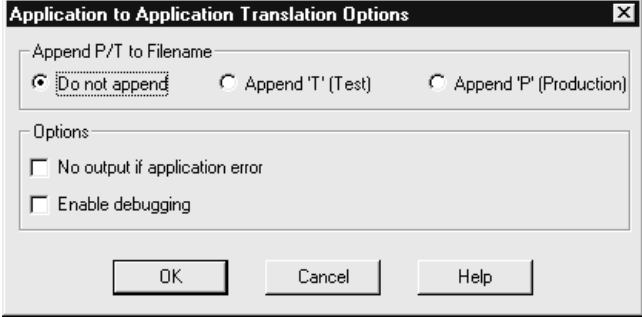
Reference
See How to Set Translation Selections for instructions on setting translation selections for a specific Trading Partnership.

When to use

Use this procedure to set the translation selections that you normally use for one or more specific types of translation.

Procedure

Use this procedure to set default translation selections for a translation type.

Step	Action
1	Click **Translation Settings** from the **Settings** menu.
2	Select the translation type you want to set from the submenu. (Continued on next page)

(Contd) Step	Action
	System Response The system displays the Translation Options dialog box for the translation type you selected. **Example** ![Application to Application Translation Options dialog box] **Reference** See the Translation Options Dialog Boxes topic for descriptions of the translation options.
3	Select the default selections that you want to apply to this type of translation.
4	Click **OK** to save your changes.

How to Search for a Trading Partnership Record

Overview
This topic explains how to search for a Trading Partnership record. You can search for Trading Partnership records by ranges of values in various fields. You can also search for specific values such as:

◗   Interchange or Group IDs

◗   Organization codes

◗   Trading Partnership Codes

◗   Standard versions

◗   Set IDs, categories, and name.

You enter the search criteria on the Trading Partner Search dialog box. You can run Trading Partner Search on all views except Trading Partner Only view.

Trading Partner Search dialog box example
This illustration shows the Trading Partner Search dialog box.

Fields and functions

This table lists the fields of the Trading Partnership Search dialog box and their functions.

Field	Function
Enter a range of trading partnership codes	Searches for Trading Partnerships with Trading Partnership codes that fall within the specified range.
Target EDI interchange and/or group IDs	Searches for Trading Partnerships with the interchange or group IDs you specify. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.
Target a specific standard version	Searches for Trading Partnerships using a specific EDI standard version. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.
Target message/transaction set ID	Searches for Trading Partnerships with documents that have message or set ID you specify. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.
Target one or more trading partnership categories	Searches for Trading Partnerships based on a category you have defined.
Target a specific translation mapping table	Searches for Trading Partnerships using a specific compiled map or translation object (TPL table).
Specify the trading partnership name/ description	Searches for a Trading Partnership by name.
Target a specific translation type	Searches for Trading Partnerships using one of the valid translation types. The **Edit** button displays a list of choices.

Note

You can select multiple fields when searching for Trading Partnership records. Each field that you select narrows the search and enables you to focus on a specific Trading Partnership.

Trading Partner Search Results dialog box

After you enter the search criteria, Gentran:Server displays the Trading Partner Search Results dialog box. This dialog box displays a list of all Trading Partnership records that meet the criteria. You then select the record that you want to view or edit from the list. This illustration shows an example of the Trading Partner Search Results dialog box.
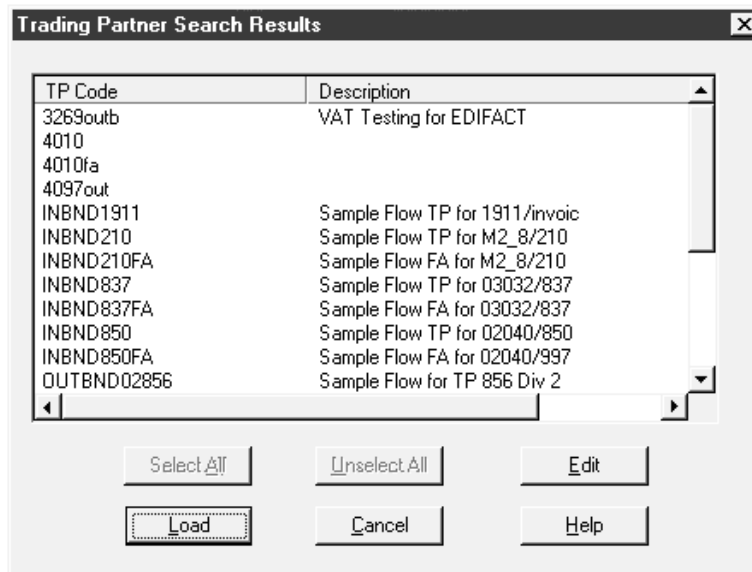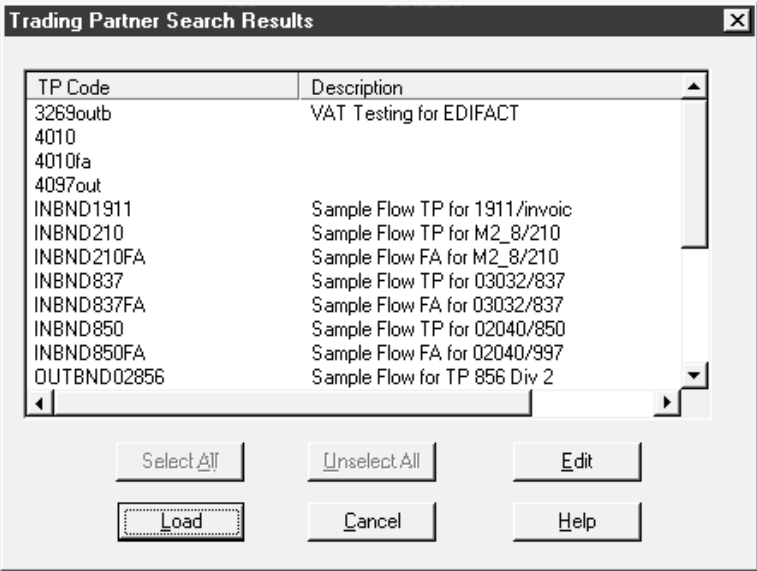

Fields and functions

This table lists the fields of the Trading Partner Search Results dialog box and their functions.

Field	Function
TP Code	Lists the Trading Partnership IDs that match your search criteria.
Description	Displays the Trading Partnership descriptions that match your search criteria.

(Continued on next page)

Procedure   Follow this procedure to search for a Trading Partnership record.

Step	Action
1	Open Trading Partner Administration.
2	Select **Search** from the **Edit** menu or click on the **binocular** icon. **System Response** Trading Partnership Administration displays the Trading Partner Search dialog box.
3	Select your search criteria by clicking the check box in front of the criteria you want to use.
4	Click the **Edit** button to display a dialog box for entering your search criteria.
5	Do you want to narrow your search by selecting more criteria? ◗　If YES, repeat Steps 3 and 4 to select additional search criteria. ◗　If NO, continue with Step 6. (Continued on next page)

(Contd) Step	Action
6	Click **OK** to search your Trading Partnership records. **System Response** The Trading Partner Search Results dialog box displays the list of records that match your search criteria. (Continued on next page)

(Contd) Step	Action	
7	Use this table to determine your next step.	
	IF you want to…	**THEN…**
	Select all the Trading Partnerships in the list	Click **Select All**.
	Unselect all the Trading Partnerships in the list	Click **Unselect All**.
	Edit the Trading Partnership record	Select the Trading Partner Code from the list and click **Edit**.
	Display the Interchange code associated with the Trading Partnership record	Select the Trading Partner Code from the list and click **Load**.
	Load a Trading Partner record that is not associated with an Interchange Organization Record to the tree view	Select the Trading Partner Code from the list and click **Load**.
	Close the Trading Partner Search Results dialog box	Click **Cancel**.

The Trading Partnership Editor

Introduction   You use the Trading Partnership Editor to create and maintain your Trading Partnership records. When you create a new Trading Partnership record, the Trading Partnership Editor guides you through the information that you need to define for the type of translation you select. You also use the Trading Partnership Editor to alter and update Trading Partnership information in existing records.

Accessing the Trading Partnership Editor dialog box   Use this procedure to open the **Trading Partnership Editor**.

Step	Action
1	Open the **Trading Partnership Administration** window.
2	Right-click on the Trading Partnership record you want to edit. **System Response** The system displays a menu of available functions.
3	Click **Edit**. **System Response** The system displays the Trading Partnership Editor dialog box.

Trading Partnership Editor tabs   These are the Trading Partnership Editor tabs:

◗ Trading Partnership
◗ Inbound EDI Information
◗ Outbound EDI Information
◗ Runtime Information
◗ Archive Information
◗ Inbound Acknowledgment
◗ Outbound Acknowledgment
◗ Categories.

Note
The Trading Partnership Editor displays only the tabs that apply to the Trading Partnership record. You probably will not see all eight tabs for every record.

Trading Partnership tab

This illustration shows an example of the Trading Partnership tab of the Trading Partnership Editor dialog box.

Fields and functions

This table lists the fields of the Trading Partnership tab of the Trading Partnership Editor dialog box and their functions.

Field	Function
Trading Partnership Code	Defines a unique identifier for the Trading Partnership and document combination. The translator uses this code when creating the segment envelope. ◗ Size: 1-15 alphanumeric characters ◗ No spaces or special characters allowed ◗ Suggestion: Use abbreviation of trading partner's name and Set ID (for example, DYNO810).
Trading Partnership Description	Provides a brief description of the Trading Partnership. (maximum of 30 characters) (Continued on next page)

(Contd) Field	**Function**
Translation Type	Defines the type of translation for the Trading Partnership. **Note** The **Translation Type** box is view only. You can select a translation type only when creating a new Trading Partnership.
Map Name	Defines the compiled map (translation object) to use when translating data for the Trading Partnership. Only translation objects (TPL files) are valid selections. **Note** To search for a specific compiled map, click the **Map Search** icon to the right of the **Map Name** text box. **Reference** See the topic How to Choose a Different Map (Translation Object) in this chapter for more information.
Map description	Displays the Gentran:Server product and version used to generate the map.
File Definitions filename	Specifies the file definition (.DDF file) that the map uses if the input file is an application file. If the input file is not an application file, this box is disabled. The browser associated with this box displays all the file definitions that reside in the File Definitions directory. **Note** This field links the Trading Partnership record to the file definition. You must complete this box when the input side of the map is an application format.

Inbound EDI Information tab

This illustration shows an example of the Inbound EDI Information tab on the Trading Partnership Editor dialog box.

Fields and functions

This table lists the fields of the Inbound EDI Information tab of the Trading Partnership Editor dialog box and their functions.

Field	Function
Standard version	Lists the available EDI standards and versions. This is a key field that Gentran:Server uses to identify the Trading Partnership record. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.
Document ID	Lists the available EDI document types that you may translate. This is a key field that Gentran:Server uses to identify the Trading Partnership record. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.
Accept Messages/Sets with error(s)	Enables you to accept messages or transaction sets containing compliance errors from your trading partner.
Control Numbers	
Interchange	Initializes the counter used for control numbers for business documents that you receive from your trading partner. **Note** If you click **Maintain globally** on the Runtime Information tab, this field is unavailable.
Validate sequence	Checks the Interchange control numbers to ensure that they are in sequence.
Group	Initializes the counter used for control numbers for business documents that you receive from this trading partner. **Note** If you click **Maintain globally** on the Runtime Information tab, this field is unavailable. (Continued on next page)

(Contd) Field	Function
Validate sequence	Checks the Group control numbers to ensure that they are in sequence.
Message/Set	Initializes the counter used for control numbers for business documents that you receive from this trading partner.
Validate sequence	Checks the Message or Transaction Set control numbers to ensure that they are in sequence.
File generation	
Validate sequence	Checks the File generation control numbers to ensure that they are in sequence.
Xref Table	Displays the Standard Cross Reference dialog box.

Outbound EDI tab

This illustration shows an example of the Outbound EDI tab of the Trading Partnership Editor dialog box.

Fields and functions

This table lists the fields of the Outbound EDI tab of the Trading Partnership Editor dialog box and their functions.

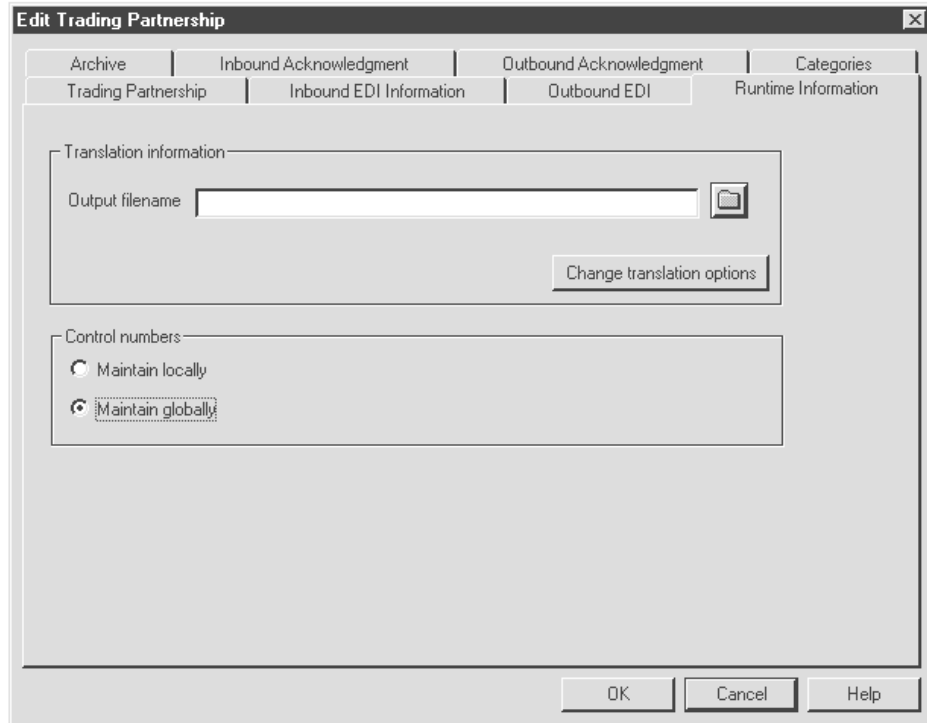| Field	Function
Standard version	Lists the available EDI standards and versions. This is a key field that Gentran:Server uses to identify the Trading Partnership record. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.
Document ID	Lists the available EDI document types that you may translate. This is a key field that Gentran:Server uses to identify the Trading Partnership record. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.
Element separator	Defines the character used to separate elements. **Note** You can use the same character for the element and subelement separators if the **Allow same character for element, subelement and repeating element separators** option on the Trading Partnership Administration Preferences dialog box is checked.
Component/sub-element separator	Defines the character used to separate component or sub-elements. **Note** You can use the same character for the element and subelement separators if the **Allow same character for element, subelement and repeating element separators** option on the Trading Partnership Administration Preferences dialog box is checked.
Segment terminator	Defines the character used to indicate the end of a segment. (Continued on next page)

(Contd) Field	Function
Interchange control header	Defines the type of header segment used for the Interchange envelope. **Reference** See the How to Modify the Outbound Envelope Segments topic in this chapter for instructions on editing the elements in the Interchange envelope segment. **Warning** You MUST click **OK** to save your changes or the segment will be blank.
Group control header	Defines the type of group segment Gentran:Server uses for the Functional Group envelope. **Reference** See the How to Modify the Outbound Envelope Segments topic in this chapter for instructions on editing the elements in the group envelope segment.
Decimal mark	Defines the ASCII hexadecimal character that serves as the decimal point in your EDIFACT data.
Release character	Defines the character that allows a separator or terminator to be used as part of the EDIFACT data. **Example** Your element separator is a plus sign (+). The name of your trading partner is Toys+Joys. The release character prevents the translator from incorrectly interpreting the number of elements in this segment by allowing the plus sign to pass through as data and not as an element separator.
Syntax level	Defines the level A or B separators to use in your EDIFACT data.
Generate UNA	Enables you to create an EDIFACT UNA segment with user defined separators.
UNH --- Association assigned code	Defines the value for the UNH 0057 field, which is mandatory for EANCOM.
Enable enveloping	Restricts outbound EDI data to a single interchange's worth of data in a file.
Generate Messages/ Sets with errors	Enables you to create output that contains errors.

Runtime Information tab

This illustration shows an example of the Runtime Information tab of the Trading Partnership Editor dialog box. The options displayed on this tab depend on the type of translation you selected.

Fields and functions

This table lists the fields of the Runtime Information tab of the Trading Partnership Editor dialog box and their functions.
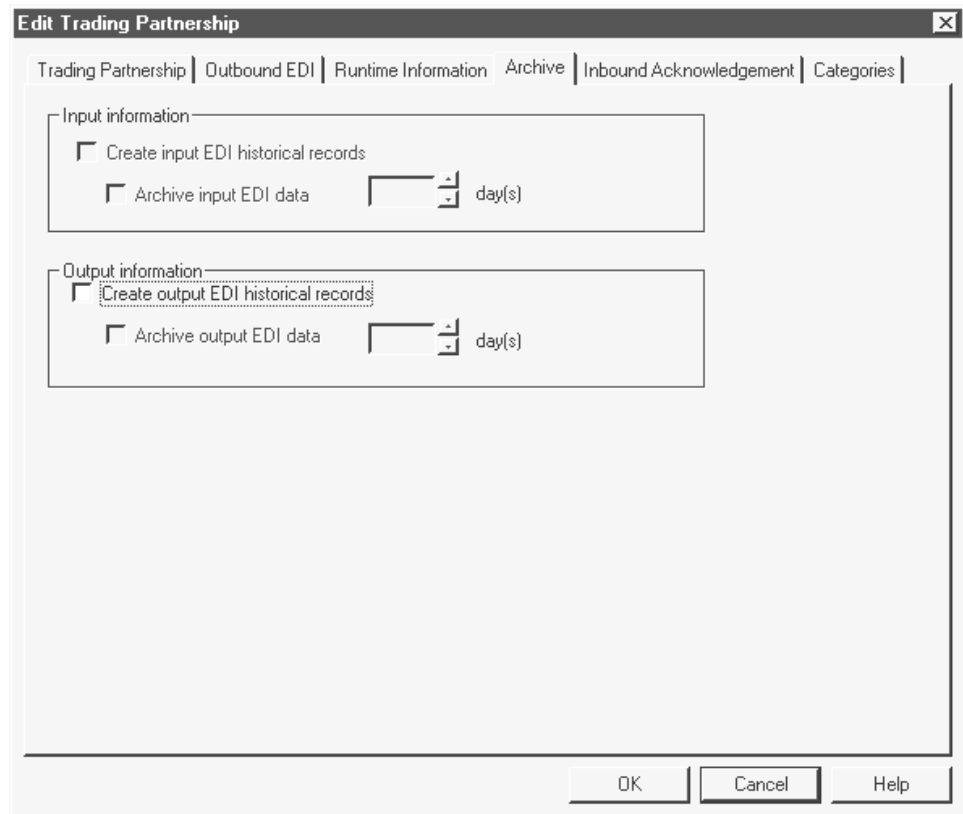
Field	Function
Output filename	Defines the name of the file in which Gentran:Server will store the translated data.
Change translation options	Displays the Translation Options dialog box, which enables you to override (for this Trading Partnership) the default translation option that appends or does not append T or P to the file name.
	(Continued on next page)

(Contd) Field	Function
Maintain locally	Enables you to maintain control numbers at the Trading Partnership level. This is the default setting.
	Note
	The Maintain globally/locally options are displayed when the output file is an EDI standard.
Maintain globally	Enables you to maintain control numbers for an entire organization.
	Note
	The Maintain globally/locally options are displayed when the output file is an EDI standard.

Archive tab   This illustration shows an example of the Archive tab of the Trading Partnership Editor dialog box.

Fields and functions

This table lists the fields of the Archive tab of the Trading Partnership Editor dialog box and their functions.

Field	Function
Create input EDI historical records	Enables you to store translation status information for inbound EDI data.
Archive input EDI data	Enables you to store both the translation status information for your inbound EDI data as well as the translated data itself.
Day(s)	Selects the number of days to store archived data (up to a maximum of 365 days). **Note** The default is 365 days unless you set another value in the Archive/Retrieval Setup dialog box.
Create output EDI historical records	Enables you to store translation status information for your outbound EDI data.
Archive output EDI data	Enables you to store both the translation status information for your outbound EDI data as well as the translated data itself.
Day(s)	Selects the number of days to store archived data (up to a maximum of 365 days). **Note** The default is 365 days unless you set another value in the Archive/Retrieval Setup dialog box.

Inbound Acknowledgent tab

This illustration shows an example of the Inbound Acknowledgment tab of the Trading Partnership Editor dialog box.

Fields and functions

This table lists the fields of the Inbound Acknowledgment tab of the Trading Partnership Editor dialog box and their functions.

Field	Function
Settings	
Expect acknowledgment for outbound document in:	Enables you to define a time limit for receipt of an inbound acknowledgment.
Day(s)	Selects the number of days until receipt of an inbound acknowledgment is expected.
Hour(s)	Selects the number of hours until receipt of an inbound acknowledgment is expected.
	(Continued on next page)

(Contd) Field	Function
Minute(s)	Selects the number of minutes until receipt of an inbound acknowledgment is expected.
Second(s)	Selects the number of seconds until receipt of an inbound acknowledgment is expected.
Specify alternate IDs to reconcile inbound acknowledgments	Enables you to define alternate Interchange and Group IDs to be used when reconciling inbound acknowledgments.
Interchange IDs	
Yours	Defines your alternate Interchange ID. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.
Partner's	Defines your Trading Partner's alternate Interchange ID. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.
Group IDs	
Yours	Defines your alternate Group ID. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.
Partner's	Defines your Trading Partner's alternate group ID. **Note** The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for this field.

Outbound Acknowledgment tab

This illustration shows an example of the Outbound Acknowledgment tab of the Trading Partnership Editor dialog box.

Fields and functions

This table lists the fields of the Outbound Acknowledgment tab of the Trading Partnership Editor dialog box and their functions.

Field	Function
Create acknowledgment for inbound document	Tells Gentran:Server to create an acknowledgment for an inbound document.
Type	Defines the type of acknowledgment you will send to your trading partner. **Notes** In an X12 standard document, the type of acknowledgment is either a 997 or 999 acknowledgment. In an EDIFACT standard message, the type of acknowledgment is a CONTRL message. (Continued on next page)

(Contd) Field	Function
Level	Defines the level of detail used in the acknowledgment. ◗ **Group** - your trading partner will receive response information for each group in the interchange envelope. ◗ **Set** - your trading partner will receive response information for each set in the group envelope. ◗ **Segment** - your trading partner will receive response information for each segment contained in the data. **Reference** See the <u>Setting Up Your System to Handle Functional Acknowledgments</u> topic in the Archiving Translation Data chapter in the *Gentran:Server for UNIX and Workstation Application Integration User's Guide* .
Output filename	Defines the file name of the translated acknowledgment you are sending to your trading partner. **Example** /opt/gentran/data/outbound.997 *(Continued on next page)*

(Contd) Field	Function
Separator/terminator information	
Element separator	Defines the character used to separate elements. **Notes** You can use the same character for the element and subelement separators if the **Allow same character for element, subelement and repeating element separators** option on the Trading Partnership Administration Preferences dialog box is checked.
Component/Sub-elem. separator	Defines the character used to separate component or sub-elements. **Notes** You can use the same character for the element and subelement separators if the **Allow same character for element, subelement and repeating element separators** option on the Trading Partnership Administration Preferences dialog box is checked.
Segment terminator	Defines the unique character used to indicate the end of a segment.
Header Information	
Interchange control header	Defines the type of header segment used for the Interchange envelope. **Reference** See the How to Modify the Outbound Envelope Segments topic in this chapter for instructions on editing the elements in the Interchange envelope segment.
Group control header	Defines the type of group segment used for the Functional Group envelope. **Reference** See the How to Modify the Outbound Envelope Segments topic in this chapter for instructions on editing the elements in the group envelope segment.

(Contd) Field	Function
EDIFACT only	
Decimal mark	Defines the ASCII/hexadecimal character that serves as the decimal point for EDIFACT data.
Release character	Defines the character used to indicate that what appears to be a separator or terminator is actually part of the EDIFACT data. The release character precedes the defined separator or terminator. **Example** Your element separator is a plus sign (+). The name of your trading partner is Toys+Joys. In order for the name to be interpreted correctly, you need a release character (?) in front of the element separator (Toys?+Joys). The release character prevents the translator from incorrectly interpreting the number of elements in this segment by allowing the plus sign to pass through as data and not to be interpreted as an element separator.
Syntax level	Specifies the EDIFACT syntax level. **Note** The syntax level determines the default value that the system displays in the separator/terminator box. (Continued on next page)

(Contd) Field	Function
Generate UNA	Tells Gentran:Server to create an EDIFACT UNA segment as part of the outbound EDIFACT message. Enables the separator, terminator, and release character boxes so that you can change the values. **Example** The following is an example of a UNA segment. UNA : + , ? _ ' where the : represents the component separator + represents the element separator , represents the decimal mark _ reserved for repeating elements ? represents the release character ' represents the segment terminator
Generate acknowledgment based on UNB09	Checks to see if your trading partner requested an acknowledgment. **Note** If the UNB09 element of the inbound document equals 1, then Gentran:Server sends an acknowledgment. If the UNB09 element equals 0, then Gentran:Server does not send an acknowledgment.
UNH --- Association assigned code	Defines the value for the UNH 0057 field, which is mandatory for EANCOM.
General information	
Enable Enveloping	Enables Gentran:Server to place multiple outbound documents in one envelope.

Categories tab   This illustration shows an example of the Categories tab of the Trading Partnership Editor dialog box.
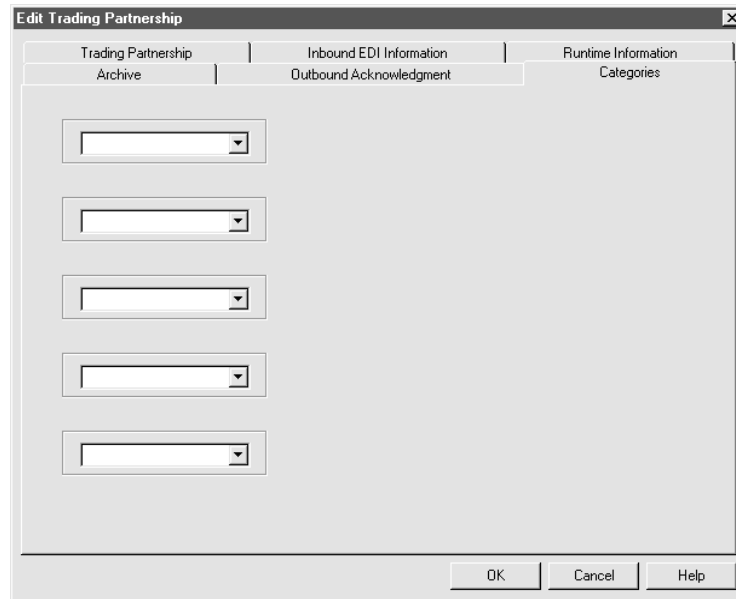
Fields and functions

The fields on the Categories tab are specific to your organization. Each field represents a category type that you can assign to the current Trading Partnership.

You must define the categories on the User Defined Category Types dialog box.

Reference
See the topic How to Create Categories in this chapter for information about defining Trading Partnership category types.

Creating Trading Partnerships

The Flow of Work

Introduction   Gentran:Server supports Trading Partnership records based on the types of translation.

Process   The table describes the process of creating Trading Partnership records.

Stage	Description
1	Create Interchange Organization records. **Reference** See the topic How to Create an Interchange Record in this chapter for instructions on creating Interchange Organization records.
2	Create Group Organization records. **Reference** See the topic How to Create a Group Organization Record in this chapter for instructions on creating Group Organization records. (Continued on next page)

(Contd) Stage	Description	
3	Create Trading Partnership records.	
	IF you want to translate from…	**THEN create…**
	EDI standard format to application file definition	A standard-to-application Trading Partnership record.
	Application file definition to EDI standard format	An application-to-standard Trading Partnership record.
	An EDI standard format to another EDI standard format	A standard-to-standard Trading Partnership record.
	Application file definition to application file definition	An application-to-application Trading Partnership record.
	An EDI standard format to an ODBC file format	A standard-to-ODBC Trading Partnership record.
	Application file definition to ODBC file format	An application-to-ODBC Trading Partnership record.
	ODBC file format to an EDI standard format	An ODBC-to-standard Trading Partnership record.
	ODBC file format to an application file definition	An ODBC-to-application Trading Partnership record.
	ODBC file format to another ODBC file format	An ODBC-to-ODBC Trading Partnership record.

Reference

If you have the XML translation option, see Creating XML Trading Partnership Records in the *Gentran:Server for UNIX and Workstation XML User's Guide* for more information on translating from and creating an XML Trading Partnership record.

Using wildcard characters

When you create a Trading Partnership record, you can enter a wildcard indicator into any of the six key fields that the system uses to identify the record. A wildcard indicator instructs Gentran:Server to accept any value for that field during a search for a Trading Partnership record. Wildcard indicators enable you to create more generic Trading Partnership records.

The Trading Partnership Editor accepts a dollar sign ($) as a wildcard indicator for these fields:

◗ Your Interchange ID

◗ Your Partner's Interchange ID

◗ Your Group/Application ID

◗ Your Partner's Group/Application ID

◗ Set ID

◗ Standard Version

Note
You can only use wildcards for the following trading partnership files:

◗ standard-to-application

◗ standard-to-standard

◗ application-to-standard

Sending outbound functional acknowledgments

When you create a Trading Partnership record, you can specify whether or not to send an outbound functional acknowledgment to your trading partner. The outbound functional acknowledgment notifies your trading partner that you received the data they sent to you.

Reference
See the The Trading Partnership Editor topic for more information on the Outbound Acknowledgment tab fields and their functions.

See How to Specify Outbound Acknowledgments to your Trading Partners topic in this guide for instructions.

How to Create an Interchange Record

Introduction   The first stage in creating a new Trading Partnership is to create an Interchange Organization record. This record contains the information that uniquely identifies your relationship to this trading partner.

New Interchange Organization dialog box example   This illustration shows an example of the New Interchange Organization dialog box.

Fields and functions   This table describes the fields of the New Interchange Organization dialog box and their functions.

Field	Function
Organization Information	
Code	Defines a unique organization code.
Description	Describes the Interchange Organization.
	(Continued on next page)

(Contd) Field	Function
Interchange IDs	
Your Interchange ID	Defines your unique interchange identifier that is inserted into the Interchange envelope when the data is translated.
Partner's Interchange ID	Defines your trading partner's unique interchange identifier that is inserted into the Interchange envelope when the data is translated.
Control Numbers	
Last Sent Control Numbers	Initializes the counter used for control numbers for your business documents. (Outbound only)
Last Received Control Numbers	Initializes the counter used for control numbers for business documents that you receive from this trading partner. (Outbound only)

Note
If you maintain your control numbers globally, the control numbers you specified at the Interchange Organization level will track the documents you exchange with your trading partner.

Procedure   Use this procedure to create an Interchange Organization record.

Step	Action
1	Open the Trading Partnership Administration Explorer.
2	Click **New** from the **File** menu.
3	Click **Interchange Organization**. **System Response** The system displays the New Interchange Organization dialog box. **Reference** See the New Interchange Organization dialog box example in this topic for more information. (Continued on next page)

(Contd) Step	Action
4	Fill in the fields as necessary and click **OK**. **System Response** The system creates and displays a New Interchange Organization record in the Trading Partnership Administration Explorer.
5	GO TO How to Create a Group Organization Record next. **Note** You must create at least one Group Organization record for each Interchange Organization record.

How to Create a Group Organization Record

Introduction

The second stage in creating a new Trading Partnership is to create a Group Organization record. Gentran:Server uses the information that you include in this record to build group-level envelopes during data translation.

You can create multiple group organizations and associate them with a single interchange organization.

You can group many Trading Partnership records under a single Group Organization record. You do not need to create a separate Group Organization record for each Trading Partnership record.

New Group Organization dialog box example

This illustration shows an example of the New Group Organization dialog box.

Fields and functions

This table describes the fields of the New Group Organization dialog box and their functions.

Field	Function
Organization Code	Defines a unique Group Organization code that is inserted into the Group envelope when the data is translated.
Organization Description	Describes the Group Organization.
Your Group ID	Defines your unique group identifier that is inserted into the group envelope when the data is translated.
Partner's Group ID	Defines your trading partner's unique Group identifier that is inserted into the group envelope when the data is translated.
Last Sent Control Numbers	Initializes the counter used for control numbers for your business documents.
Last Received Control Numbers	Initializes the counter used for control numbers for business documents that you receive from this trading partner.

Note

If control numbers are Maintained globally, documents you exchange with a trading partner are tracked by the group control numbers specified at the Group Organization level.

Procedure

Use this procedure to create a Group Organization record.

Step	Action
1	Select the Interchange Organization that you want to associate with this Group Organization record.
2	Click **New** from the **File** menu.

(Contd) Step	Action
3	Click **Group Organization**. **System Response** The system displays the New Group Organization dialog box. **Reference** See the New Group Organization dialog box example in this topic for more information.
4	Fill in the boxes as necessary and click **OK**. **System Response** The system creates and displays a new Group Organization record in the Trading Partnership Administration Explorer.
5	GO TO How to Create a Trading Partnership Record next.

How to Create a Trading Partnership Record

Introduction

The final stage in creating a new Trading Partnership is to create a Trading Partnership record. The information you need to provide for a Trading Partnership is based upon the type of translation you intend to perform.

This topic is divided into these subtopics:

◗ Creating a standard-to-standard Trading Partnership record

◗ Creating a standard-to-application Trading Partnership record

◗ Creating an application-to-standard Trading Partnership record

◗ Creating an application-to-application Trading Partnership record

Note
You use the Trading Partnership Editor to create new Trading Partnership records. When you create new Trading Partnership records, the Trading Partnership Editor collects only the information that pertains to the new Trading Partnership you are creating.

Optional Trading Partnership records

If your company has the Gentran:Server XML translation option, you can also create these types of Trading Partnership records.

◗ Creating an application-to-XML Trading Partnership record

◗ Creating an XML-to-application Trading Partnership record

◗ Creating a XML-to-standard Trading Partnership record

◗ Creating a standard-to-XML Trading Partnership record

◗ Creating an XML-to-XML Trading Partnership record

Reference
See the *Gentran:Server for UNIX and Workstation XML User's Guide* for information about creating these Trading Partnership records.

Creating a standard-to-standard Trading Partnership record

Use this procedure to create a Trading Partnership record that is based on a standard-to-standard translation scheme.

Step	Action
1	Open **Trading Partnership Administration**.
2	Select the Group Organization record that you want to associate with this Trading Partnership.
3	Click **New** from the **File** menu.
4	Click **Trading Partnership**. **System Response** The system displays the Trading Partnership Wizard.
5	Complete the fields on the Trading Partnership dialog box and click **Next**. **Reference** See the Trading Partnership tab topic in this chapter for field descriptions. Note requirements for the Trading Partnership Code.
6	Complete the fields on the Inbound EDI Information dialog box and click **Next**. **References** See the Inbound EDI Information tab topic in this chapter for field descriptions. For information about how to add an entry to the Standard Cross Reference Table, see the How to Add a Record to the Standard Cross Reference Table topic.
7	Complete the fields on the Outbound EDI dialog box and click **Next**. **References** See the Outbound EDI tab in this chapter for field descriptions. See the How to Modify the Outbound Envelope Segments topic in this chapter for instructions on editing the interchange or group envelope segments.
8	Complete the fields on the Runtime Information dialog box and click **Next**. **Reference** See the topic Runtime Information tab in this chapter for field descriptions. *(Continued on next page)*

(Contd) Step	Action
9	Complete the fields on the Archive dialog box and click **Next**. **Reference** See the topic <u>Archive tab</u> in this chapter for field descriptions.
10	Complete the fields on the Inbound Acknowledgment dialog box and click **Next**. **Reference** See the topic <u>Inbound Acknowledgent tab</u> in this chapter for field descriptions.
11	Complete the fields on the Outbound Acknowledgment dialog box and click **Next**. **References** See the topic <u>Outbound Acknowledgment tab</u> in this chapter for field descriptions. See the topic <u>How to Specify Outbound Acknowledgments</u> in this chapter to generate outbound acknowledgments.
12	Select appropriate categories for this Trading Partnership and click the **Finish** button. **Reference** See the topic <u>Categories tab</u> in this chapter for field descriptions. **Note** The Categories dialog box is displayed only if categories are currently defined.

Creating a standard-to-application Trading Partnership record

Use this procedure to create a Trading Partnership record that is based on a standard-to-application translation scheme.

Step	Action
1	Open **Trading Partnership Administration**.
2	Select the Group Organization record that you want to associate with this Trading Partnership.
3	Click **New** from the **File** menu.

(Contd) Step	Action
4	Click **Trading Partnership**. **System Response** The system displays the Trading Partnership wizard.
5	Complete the fields on the Trading Partnership dialog box and click **Next**. **Reference** See the topic Trading Partnership tab in this chapter for field descriptions. Note requirements for the Trading Partnership Code.
6	Complete the fields on the Inbound EDI Information dialog box and click **Next**. **Reference** See the topic Inbound EDI Information tab in this chapter for field descriptions. For information about how to add an entry to the Standard Cross Reference Table, see the How to Add a Record to the Standard Cross Reference Table topic.
7	Complete the fields on the Runtime Information dialog box and click **Next**. **Reference** See the topic Runtime Information tab in this chapter for field descriptions.
8	Complete the fields on the Archive dialog box and click **Next**. **Reference** See the topic Archive tab in this chapter for field descriptions.
9	Complete the fields on the Outbound Acknowledgment dialog box and click **Next**. **References** See the topic Outbound Acknowledgment tab in this chapter for field descriptions. See the topic How to Specify Outbound Acknowledgments in this chapter to generate outbound acknowledgments. **System Response** If categories are currently defined, the system displays the Categories dialog box. *(Continued on next page)*

(Contd) Step	Action
10	If the Categories dialog box is displayed, select appropriate categories for this Trading Partnership. **Reference** See the topic Categories tab topic in this chapter for field descriptions.
11	Click the **Finish** button.

Creating an application-to-standard Trading Partnership record

Use this procedure to create a Trading Partnership record that is based on an application-to-standard translation scheme.

Step	Action
1	Open **Trading Partnership Administration**.
2	Select the Group Organization record that you want to associate with this Trading Partnership.
3	Click **New** from the **File** menu.
4	Click **Trading Partnership**. **System Response** The system displays the Trading Partnership wizard.
5	Complete the fields on the Trading Partnership dialog box and click **Next**. **Note** You must complete the **File Definitions filename** box. **Reference** See the Trading Partnership tab topic in this chapter for field descriptions. Note requirements for the Trading Partnership Code. (Continued on next page)

(Contd) Step	Action
6	Complete the fields on the Outbound EDI dialog box and click **Next**. **References** See the <u>Outbound EDI tab</u> topic in this chapter for field descriptions. See the <u>How to Modify the Outbound Envelope Segments</u> topic in this chapter for instructions on editing the interchange or group envelope segments.
7	Complete the fields on the Runtime Information dialog box and click **Next**. **Reference** See the topic <u>Runtime Information tab</u> in this chapter for field descriptions.
8	Complete the fields on the Archive dialog box and click **Next**. **Reference** See the topic <u>Archive tab</u>in this chapter for field descriptions.
9	Select appropriate categories for this Trading Partnership and click the **Finish** button. **Reference** See the topic <u>Categories tab</u> in this chapter for field descriptions. **Note** The Categories dialog box is displayed only if categories are currently defined.

Creating an application-to-application Trading Partnership record

Use this procedure to create a Trading Partnership record that is based on an application-to-application translation scheme.

Step	Action
1	Open **Trading Partnership Administration**.
2	Select the group organization record that you want to associate with this Trading Partnership.
3	Click **New** from the **File** menu.

(Contd) Step	Action
4	Click **Trading Partnership**. **System Response** The system displays the Trading Partnership wizard.
5	Complete the fields on the Trading Partnership dialog box and click **Next**. **Note** You must complete the **File Definitions filename** box. **Reference** See the topic <u>Trading Partnership tab</u> in this chapter for field descriptions. Note requirements for the Trading Partnership Code.
6	Complete the fields on the Runtime Information dialog box and click **Next**. **Reference** See the topic <u>Runtime Information tab</u> in this chapter for field descriptions.
7	Select appropriate categories for this Trading Partnership and click the **Finish** button. **Reference** See the topic <u>Categories tab</u> in this chapter for field descriptions. **Note** The Categories dialog box is displayed only if categories are currently defined.

How to Specify Outbound Acknowledgments

Introduction

This topic explains how to use the Outbound Acknowledgment tab to generate outbound acknowledgments for documents you receive from a trading partner.

When to use

Complete the Outbound Acknowledgment tab when a trading partner requests an acknowledgment in response to a document they sent you.

Generating outbound acknowledg-ments for X12 documents

Use this procedure to generate outbound acknowledgements for this trading partner.

Step	Action
1	Select the check box **Create acknowledgement for inbound document**.
2	From the **Type** drop-down list, select the type of acknowledgment you are sending to your trading partner.
3	From the **Level** drop-down list, select the appropriate acknowledgment level.
4	Type the path name you want to use for the output file.
5	From the **Element separator** drop-down list, select the element separator.
6	From the **Component/Sub-elem.** drop-down list, select the component or sub-element separator.
7	From the **Segment terminator** drop-down list, select the segment terminator.
8	From the **Interchange control header** drop-down list, select the interchange control header.
9	Accept the default value for the group control header.
10	Do you want to envelope multiple outbound acknowledgments? ◗ If YES, select the **Enable Enveloping** check box and continue with Step 11. ◗ If NO, continue with Step 11.
11	Click **OK** to save your changes.

Generating outbound acknowledg- ments for EDIFACT message types

Use this procedure to generate outbound acknowledgements for this trading partner.

Step	Action
1	Select the check box **Create acknowledgement for inbound document**.
2	Accept the **CONTRL** default in the Type box.
3	From the **Level** drop-down list, select the appropriate acknowledgment level.
4	Type the path name you want to use for the output file.
5	From the **Decimal mark** drop-down list, select the decimal mark.
6	From the **Syntax level** drop-down list, select the syntax level. **Note** If a syntax level above UNOB is needed, specify syntax level B, then indicate the actual syntax level by clicking **Edit** by the Interchange control header and typing the desired value in the syntax identifier field (UNOC, UNOD…UNOG).
7	Do you want to generate a UNA segment? ◗ If YES, select the **Generate UNA** check box and continue with Step 8. **System Response** The system displays and enables the Element separator, Component/sub-elem. separator, Segment terminator, and release character box. ◗ If NO, continue with Step 13.
8	Do you want to change the Separator/Terminator information? ◗ If YES, continue with the next step. ◗ If NO, GO TO Step 12. (Continued on next page)

(Contd) Step	Action
9	Do you want to change the value of the element separator? ◗ If YES, from the **Element separator** drop-down list, select the Element separator; continue with the next step. ◗ If NO, continue with the next step. **CAUTION** **Checking the Generate UNA box enables the element separator box, so that you can change the default value. Before you change the default value, you must verify that your trading partner can process the new value.**
10	Do you want to change the value of the component or sub-element? ◗ If YES, from the **Component/Sub-elem.** drop-down list, select Component/Sub-elem. separator; continue with the next step. ◗ If NO, continue with the next step. **CAUTION** **Checking the Generate UNA box enables the Component/Sub-elem. box, so that you can change the default value. Before you change the default value, you must verify that your trading partner can process the new value.**
11	Do you want to change the value of the segment terminator? ◗ If YES, from the **Segment terminator** drop-down list, select the Segment terminator; continue with the next step. ◗ If NO, continue with the next step. **CAUTION** **Checking the Generate UNA box enables the Segment terminator box, so that you can change the default value. Before you change the default value, you must verify that your trading partner can process the new value.** (Continued on next page)

(Contd) Step	Action
12	Do you want to change the value of the release character? ◗ If YES, from the **Release character** drop-down list, select the release character; then continue with the next step. ◗ If NO, continue with the next step. **CAUTION** **Checking the Generate UNA box enables the Release character box, so that you can change the default value. Before you change the default value, you must verify that your trading partner can process the new value.**
13	Do you want to generate an acknowledgment based upon the incoming UNB09 value? ◗ If YES, select the **Generate acknowledgment upon UNB09 request** check box and continue with the next step. ◗ If NO, continue with the next step.

14	Do you want to define the value for the UNH 0057 field?
	
	◗ If YES, type the value in the UNH --- Associated assigned code text box; continue with the next step.
	
	    Note
	    The UNH 0057 field is mandatory for EANCOM.
	
	◗ If NO, continue to the next step.

15	From the **Interchange control header** drop-down list, select the Interchange control header.
16	Do you want to accept the default value for the group control header. ◗ If YES, continue with the next step. ◗ If NO, from the Group control header drop-down list, select the group control header value; continue with the next step. **CAUTION** **Before you change the default value, you must verify that your trading partner can process the new value.** (Continued on next page)

(Contd) Step	Action
17	Do you want to envelope multiple outbound acknowledgments? ▸ If YES, select the **Enable Enveloping** check box and continue with the next step. ▸ If NO, continue with the next step.
18	Click **OK** to save your changes.

Using Standard Cross Reference Tables

Overview

Introduction   Some Gentran:Server components, such as the translator, look in the Trading Partnership record to determine which compiled map (translation object) to use for translation. To find the correct Trading Partnership record, Gentran:Server examines each inbound file to determine the six key pieces of information needed to find the Trading Partnership record. These are the six key fields:

◗   Your Interchange ID

◗   Trading Partner's Interchange ID

◗   Your Group ID

◗   Trading Partner's Group ID

◗   Document (Set) ID

◗   Standard Version

Standard version as key field   If the files that a trading partner sends you contain values in the standard version field that Gentran:Server does not recognize, the system cannot find the Trading Partnership record.

Note
Gentran:Server may find the Trading Partnership record if you used a dollar sign ($) as a wildcard indicator for the Standard version field on the Inbound EDI tab of the Trading Partnership record.

Purpose of the Standard Cross Reference Table   The Standard Cross Reference Table is a Gentran:Server feature that enables you to specify the standard version that you want Gentran:Server to use when it finds certain values in the inbound document's standard version fields. Gentran:Server then uses the standard version reference in the table and the other key fields to find the Trading Partnership record.

Example

Your trading partner uses an older version of the EDIFACT standards. The inbound documents this trading partner sends you have 0 in the message release number field and 0 in the release number field. (These are the fields Gentran:Server uses to determine the standard version.) When the translator checks these fields for the standard version, it will try to find a Trading Partnership record for standard version 00000 for the trading partner's Interchange and Group IDs.

You create an entry in the Standard Cross Reference Table. The entry cross references the message release number "0" and release number "0" with standard version 001911. This entry directs the translator to use standard version 001911 in a subsequent search if it cannot find a Trading Partnership record for standard version 00000 in its first search.

Translation types

You can use the Standard Cross Reference Table to locate Trading Partnership records that are based on these translation types:

◗ Standard-to-application

◗ Standard-to-standard

◗ Standard-to-XML (if you have the XML translation option)

Standard Cross Reference Table files

The Gentran:Server installation process installs an empty Standard Cross Reference Table automatically. The files are DISAM files named *xtable.dat* and *xtable.idx*. Gentran:Server stores the files in the directory that holds your Trading Partner files.

Gentran:Server provides load and unload utilities to move the *xtable.dat* and *xtable.idx* files.

Reference

See the Moving Files chapter of the *Gentran:Server for UNIX Maintenance and Troubleshooting Guide* or the Moving Files chapter of the *Gentran:Server Workstation Maintenance and Troubleshooting Guide* for information about the load and unload utilities.

Standard Cross Reference Table entry

You access the table from the Inbound EDI tab of the Trading Partnership record. Each entry you make to the table is unique to the Interchange ID and Group ID of your trading partner. When Gentran:Server checks the table, first it looks for the Interchange ID and Group ID and then for the standard version cross-reference under these IDs.

Standard Cross Reference Table report

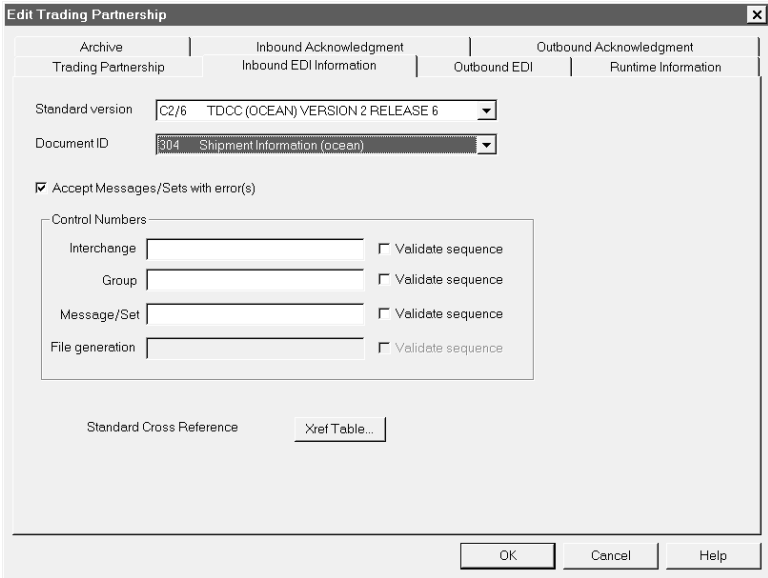Gentran:Server provides the **xtablerpt** command to create a report that shows all the entries in the Standard Cross Reference Table.
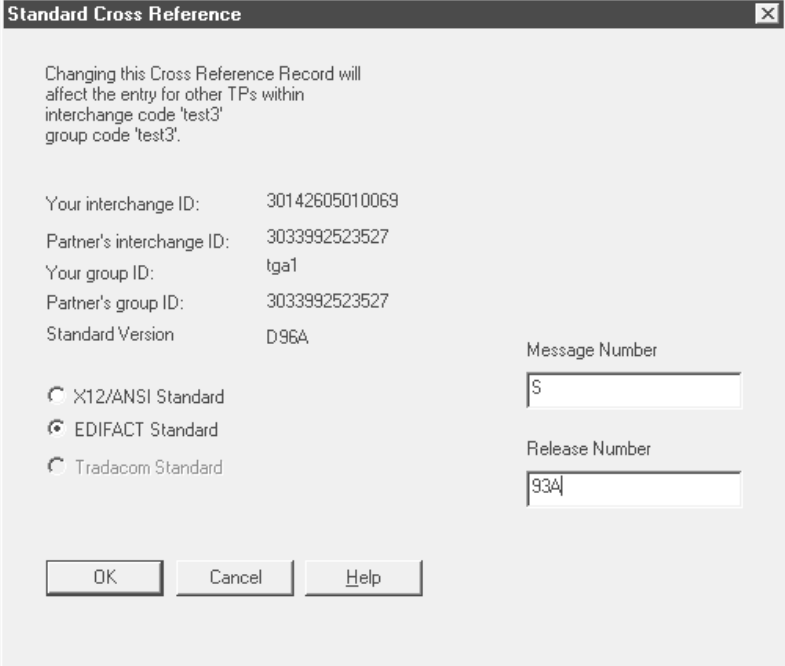
Reference

See the Command Reference chapter in the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for instructions.

How to Add a Record to the Standard Cross Reference Table

Introduction   A record in the Standard Cross Reference Table connects two standard version values:

◗   An invalid standard version. This is the value or values that you expect Gentran:Server to extract from the inbound documents from a particular interchange and group ID.

◗   A valid standard version. This is the standard version that you want Gentran:Server to use as one of the six key fields it uses to find the Trading Partnership record for the interchange and group ID.

Standard Cross Reference dialog box-EDIFACT   This illustration shows the Standard Cross Reference dialog box for an EDIFACT standard.

Standard Cross Reference dialog box-X12/ANSI

This illustration shows the Standard Cross Reference dialog box for an X12/ANSI standard.

Fields and functions

This table lists the fields of the Standard Cross Reference dialog box and their functions.

Field	Function
Your interchange ID	Displays your interchange ID
Partner's interchange ID	Displays your trading partner's interchange ID
Your group ID	Displays your group ID.
Partner's group ID	Displays your trading partner's group ID.
Standard Version	Displays the standard version used as a key field to identify this Trading Partnership record. This is the value you selected from the **Standard version** box on the Inbound EDI Information tab.
X12/ANSI Standard	Selects X12/ANSI as the standard type.
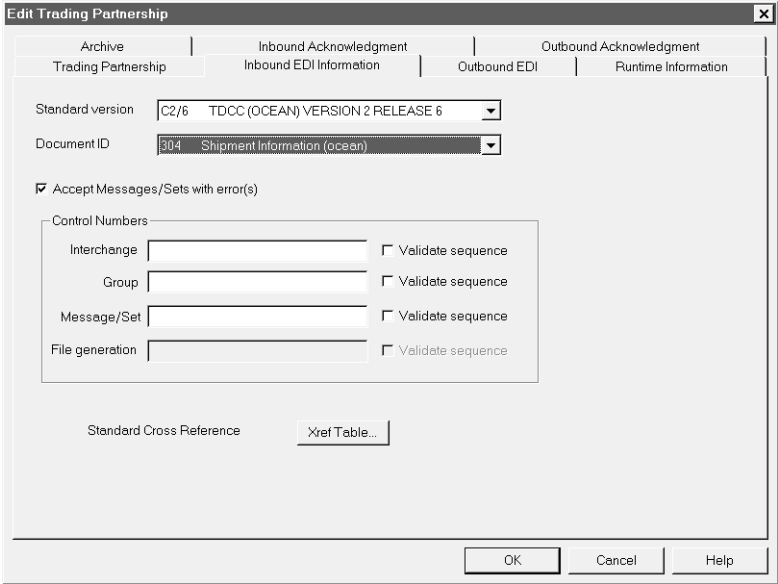
(Contd) Field	Function
EDIFACT Standard	Selects EDIFACT as the standard type.
Tradacom Standard	Reserved for future use. Disabled in the current release.
Message number/ Standard Version	**EDIFACT documents** Specifies the value in the inbound document's Message number field. Gentran:Server uses this value and the Release number value to determine the standard version from an inbound document. This is one of the values you are cross-referencing to the valid standard version so that Gentran:Server can find the Trading Partnership record. **X12/ANSI documents** Specifies the value in the inbound document's Standard Version field. Gentran:Server uses this value to determine the standard version from an inbound document. This is the value you are cross-referencing to the valid standard version so that Gentran:Server can find the Trading Partnership record.
Release number	**EDIFACT documents** Specifies the value in the inbound document's Release number field. Gentran:Server uses this value and the Message number value to determine the standard version from an inbound document. This is one of the values you are cross-referencing to the valid standard version so that Gentran:Server can find the Trading Partnership record. **X12/ANSI documents** This field is disabled for these standards.
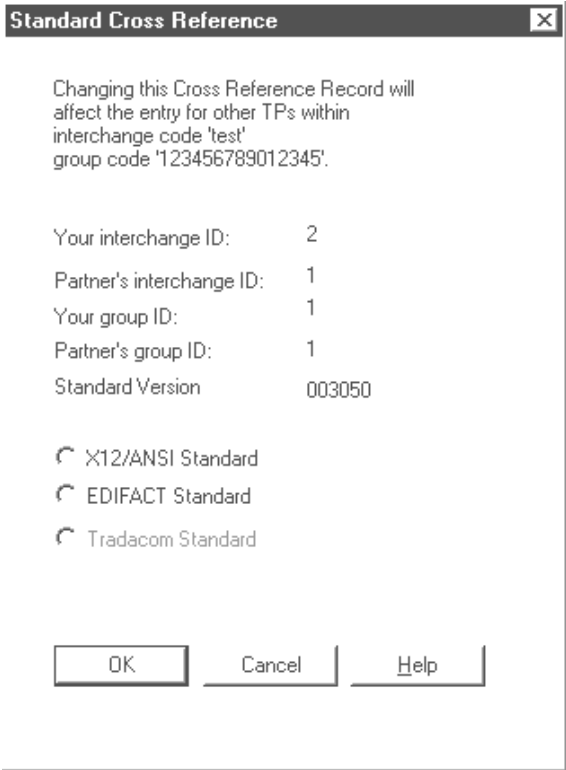
Before you begin   You access the Standard Cross Reference feature from the Inbound EDI tab of the Trading Partnership record. You can add an entry to the table at the time you create the Trading Partnership record or when you edit the record.

Procedure   Use this procedure to add an entry to the Standard Cross Reference Table from an open inbound Trading Partnership record.

Step	Action
1	Click the Inbound EDI Information tab of the Trading Partnership record.

(Continued on next page)

(Contd) Step	Action
2	Click the **Xref Table** button. **System Response** The system displays the Standard Cross Reference dialog box. ![Standard Cross Reference dialog box] **Standard Cross Reference** ✕ Changing this Cross Reference Record will affect the entry for other TPs within interchange code 'test' group code '123456789012345'. Your interchange ID: 2 Partner's interchange ID: 1 Your group ID: 1 Partner's group ID: 1 Standard Version 003050 ○ X12/ANSI Standard ○ EDIFACT Standard ○ Tradacom Standard [OK] [Cancel] [Help]
3	Verify that the interchange ID, group ID, and standard version information are correct.
4	Select the type of standard: ◗ X12/ANSI Standard ◗ EDIFACT Standard **System Response** The dialog box expands and displays additional text boxes. (Continued on next page)

(Contd) Step	Action
5	Use this table to determine your next step.

IF the standard type is…	THEN enter the…
X12/ANSI	Value that Gentran:Server will find in the **Standard version** field of the trading partner's inbound documents
EDIFACT	Values that Gentran:Server will find in the **Message number** and **Release number** fields of the trading partner's inbound documents

(Contd) Step	Action
6	Click **OK** to add the new entry to the Standard Cross Reference Table.

Standard Cross Reference Table report

To create a report that shows all the entries in the Standard Cross Reference Table, use the **xtablerpt** command.

Reference
See the Command Reference chapter in the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for instructions.

How to Edit a Standard Cross Reference Record

Procedure   Use this procedure to edit a Standard Cross Reference record from an open inbound Trading Partnership record.

Step	Action
1	Click the Inbound EDI Information tab of the Trading Partnership record. (Continued on next page)

(Contd) Step	Action
2	Click the **Xref Table** button. **System Response** The system displays the Standard Cross Reference dialog box.
3	Modify the record.
4	Click **OK** to save your changes to the Standard Cross Reference Table.
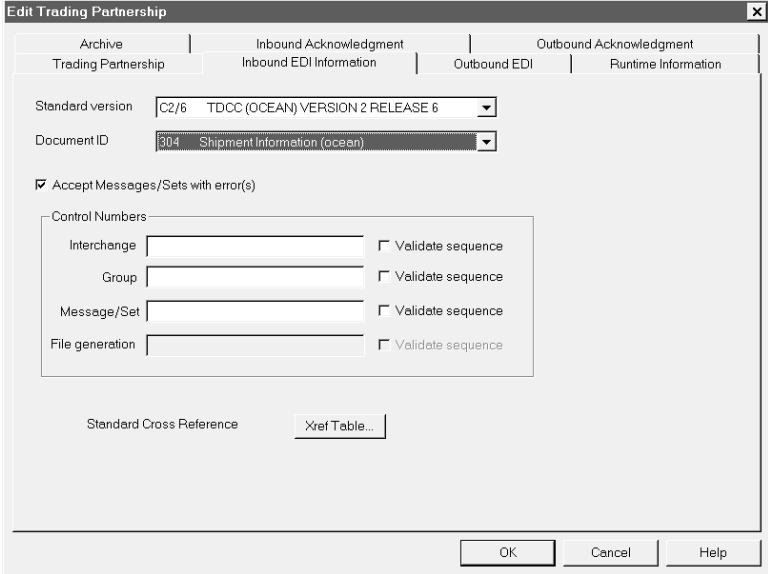
How to Delete a Standard Cross Reference Record
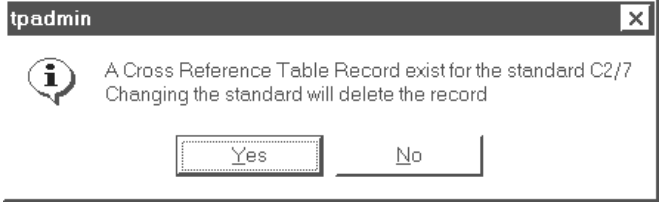
Introduction   This topic explains how to delete a Standard Cross Reference record.

CAUTION

Gentran:Server **maintains Standard Cross Reference records at the Group Organization level. When you delete a Standard Cross Reference record for one Trading Partnership in a specific Group Organization, you delete the Standard Cross Reference record for every Trading Partnership in the Group Organization.**

Procedure   Use this procedure to delete a Standard Cross Reference record from an open inbound Trading Partnership record.

Step	Action
1	Click the Inbound EDI Information tab of the Trading Partnership record. (Continued on next page)

(Contd) Step	Action
2	Select a different standard version from the **Standard version** drop-down box. **System Response** The system displays a prompt to delete the cross reference record.
3	Click **Yes**. **System Response** The system displays the Inbound EDI tab.
4	Click **OK** to save your changes to the Trading Partnership record.

Creating Optional Records

Overview

Introduction   Contact records and Category records are optional Trading Partnership records.

Contact records   Contact records are used to hold names, addresses, and phone numbers of the people you communicate with in your trading partner's organization.

Category records   The category feature enables you to group your Trading Partner records by common category. It is a flexible feature that you can use organize your Trading Partner records in your own way.

Edit Organization Contact Dialog Box

Edit Organization Contact dialog box example

This illustration shows an example of the Edit Organization Contact dialog box.

Fields and functions

This table describes the fields of the Organization Contact dialog box and their functions.

Field	Function
Contact Name	Defines the name of the person to contact.
Address Line 1	Defines the street address of the contact.
Address Line 2	Defines a second street address (room or mail stop) of the contact.
City	Defines the city.
State	Defines the state or province (2 characters).
Zip	Defines the zip code.

(Contd) Field	Function
Country	Defines the country.
Phone 1	Defines a phone number and extension.
Phone 2	Defines a second phone number and extension.
Fax	Defines the fax number of the contact.
Email	Defines the e-mail address of the contact.
Comments	Provides a method to record additional information about this contact.

How to Create Contact Records

Introduction

Use Contact records to maintain information about personnel in your trading partner's organization. You can create unique contact records at interchange organization, group organization, and Trading Partnership levels.

Procedure

Use this procedure to create a new Contact record.

Step	Action
1	Open the Trading Partnership Administration explorer window.
2	Select the Interchange Organization, Group Organization, or the Trading Partnership record to which you want to add contact information.
3	Click **Contact** from the **Edit** menu. **System Response** The system displays the Edit Organization Contact dialog box. **Reference** See the Edit Organization Contact Dialog Box in this topic for more information.
4	Fill in the fields as necessary and click **OK**.

How to Create Categories

Introduction

It is often useful to group Trading Partner records by common category. After you define categories, you can use them as selection criteria for viewing, changing, or deleting Trading Partnership records.

You define the categories according to similarities of interest. You may define up to five category types. After you define categories, the category types appear on the Trading Partnership editor's Categories tab.

Example

You create a category called INDUSTRY so that you can group trading partners by industry. Next, you create values for different types of industries, such as retail, manufacturing, or warehousing. You then use these values to locate relevant records when you need to search for a particular group of Trading Partnerships.

Procedure

Use this procedure to define Trading Partnership category types.

Step	Action
1	Click **Categories…** from the **Settings** menu. **System Response** The system displays the User Defined Category Types dialog box.
2	Enter the following information for each category you are setting up: ▶ Category—The prompt that you want to appear on the Partnership Categories dialog box. ▶ Value—Appropriate values for each category.
3	Click **OK** to save your categories.

Modifying Trading Partnership Records

Overview

In this section   This section explains how to:

◗   Edit Trading Partnership records

◗   Modify outbound envelope segments

◗   Choose a different map for a Trading Partnership record

◗   Delete a Trading Partnership record

◗   Print Trading Partnership records

◗   Change the translation option for a Trading Partnership record.

About file locking in UNIX version   In Gentran:Server for UNIX, the Security subsystem locks Trading Partnership records when you open them for editing. When a file is locked, Gentran:Server:

◗   Prevents you from opening it in another session when you run multiple sessions on your desktop.

◗   Stops another user from opening the file for editing and displays a message to let the user know that the file is not available for editing, but does allow the user to view the file.

Gentran:Server locks these Trading Partnership files:

◗   Trading Partner record

◗   Group Organization record

◗   Interchange Organization record

Gentran:Server does not lock these files; so, you must take care not to edit the same file in a parallel session or on another desktop.

◗   Files edited in the File Browser

◗   Trading Partnership Category files

◗   Trading Partnership Contact files.

Note
If the same file is edited in two different parallel sessions, the last changes saved overwrite earlier changes from another session.

How to Edit Trading Partnership Records

Introduction   This topic explains how to edit a Trading Partnership record.

Procedure   Use this procedure to edit Trading Partnership records.

Step	Action
1	Open Trading Partnership Administration and display the Trading Partnership records.
2	Select the Trading Partnership that you want to modify.
3	Right-click and select **Edit** from the menu. **System Response** If the Trading Partnership record is not being used by another user or process, the system displays the Trading Partnership record.
4	Click the tab that contains the information you want to change.
5	Modify the Trading Partnership record and click **OK**.

Reference
If you need to search for the record you want to edit, see the topic How to Search for a Trading Partnership Record in this chapter.

How to Modify the Outbound Envelope Segments

Introduction
You can modify the values of the elements in the interchange and group envelope segments that Gentran:Server inserts in the envelope segments of a Trading Partnership's:

◗ Outbound EDI documents

◗ Outbound Functional Acknowledgments.

Standards compliance
The elements in the interchange and group envelope segments depend on the standard, the standard version, or (for EDIFACT) the syntax level. Gentran:Server bases the options it displays for the elements on the information you specify.

Example 1
If you specify X12/ANSI standard version 4020 and later, Gentran:Server displays "Repeating Element Separator" and a drop-down list of valid values for element ISA11 on the Interchange Information dialog box. For standard versions prior to 4020, Gentran:Server displays "Interchange Standards ID" in the ISA11 element field.

Example 2
If you specify UN/EDIFACT Syntax Version 4 for UNB element 0002 on the Interchange Header Information dialog box, Gentran:Server shows an 8-digit format (for example, 19991129) for element 0017, Date of Preparation, on the UNB and UNG dialog boxes.

Modifying the envelope segments for outbound EDI documents

Use this procedure to modify the envelope segments of outbound EDI documents for a Trading Partnership.

Step	Action
1	Open the Trading Partnership record that you want to modify. **Reference** See How to Search for a Trading Partnership Record or How to Edit Trading Partnership Records for instructions on opening Trading Partnership records.
2	Click the Outbound EDI tab. (Continued on next page)

(Contd) Step	Action
3	Click the **Edit** button of the **Interchange control header** or **Group control header** field. **System Response** The system displays the Information dialog box for the segment you selected from the drop-down box. The standard version and EDIFACT syntax level determine which elements are displayed on the dialog box. This illustration is for the ISA envelope segment for X12/ANSI standard version 4020.
4	Modify the element or elements you want to change. **Reference** See your Standards documentation for the element definitions of the interchange and group control header segments. (Continued on next page)

(Contd) Step	Action
5	Click **OK**. **Warning** You MUST click **OK** on this dialog box to save your changes. **System Response** The system displays the Outbound EDI tab.
6	Click **OK** to save your changes. **Warning** You MUST click **OK** on the Outbound EDI tab to save your changes to the envelope segment.

Modifying the envelope segments for outbound Functional Acknowledg-ments

Use this procedure to modify the envelope segments of outbound functional acknowledgments for a Trading Partnership record.

Step	Action
1	Open the Trading Partnership record that you want to modify. **Reference** See <u>How to Search for a Trading Partnership Record</u> or <u>How to Edit Trading Partnership Records</u> for instructions on opening Trading Partnership records. (Continued on next page)

(Contd) Step	Action
2	Click the Outbound Acknowledgment tab.
3	Click the **Edit** button of the **Interchange control header** or **Group control header** field. **System Response** The system displays the Information dialog box for the segment
4	Modify the element or elements you want to change. **Reference** See your Standards documentation for the element definitions of the interchange and group control header segments.
5	Click **OK**. **System Response** The system displays the Outbound Acknowledgment tab.
6	Click **OK** to save your changes. **Warning** You MUST click **OK** on the Outbound Acknowledgment tab to save your changes to the envelope segment.

How to Choose a Different Map (Translation Object)

Introduction

When working with your Trading Partnership records, you may want to change the compiled map (translation object) you use to translate data. To locate a compiled map, you enter the search criteria into the Compiled Map Search dialog box.

Compiled Map Search dialog box

This illustration shows the Compiled Map Search dialog box. You access this dialog box via the Trading Partnership tab of the Trading Partner Editor.
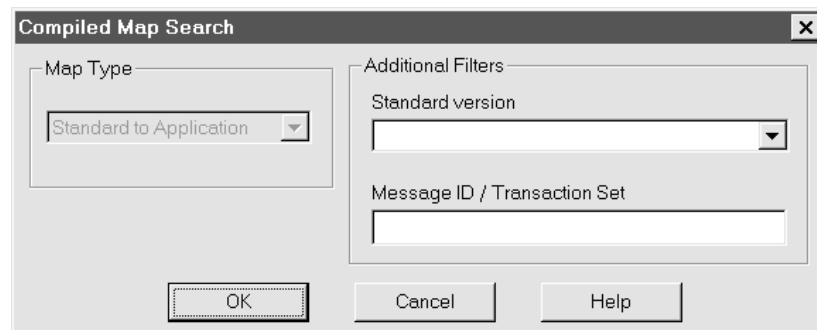
Note

You can search only for maps that match the type you selected on the Trading Partnership tab of the Trading Partnership Editor.

Fields and functions

This table describes the fields of the Compiled Map Search dialog box and their functions.

Field	Function
Map type	Selects the type of compiled map (translation object) to search for.
	Notes
	If you selected a translation type on the Trading Partnership tab, then this field is already selected when you open the Map Search dialog box.
	If you did not specify a Translation type on the Trading Partnership dialog box, the Compiled Map Search dialog box displays "All" in this field.
	(Continued on next page)

(Contd) Field	Function
Standard version	Selects a specific EDI standard version.
Message ID / Transaction Set	Select a specific EDI document or transaction set.

Procedure   Use this procedure to search for the compiled map (translation object) you want to use for a specific Trading Partnership.

Step	Action
1	Open the Trading Partnership Administration explorer window.
2	Select the Trading Partnership record you want to modify.
3	Select the Trading Partnership tab.
4	Click the binoculars icon next to the Map name box. **System Response** The system displays the Compiled Map Search dialog box.
5	Complete the search fields and click **OK**. **System Response** The system displays Map Search Results dialog box, which displays a list of the compiled maps that match the criteria you entered.
6	Select the compiled map (translation object) you want to use with the Trading Partnership and then click **OK**. **System Response** The system replaces the map name specified on the Trading Partnership tab with the map you selected.

How to Delete Trading Partner Records

Introduction

This topic describes the procedure for deleting Interchange Organization, Group Organization, and Trading Partnership records.

Caution
When you delete an Interchange Organization record, you also delete all Group Organization and Trading Partnership records in that Interchange Organization.

When you delete a Group Organization record, you also delete all Trading Partnership records in that Group Organization.

Procedure

Use this procedure to completely remove Trading Partner records from Gentran:Server.

Step	Action
1	Open the Trading Partner Administration explorer.
2	Use this table to determine your next step. **IF you want to delete…** / **THEN select the…** / **AND click…** (see table below) **System Response** The system asks you to confirm your intention to delete this item.
3	Click **OK** to confirm the deletion.

IF you want to delete…	THEN select the…	AND click…
an Interchange Organization record	interchange	**Delete** from the **Edit** menu.
a Group Organization record	group	**Delete** from the **Edit** menu.
a Trading Partnership record	Trading Partnership	**Delete** from the **Edit** menu.

How to Print Trading Partnership Reports

Introduction   This topic describes how to print Trading Partnership records.

When to use   Use this procedure when you want to:

◗ Review the information that you defined in a Trading Partnership

◗ Check for errors

◗ Provide Trading Partnership information to other individuals or organizations.

Procedure   Use this procedure to print a Trading Partnership record.

Step	Action
1	Open the Trading Partner Administration Explorer.
2	Select the Interchange Organization, Group Organization, or Trading Partner record that you want to print.
3	Click **Print** from the **File** Menu. **System Response** Gentran:Server displays the Trading Partnership Print dialog box.
4	Click **OK** to print the selected record.

Translation Options Dialog Boxes

Introduction

This topic describes the Translation Options dialog boxes for the various types of translation (standard-to-application, standard-to-standard, etc.).

You can use these dialog boxes to set particular default translation options for your Trading Partnership records, or to set options for an individual Trading Partnership record.

Reference

To set default translation options for one or more translation types, see How to Set Default Translation Selections.

To set translation options for an individual Trading Partnership, see How to Set Translation Selections.

Standard to Application Translation Options dialog box

This illustration shows the Standard to Application Translation Options dialog box.
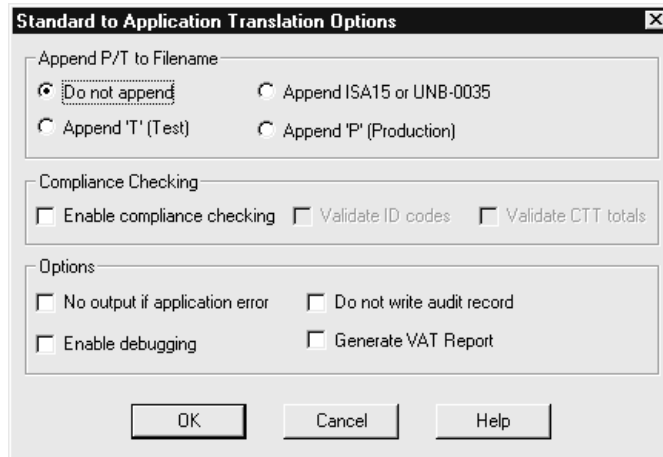
Standard to Standard Translation Options dialog box

This illustration shows the Standard to Standard Translation Options dialog box.

Standard to XML Translation Options dialog box

This illustration shows the Standard to XML Translation Options dialog box.

Application to Standard Translation Options dialog box

This illustration shows the Application to Standard Translation Options dialog box.
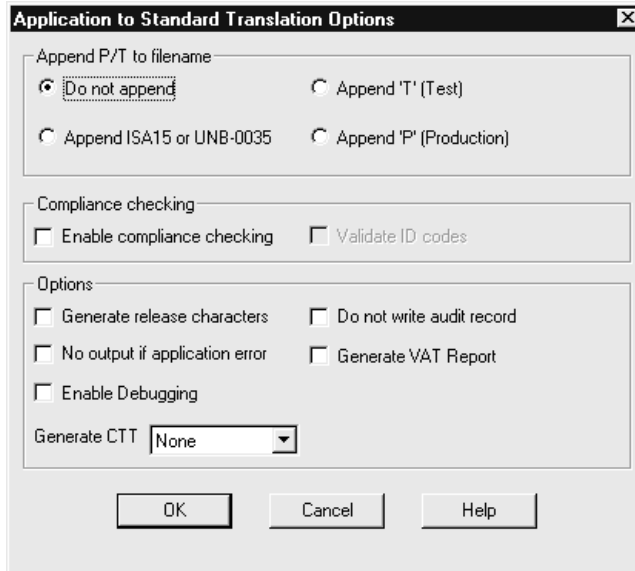
Application to Application Translation Options dialog box

This illustration shows the Application to Application Translation Options dialog box.
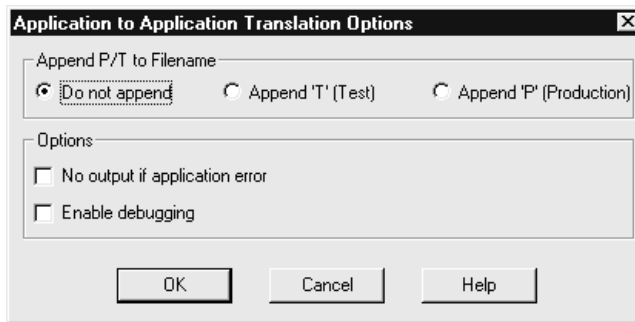
Application to XML Translation Options dialog box

This illustration shows the Application to XML Translation Options dialog box.
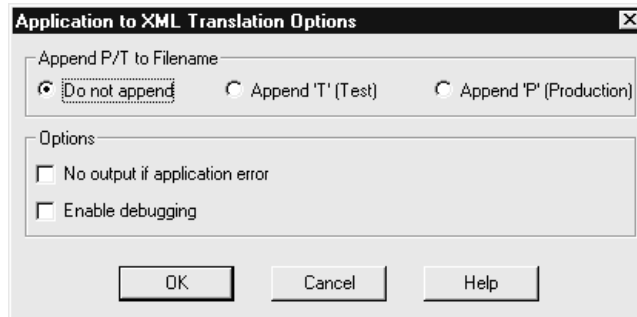
XML to Application Translation Options dialog box

This illustration shows the XML to Application Translation Options dialog box.
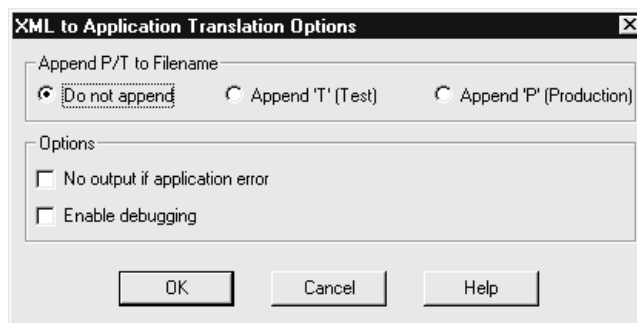
XML to Standard Translation Options dialog box

This illustration shows the XML to Standard Translation Options dialog box.
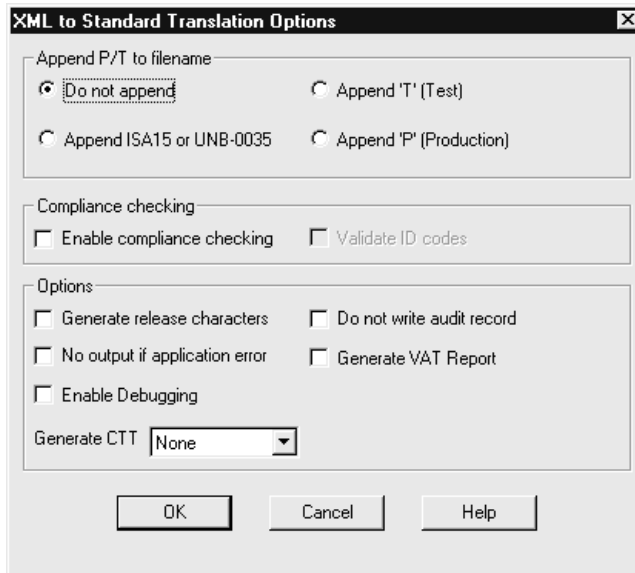
XML to XML Translation Options dialog box

This illustration shows the XML to XML Translation Options dialog box.
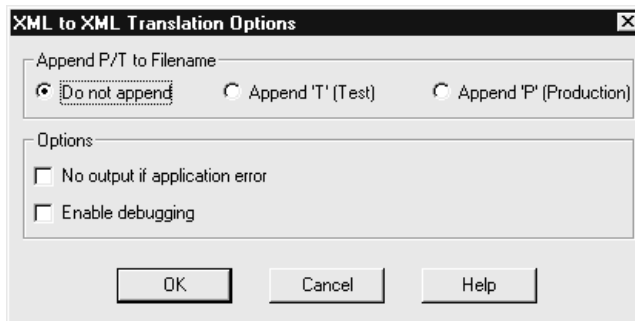
Fields and functions

This table describes all the possible fields of the Translation Options dialog box and their functions. The available values to choose from within the Options and Compliance checking sections of the Translation Options dialog box depend upon the type of translation selected and the type of map file used.

Field	Function
Input Compliance checking	Checks the incoming data for compliance to EDI standard versions. **Note** This translation option is available only when you are translating data with a Visual Mapper (*.vmp) map file type.
Output compliance checking	Checks the outgoing data for compliance to EDI standard versions. **Note** This translation option is available only when you are translating data with a Visual Mapper (*.vmp) map file type.
Do not append	Does not append a test or production flag to the end of the file name.
Append 'T' (Test)	Appends the letter T to the end of the file name to mark it as a test file.
Append 'P' (Production)	Appends the letter P to the end of the file name to mark it as a production file.
Append ISA15 or UNB-0035	Appends the value of ISA15 or UNB0035 to the end of the file name.
Generate release characters	Generates the character that allows a separator or terminator to be used as part of the EDIFACT data. **Note** This translation option is available only when you are translating data with a Visual Mapper (*.vmp) map file type. *(Continued on next page)*

(Contd) Field	Function
Enable Compliance Checking	Checks EDI data for compliance against the EDI Standard .CND file and the map. **Note** This translation option is available only when you are translating data with a Visual Mapper (*.vmp) map file type.
Validate ID codes	Validates the ID codes from the EDI data against the .CDS standard file. **Note** This translation option is available only when you are translating data with a Visual Mapper (*.vmp) map file type.
Validate CTT totals	Checks the incoming data for valid CTT totals. **Note** This translation option is available only when you are translating data with a Visual Mapper (*.vmp) map file type.
No output if application error	Does not generate an output file if there are errors in the application. **Note** This translation option is available only when you are translating data with a Visual Mapper (*.vmp) map file type.
Enable debugging	Lists the segment IDs to screen during translation and generates the Diagnostics Report.
Do not write audit record	Stops the translator from writing an audit record.
Generate VAT Report	Saves EDIFACT VAT information from TAXCON messages.
Generate CTT	Generates the CTT outbound segment. **Note** This translation option is available only when you are translating data with a Visual Mapper (*.vmp) map file type.
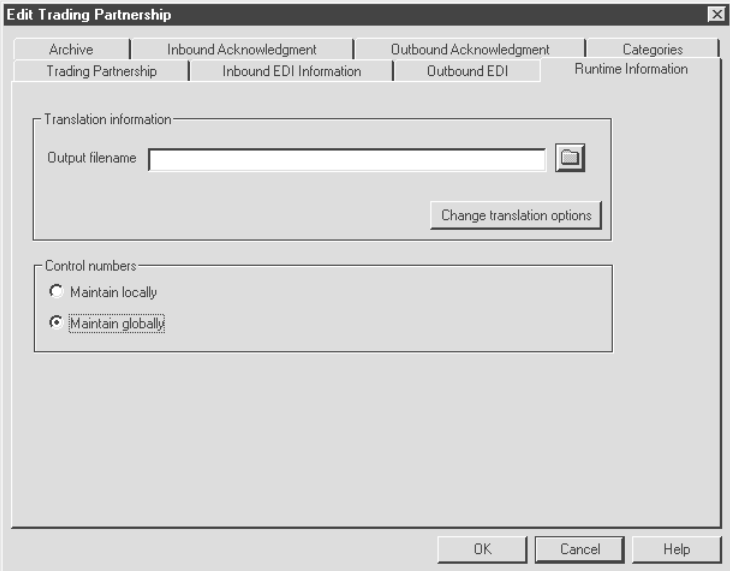
How to Set Translation Selections
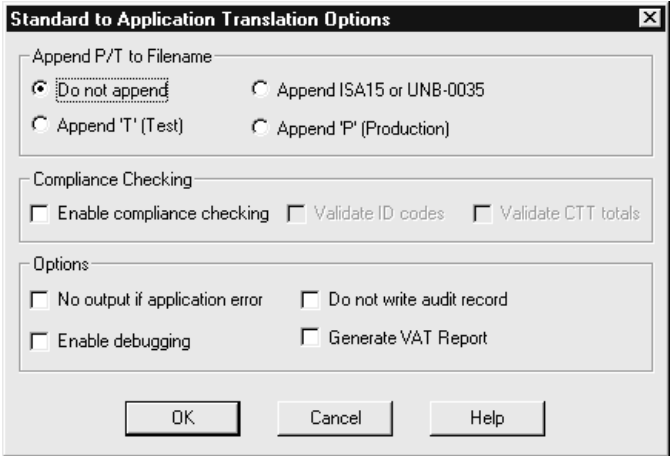
Introduction   This topic explains how to set certain translation selections for a specific Trading Partnership.

Note
The selections you choose in the Trading Partnership record override any default selections you may have set.

Reference
See How to Set Default Translation Selections for instructions on setting translation selections that your system automatically assigns when you create a new Trading Partnership record for a specific translation type.

Procedure   Use this procedure to set default translation selections for a translation type.

Step	Action
1	Open the Trading Partnership record that you want to change.
2	Click the Runtime Information tab.

Reference
See the topic The Trading Partnership Editor in this chapter for more information on the Runtime Information tab.

(Contd) Step	Action
3	Click the **Change translation options** button. **System Response** The system displays the Translation Options dialog box for the translation type selected for this Trading Partnership record. **Example** **Note** The selections on the dialog box depend on the type of translation.
4	Select the default selections that you want to apply to this type of translation.
5	Click **OK** to save your changes.

Making Mass Changes

The Mass Change Facility

Introduction

You use the Mass Change facility to modify multiple Interchange Organization, Group Organization, or Trading Partnership records at a time. The Mass Change facility is accessed from the Trading Partnership Administration window.

The access path for Mass Change

This illustration shows an example of how to access the Mass Change facility from Trading Partnership Administration.
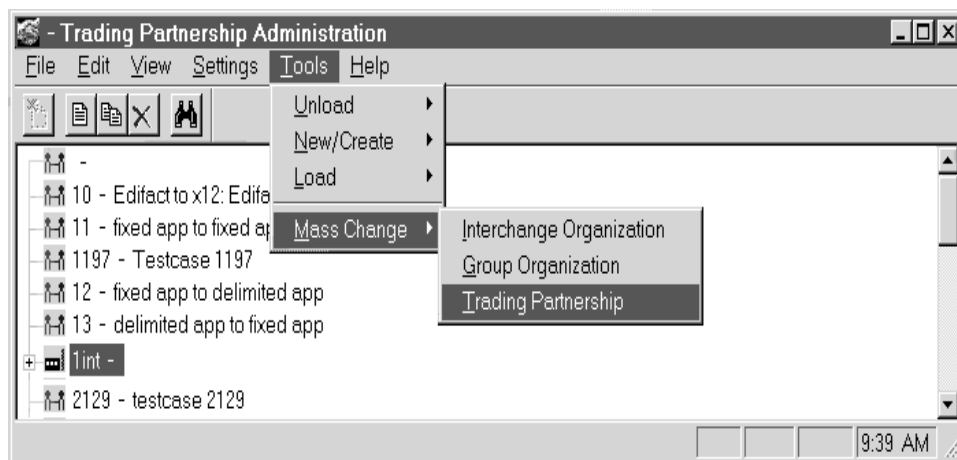
How to access Mass Change

Use this procedure to access the Mass Change facility.

Step	Action
1	Open Trading Partnership Administration.
2	Click the Interchange, Group, or Trading Partnership record you want to modify.
3	Click **Tools** on the menu bar.
4	Click **Mass Change** on the **Tools** menu.
	(Continued on next page)

(Contd) Step	Action
5	Use this table to determine your next action.

To change...	Click...
An Interchange Organization record	Interchange Organization
A Group Organization record	Group Organization
A Trading Partnership record	Trading Partnership

System Response
The TP Mass Change facility then displays the dialog box appropriate for the item you selected.

Reference
See the following topics in this chapter for more information about the different TP Mass Change options:

◗   How to Change Interchange Organization Records

◗   How to Change Group Organization Records

◗   How to Change Trading Partnership Records

How to Change Interchange Organization Records

Introduction

Changes to Interchange Organization records can be necessary in certain circumstances. You can use Mass Change to modify some or all of your Interchange Organization records at once, rather than one at a time.

The Interchange Organization Mass Change dialog box

This illustration shows an example of the Interchange Organization Mass Change dialog box.

Fields and functions

This table lists the fields of the Interchange Organization Mass Change dialog box and their functions.

Field	Function
Interchange IDs	
Change Your ID	Indicates that you are changing your interchange ID to the identifier you type in the box.
Change Partner's ID	Indicates that you are changing your trading partner's interchange ID to the identifier you type in the box. (Continued on next page)
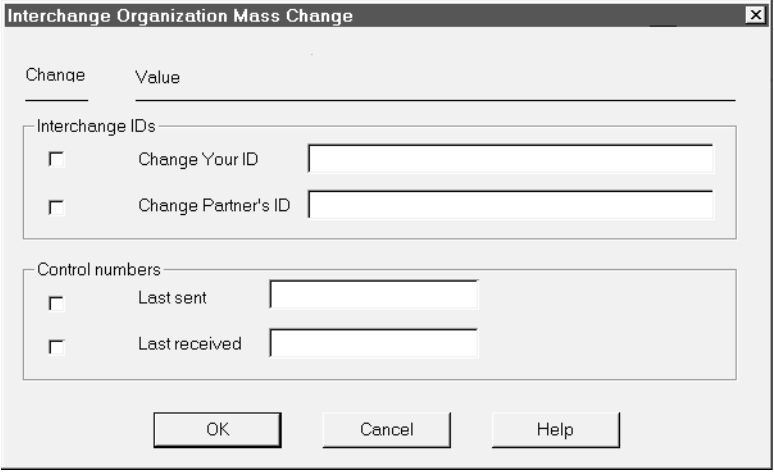
(Contd) Field	Function
Control numbers	
Last sent	Initializes the counter used for control numbers for your business documents.
Last received	Initializes the counter used for control numbers for business documents that you receive from this trading partner.
OK	Saves the changes you made on this dialog box.
Cancel	Closes this dialog box without saving your changes.
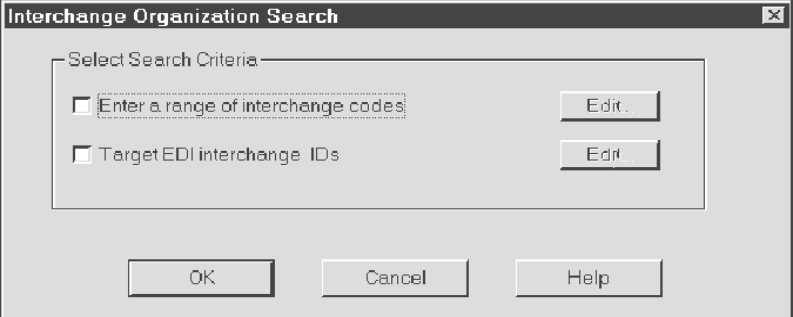
How to change multiple Interchange Organization records
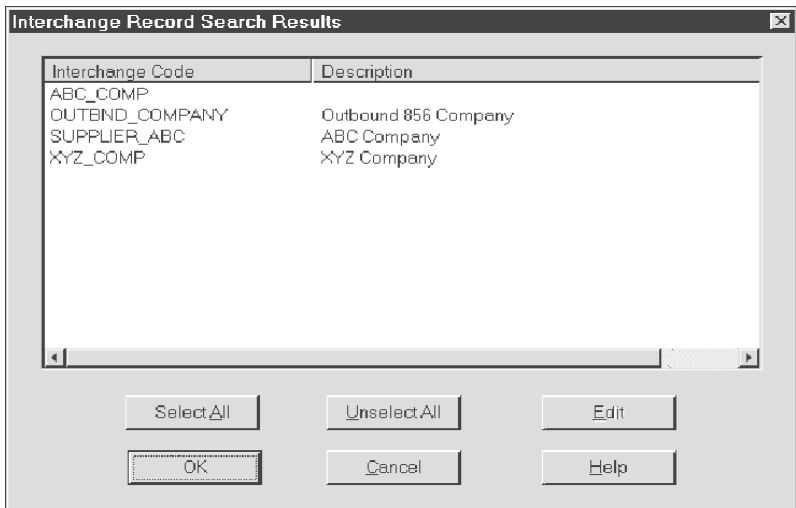
Use this procedure to change multiple Interchange Organization records at one time.

Step	Action
1	Open Trading Partnership Administration.
2	Click **Tools** on the menu bar.
3	Click **Mass Change** on the **Tools** menu.

(Contd) Step	Action
4	Click **Interchange Organization** on the **Mass Change** menu. **System Response** The TP Mass Change facility then displays the Interchange Organization Mass Change dialog box.
5	Use the table below to determine your next action

If you want to change...	Then click the check box for...
Your ID	**Change Your ID** and enter a new value into the text box
A Partner's ID	**Change Partner's ID** and enter a new value into the text box
The last control number sent	**Last sent** and enter a new value in to the text box.
The last control number received	**Last received** and enter a new value into the text box.

WARNING

If you click the check box for an item but do not enter a new value, the system will delete the existing value for that item.

(Contd) Step	Action
6	Click **OK**. **System Response** The system displays the Interchange Organization Search dialog box.
7	Use the table below to select the search criteria. | If you want to search a...	Then click the...
Range of Interchange IDs	**Enter Range** option.
Single Interchange ID	Target EDI Interchange ID option.	
8	Click **Edit**. **System Response** The system prompts for a range of Interchange IDs or a specific ID, depending on your selection in Step 5. (Continued on next page)

(Contd) Step	Action
9	Enter the Start and End codes or the specific ID(s) of the Interchange(s) you want to search and click **OK**. **System Response** The system then searches for and displays a list of records matching the code or codes entered
10	Use the table below to select the Interchange ID or IDs to which you want to apply the changes.

	To select...	Do this...
	A single ID	Click the ID, then click **OK**.
	All IDs displayed	Click **Select All**, then click **OK**.
	Multiple items	Hold the CTRL key down and click each ID you want to select. Then, release the CTRL key and click **OK**.

	System Response The system prompts you to confirm the requested changes.
11	Click **Yes**. **System Response** The system then changes the selected Interchange Organization records, creates a log file (*masschange.log*), and returns to Trading Partnership Administration.

How to Change Group Organization Records

Introduction   Changes to Group Organization records are not common, but they can be necessary in certain circumstances. You can use the TP Mass Change facility to change some or all of your Group Organization records at once, rather than one at a time.

The Group Organization dialog box   This illustration shows an example of the Group Organization Mass Change dialog box.

How to change multiple Group Organization records   Use this procedure to change multiple Group Organization records at one time.

Step	Action
1	Open Trading Partnership Administration.
2	Click **Tools** on the menu bar.
3	Click **Mass Change** on the **Tools** menu. (Continued on next page)

(Contd) Step	Action
4	Click **Group Organization** on the **Mass Change** menu. **System Response** The TP Mass Change facility then displays the Group Organization Mass Change dialog box.
5	Use the table below to determine your next action

If you want to change...	Then click the check box for...
Your ID	**Change Your ID** and enter a new value into the text box.
A Partner's ID	**Change Partner's ID** and enter a new value into the text box.
The last control number sent	**Last sent** and enter a new value in to the text box.
The last control number received	**Last received** and enter a new value into the text box.

WARNING

If you click the check box for an item but do not enter a new value, the system will delete the existing value for that item.

(Contd) Step	Action
4	Click **Group Organization** on the **Mass Change** menu. **System Response** The TP Mass Change facility then displays the Group Organization Mass Change dialog box.
5	Use the table below to determine your next action

If you want to change...	Then click the check box for...
Your ID	**Change Your ID** and enter a new value into the text box.
A Partner's ID	**Change Partner's ID** and enter a new value into the text box.
The last control number sent	**Last sent** and enter a new value in to the text box.
The last control number received	**Last received** and enter a new value into the text box.

WARNING

If you click the check box for an item but do not enter a new value, the system will delete the existing value for that item.

(Contd) Step	Action
6	Click **OK**. **System Response** The system displays the Group Organization Search dialog box.
7	Use the table below to select the search criteria.

If you want to search a...	Then click...
Range of Group IDs	**Enter Range**.
Single Group ID	Target EDI Group ID.

(Contd) Step	Action
8	Click **Edit**. **System Response** The system prompts for a range of Group IDs or a specific ID, depending on your selection in Step 1. (Continued on next page)

(Contd) Step	Action
9	Enter the Start and End codes or the specific ID(s) of the Groups you want to search and click **OK**. **System Response** The system then searches for and displays a list of records matching the code or codes entered.
10	Use the table below to select the Group ID or IDs to which you want to apply the changes.

To select...	Do this...
A single ID	Click the ID, then click **OK**.
All IDs displayed	Click **ALL**, then click **OK**.
Multiple items	Hold the CTRL key down and click each ID you want to select. Then, release the CTRL key and click **OK**.

	System Response The system prompts you to confirm the requested changes.
11	Click **Yes**. **System Response** The system then changes the selected records, creates a log file (*masschange.log*), and returns to Trading Partnership Administration.

How to Change Trading Partnership Records

Introduction   The Mass Change facility uses a dialog box very similar to the one used to create new Trading Partnership records. The difference is a check box located to the left of items on the dialog box. You can use Mass Change to modify only those items that have a corresponding check box.

The Mass Change Trading Partnership dialog box   This illustration shows an example of the Trading Partnership Mass Change dialog box.
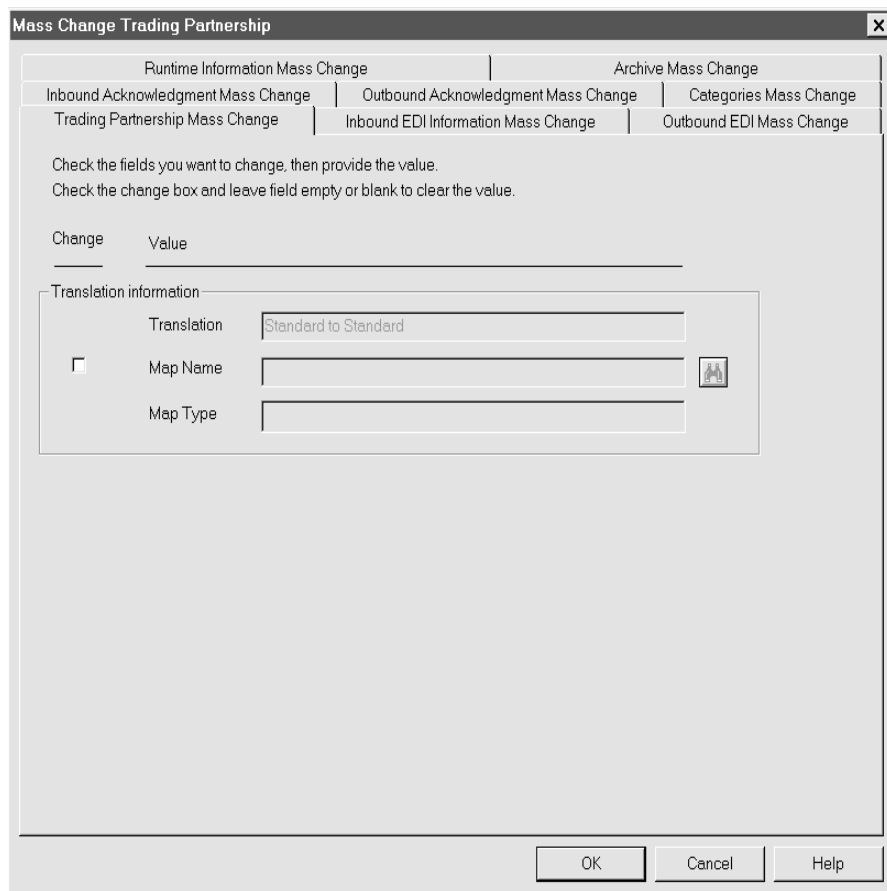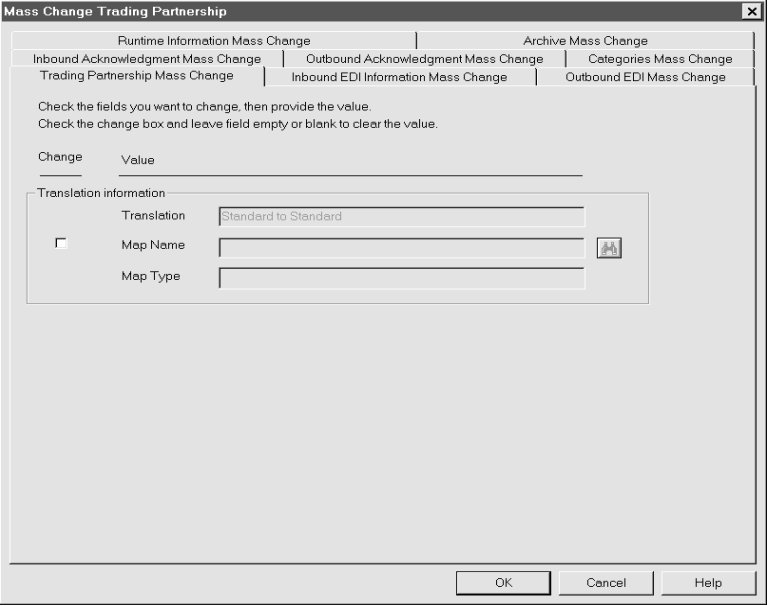
How to Access the Mass Change Trading Partnership dialog box

Use this procedure to open the Mass Change Trading Partnership dialog box.

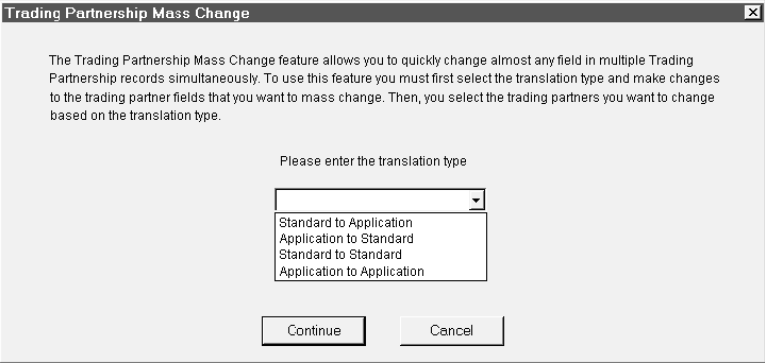| Step	Action
1	Open the Trading Partnership Administration window.
2	Click **Tools** on the menu bar.
3	Click **Mass Change** on the **Tools** menu. **System Response** The system displays a submenu of record types.
4	Click **Trading Partnership** on the **Mass Change** menu. **System Response** The system displays an introductory dialog box to the Trading Partnership Mass Change function. Trading Partnership Mass Change The Trading Partnership Mass Change feature allows you to quickly change almost any field in multiple Trading Partnership records simultaneously. To use this feature you must first select the translation type and make changes to the trading partner fields that you want to mass change. Then, you select the trading partners you want to change based on the translation type. Please enter the translation type Standard to Application Application to Standard Standard to Standard Application to Application Continue Cancel *(Continued on next page)*

(Contd) Step	Action
5	Click or enter the translation type of the Trading Partnership records you want to change.
6	Click **Continue**. **System Response** The system then displays the Mass Change Trading Partnership dialog box. You are now ready to select the values you want to change. **Reference** See the procedure under the topic <u>Modifying Trading Partnership Records</u> for more information about selecting values to change.

How to change Trading Partnership records

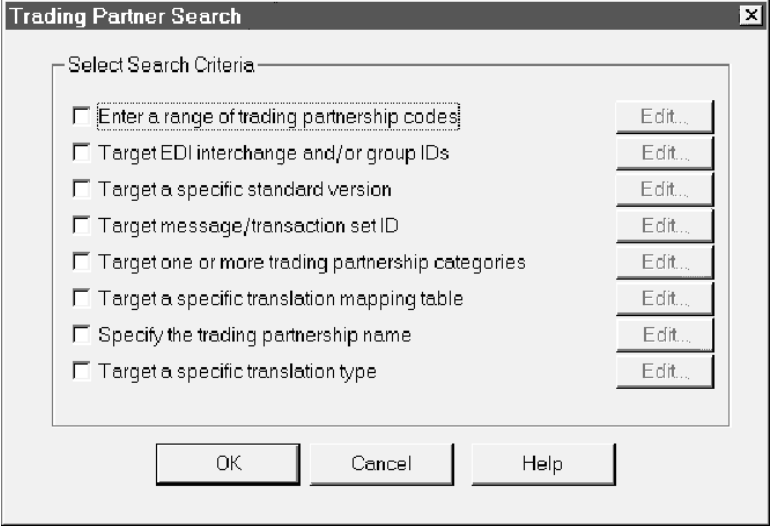Use this procedure to change multiple Trading Partnership records at one time.

Note
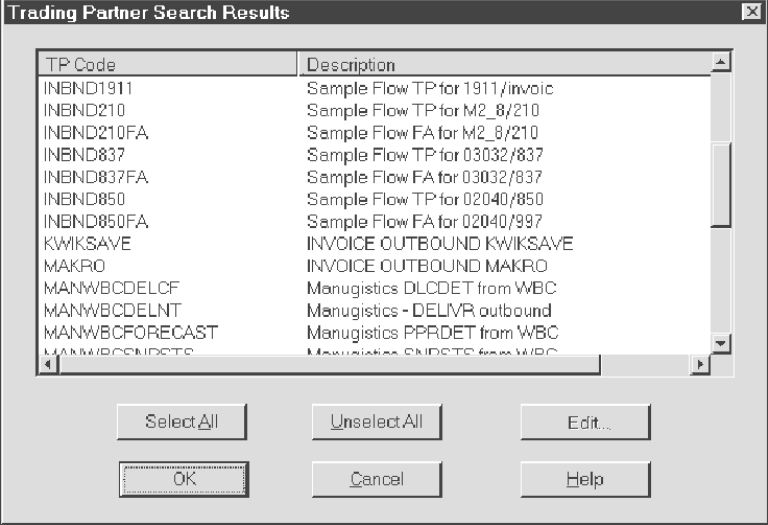This procedure begins with the Trading Partnership Mass Change dialog box active.

Reference
See How to Access the Mass Change Trading Partnership dialog box for more information about activating this dialog box.

Step	Action
1	Click the tab containing the information you want to change.
2	Click the first item you want to change and enter the new value. **Warning** If you click the check box for an item but do not enter a new value, the system will delete the existing value for that item. **Note** Some items have an Edit button to the right of the text box. Click the **Edit** button to display a dialog box listing additional items that you can change. You can then click the items you want to change and enter the appropriate values. Click **OK** when you are ready to return to the previous screen or dialog box.
3	Repeat Steps 5 and 6 until you finish marking all of the items you want to change. *(Continued on next page)*

(Contd) Step	Action
4	Click **OK**. **System Response** The system displays the Trading Partnership Search dialog box.
5	Click the check box of the selection criteria you want to use, then click the associated **Edit** button. **System Response** The system displays a secondary selection dialog box for the chosen criteria.
6	On the secondary dialog box, click the criteria you want to use and, where required, enter the value of the criteria. **Example** If you chose to select based on a range of numbers, you must enter a beginning and ending value. (Continued on next page)

(Contd) Step	Action
7	Click **OK**. **System Response** The system displays the Trading Partner Search Results dialog box.
8	Use the table below to select the Trading Partners to which you want to apply the changes.

To select...	Do this...
A single Trading Partner	Click the Trading Partner, then click **OK**.
All Trading Partners displayed	Click **Select All**, then click **OK**.
Multiple items	Hold the CTRL key down and click each Trading Partner you want to select. Then, release CTRL and click **OK**.

System Response
The system prompts you to confirm the requested changes.

| 9 | Click **Yes**.

System Response
The system then changes the selected records, creates a log file (*masschange.log*), and returns to Trading Partnership Administration. |

Running Translation

Contents

Overview

Introduction

In this chapter

This chapter covers running translation from the Translate menu.

This chapter explains:

◗   Inbound and outbound translation

◗   The Translate Menu

◗   The Translation process

◗   How to envelope output data that is in a standard format

◗   How to set up and run translations from the Translate menu

◗   How to view a translation summary.

Running translation from a script or batch file

To run translation automatically, include the lftran command in a batch file or script.

To run translations on a schedule, enter the lftran translation command and the appropriate parameters into a batch file or script, and then run the batch file or script in an unattended mode via the Task Scheduler.

References
For information about using the Task Scheduler to run batch files and scripts, see the chapter Using the Task Scheduler in this guide.

For information about the lftran command, see the chapter Command Reference in the *Gentran:Server for UNIX and Workstation Technical Reference Guide*.

About translation

Data translation is the primary function of Gentran:Server. Data translation takes business data in one computer-readable form and transforms it into another form.

During translation Gentran:Server executes the instructions defined in the:

◗   Translation object or compiled map

◗   Trading Partnership record

◗   Translation batch file, script, or translation dialog boxes.

Setting translation options

You can set translation options in several ways:

◗ Set default translation options for a particular type of translation in Trading Partnership Administration

◗ Set specific translation options for a particular Trading Partnership record. These override any default settings.

◗ Set options for a specific input file when you run translation from the Translate menu. These override any other settings.

Key terms

This table describes key terms used in this chapter.

Term	Description
archive	The process of copying and saving EDI data that has been processed.
audit record	A temporary audit file named *edistat.i* or *edistat.o*, created during translation, which contains the directory path and file name for the EDI data received or generated. The record does not contain the actual EDI data.
data file	The file that contains the inbound or outbound data.
envelope	The process of combining like interchanges.
functional acknowledgment (FA)	The standard transaction set used to acknowledge receipt of a transmission. The functional acknowledgment tells you if a document was received and processed by your trading partner and whether it contained EDI standard compliance errors. Likewise, the functional acknowledgment you send to your trading partner relays the same information to them.
historical record	The index files named *edihist.dat* and *edihist.idx* that store the location of the archived EDI data. The files are built from the audit records that translation produces. Together, they are the permanent audit file.
history audit directory	The EDI History Audit (*hisaud*) directory used to store permanent history files.

(Contd) Term	Description
inbound translation	A translation in which the input format of the source document is either an EDI standard format or an XML format that you use to exchange documents with a trading partner. **Note** Inbound translation using an XML format is available only if you have the XML translation option.
outbound translation	A translation in which the input format of your source document is either an application format or an XML format that you defined for your internal application documents. **Note** Inbound translation using an XML format is available only if you have the XML translation option.
reconciliation	The process of viewing functional acknowledgments, viewing data with the status of Errors, and correcting the EDI data.
status audit directory	The EDI Status Audit directory (stataud) used to store temporary audit files (*edistat.i* and *edistat.o*) and the history audit data.
Trading Partnership Code	A user-defined code that uniquely identifies a Trading Partnership record.
translation	The conversion of data from one format to another.

Inbound and Outbound Translation

Introduction   Translation is either inbound or outbound. The dialog boxes, translation objects, and options you use during translation differ for each type of translation.

Inbound translation   In an inbound translation, the source or input document is in a data transmission format. The data transmission format can either be an EDI standard or, if you have the XML translation option, an XML format that your trading partner uses to send you documents. These are the types of maps normally used in inbound translation:

◗   Standard-to-standard

◗   Standard-to-application

XML translation option

If your trading partner sends you files in XML format and you have the XML translation option, you can also run these types of inbound translation.

◗   Standard-to-XML (if you have the XML translation option)

◗   XML-to-application (if you receive the XML input documents from your trading partner)

◗   XML-to-XML (if you receive the XML input documents from your trading partner)

◗   XML-to-standard (if you receive the XML input documents from your trading partner)

ODBC translation option

If you have the ODBC translation option, you can define the output application file as an ODBC database format.

For example:

◗   Standard-to-application (ODBC)

Outbound translation

In an outbound translation, the source or input document is in a data format that you use for your internal applications. The input format of the source documents is either an application format or, if you have the XML translation option, an XML format that you defined for your internal application documents. These are the types of maps normally used in outbound translation:

◗　Application-to-application

◗　Application-to-standard

XML translation option

If you have the XML translation option, you can also run these types of outbound translations

◗　Application-to-XML

◗　XML-to-standard (if you send the output documents to your trading partner)

◗　XML-to-XML (if you send the output documents to your trading partner)

ODBC translation option

If you have the ODBC translation option, you can define the input application file as an ODBC database format.

For example:

◗　Application (ODBC)-to-standard

◗　Application (ODBC)-to-XML

The Translate Menu

Translate menu   This illustration shows the **Translate** menu.

```
Translate
   Documents...
   Translation Summary...
```
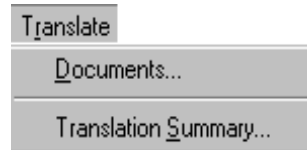
Translation menu options and functions

This table describes the options of the **Translate** menu and their functions.

Option	Function
Documents	Translates both input data files that are coming in to your system and input files that are leaving your system. Inbound data files are in a standard format and are usually translated to your application format. Outbound data files are usually in your application format and are usually translated into an EDI standard format. **Note** If you have the XML translation option, then your inbound data files can also be in XML format and translated into an XML format.
Translation Summary	Displays selected information about translations.

The Translation Process

Introduction
This topic describes the Gentran:Server data translation process. You can use translation to:

◗ Test a translation object (compiled map)

◗ Convert data to send to a trading partner

◗ Convert data received from a trading partner.

Translation process flow
This illustration shows the translation process flow if all options are used.

```
                        ┌──────────┐
                        │  Input   │
                        │  Data    │
                        └────┬─────┘
                             │
                             ▼
                        ┌──────────┐
                        │Formatting│ (Inbound translation)
                        └────┬─────┘
        ┌───────────┐        │
        │Translation├──┐     ▼
        │Object     │  │ ┌──────────┐
        └───────────┘  └─┤Translating├──────┐
        ┌───────────┐  ┌─┤          │      │
        │Trading    ├──┘ └────┬─────┘      │
        │Partnership│         │            ▼
        │Record     │         ▼        ┌────────┐
        └───────────┘    ┌──────────┐  │ Output │
    (Inbound translation)│Enveloping├─►│ Data   │
                         └────┬─────┘  └────────┘
                              │
                              ▼
                         ┌──────────┐
                         │Archiving │
                         └──┬────┬──┘
                      ┌─────┘    └─────┐
                      ▼                ▼
                 ┌────────┐       ┌────────┐
                 │Archive │       │Archive │
                 │Data    │       │History │
                 └────────┘       └────────┘
```
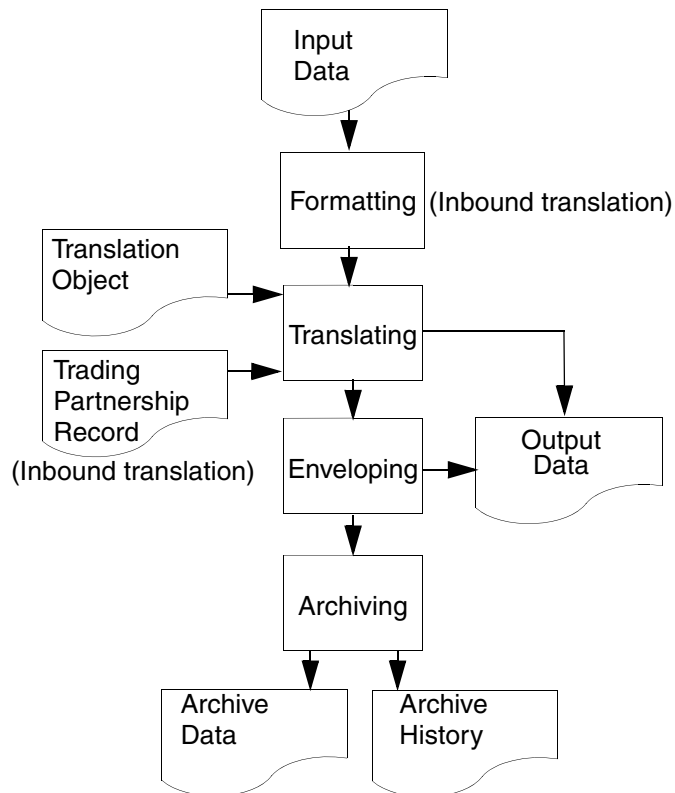
**Example:
Inbound
translation
process stages**

This example describes the typical stages in the default inbound translation process for X12 data that is being translated to application data.

Stage	Description
1	Gentran:Server determines the Trading Partnership Code.
2	Gentran:Server reads the Trading Partnership record.
3	Gentran:Server determines whether outbound acknowledgments are required. ◗ If YES, the system generates a Trading Partnership enveloping signature that includes an ACK component for each transaction set and then continues with Stage 4. ◗ If NO, the system continues with Stage 4.
4	The system translates the sets, generates the functional acknowledgments (if required), and generates the output data.
5	The system generates the inbound EDI audit record for the set (if your system is configured to archive EDI data) and then continues with the next stage.
6	Are there more sets to process? ◗ If YES, repeat Stages 1 through 5. ◗ If NO, continue with Stage 7.
7	Gentran:Server locks the Trading Partnership record.
8	Gentran:Server: ◗ Generates the interchange, group, and set headers ◗ Replaces the enveloping signature with the generated headers.
9	Gentran:Server generates the outbound audit records for the functional acknowledgments.
10	Gentran:Server unlocks the Trading Partnership records.

**Example:
Outbound
translation
process stages**

This table describes the typical stages in the default outbound translation process for an application-to-standard translation.

Stage	Description
1	Gentran:Server determines the Trading Partnership Code.
2	Gentran:Server reads the Trading Partnership record.
3	Gentran:Server generates a Trading Partnership enveloping signature for the transaction set.
4	Gentran:Server translates the set and generates the output data, which includes the enveloping signature.
5	Are there more sets to process? ◗ If YES, repeat Stages 1 through 4. ◗ If NO, continue with Stage 6.
6	Gentran:Server uses the Trading Partnership enveloping signature to find and lock the Trading Partnership records.
7	Gentran:Server generates the interchange, group, and set headers.
8	Are the header segments mapped? ◗ If YES, merge the mapped header with the one generated in Stage 7 and then continue with Stage 9. ◗ If NO, continue with Stage 9.
9	Gentran:Server uses the enveloping signature to envelope the output data.
10	Gentran:Server updates the control numbers in the Trading Partnership records.
11	Gentran:Server generates the outbound audit records.
12	Gentran:Server unlocks the Trading Partnership records.

Enveloping Outbound Data

EDI Envelopes

Envelope segments

The EDI transaction sets and functional acknowledgments that you send to a trading partner must include envelope segments. The purpose of an envelope is to ensure that the sender and receiver of the transactions are properly identified and routed properly.

Each envelope level contains two data segments:

◗   A header segment to identify the beginning of the envelope

◗   A trailer segment to signal its end.

Levels

There are three possible levels of envelopes:

◗   Interchange envelope

◗   Group envelope

◗   Transaction set envelope

The interchange and transaction set envelopes are mandatory for all standards. The group envelope is optional (or not used at all) for many EDI standards.

One interchange envelope can contain multiple groups. One group can contain multiple transaction sets of the same type.

Interchange envelope

The interchange envelope is the outer envelope. The header segment usually contains data elements that specify who the sending party is and the final destination.

Example
The ISA segment is an interchange envelope segment for the ASC X12 standard.

Group envelope

The group envelope is the second-level envelope.

The group header segment begins a group. Within the group segments are data fields that identify the type of transactions contained in the group, the sender and receiver's application IDs, the transmission date, control number, and other standard reference data.

Example
The GS segment is the functional group envelope segment for the ASC X12 standard.

Transaction set envelope

Each transaction is surrounded by a transaction envelope. All detail segments are stored in between the transaction set header and trailer segments.

Example
The ST segment is the transaction envelope for ASC X12 standard.

Location of Enveloping Information

Introduction

Gentran:Server builds the interchange, group, and transaction set envelopes from information defined in specific Trading Partnership records. During the enveloping process, Gentran:Server inserts the envelope information into the stream of data that you are sending your trading partner.

Record/envelope information table

This table shows the records that Gentran:Server uses to envelope outbound EDI data and lists the type of enveloping information each record contains.

Record	Enveloping Information
Interchange Organization record	◗ Your unique interchange identifiers, which are inserted into the Interchange envelope. ◗ Your trading partner's unique interchange identifiers, which are inserted into the Interchange envelope. ◗ The counter used for control numbers for your outbound business documents. ◗ The counter used for control numbers for inbound business documents that you receive from this trading partner.
Group Organization record	◗ A unique Group Organization code that is inserted into the Functional Group envelope. ◗ Your unique group identifier that is inserted into the Functional Group envelope. ◗ Your trading partner's unique Group identifier that is inserted into the Group envelope. ◗ The counter used for control numbers for your business documents. ◗ The counter used for control numbers for inbound business documents that you receive from this trading partner. (Continued on next page)

(Contd) Record	Enveloping Information
Outbound EDI tab of the Trading Partnership record	◗ The type of header segment used for the Interchange envelope. ◗ The type of group segment used for the Functional Group envelope.
Outbound Acknowledgment tab of the Trading Partnership record	◗ The type of header segment used for the Interchange envelope. ◗ The type of group segment used for the Functional Group envelope. ◗ The check box that turns the envelope program for outbound acknowledgments on and off.

Reference

See the How to Modify the Outbound Envelope Segments topic in this chapter for instructions on editing the elements in the Interchange envelope segment.

Enveloping Options

Introduction

You have two options for enveloping your outbound EDI data:

◗ Envelope transactions as part of the translation process

◗ Collect output transactions and delay enveloping until a later time

Setting enveloping options

You set enveloping options in the **Iftran** command:

◗ In a translation script or batch file, or

◗ From the Translation Options dialog box if you run translation in attended mode from the Translate menu

Reference
See Setting Enveloping Options in a Script or Batch File or Setting Envelope Options from the Translate Menu.

Enveloping during translation

This table shows the stages in the enveloping process.

Stage	Description
1	The translator determines that the translation output will be in an EDI standard format.
2	During translation, the **Iftran** program adds an envelope signature line to the beginning of each set and generates an output file.
3	The **Iftran** program uses the envelope signature lines in the output file to locate and lock the appropriate Trading Partnerships records.
4	If Iftran is unable to lock the Trading Partnership records, it produces a lock file. ◗ If the translation output does not include functional acknowledgments, the sets that were not enveloped are written to boxout.lok in the temp directory. ◗ If the translation output includes functional acknowledgments, the sets and functional acknowledgments that were not enveloped are written to sigout.lok in the temp directory. *(Continued on next page)*

(Contd) Stage	Description
5	The **lftran** program uses the envelope signature and the information in the Trading Partnership records to generate interchange, group, and set header segments.
6	If lftran encounters an error (other than a lock) that prevents it from generating the envelope segments, it produces an error file. ◗ If the translation output does not include functional acknowledgments, the sets that were not enveloped are written to boxout.err in the temp directory. ◗ If the translation output includes functional acknowledgments, the sets and functional acknowledgments that were not enveloped are written to sigout.err in the temp directory.
7	If the header segments are mapped, the system merges the mapped header with the generated header segment.
8	The system: ◗ Uses the header and trailer segments to envelope the data in the outbound EDI file ◗ Generates the audit record ◗ Updates the control numbers in the Trading Partnership records.
9	The system releases the locked Trading Partnership records.
10	The system looks for more sets. If it finds more sets, it repeats the process. If it does not find any more sets to translate, the translation process stops. **Note** The translation scripts supplied with Gentran:Server include commands to save *.lok files so that you can reprocess them. **Reference** For solutions on processing the sigout.lok file, see the readme file and supplementary white papers for this release of Gentran:Server.

Delayed enveloping

This table shows the stages in the delayed enveloping process.

Stage	Description
1	The translator determines that the translation output will be in an EDI standard format.
2	The **lftran** program runs with the H0 option to add a delayed envelope signature line to the beginning of each set in the output data. The program does not add envelope segments to the output data. **Note** The lftran program can run many times throughout the day.
3	The system collects the output data.
4	The **lftran** program runs with the H1 option at a later time, usually according to a schedule. The program processes the collected output data, using the envelope signature lines to locate and lock the appropriate Trading Partnerships records.
5	If lftran is unable to lock the Trading Partnership records, it produces a lock file. The system writes the sets and functional acknowledgments that were not enveloped to sigout.lok in the temp directory.
6	The **lftran** program uses the envelope signature and the information in the Trading Partnership records to generate interchange, group, and set header segments.
7	If lftran encounters an error that prevents it from generating the envelope segments, it produces an error file. The system writes the sets and functional acknowledgments that were not enveloped to sigout.err in the temp directory.
8	If the header segments are mapped, the system merges the mapped header with the generated header segments. (Continued on next page)

(Contd) Stage	Description
9	The system: ◗ Uses the header and trailer segments to envelope the data in the outbound EDI file ◗ Generates the audit record ◗ Updates the control numbers in the Trading Partnership records.
10	The system releases the Trading Partnership records and waits for the next translation and enveloping cycle. **Note** The translation scripts supplied with Gentran:Server produce a boxout.lok file. If you want to reprocess the sets in this file, you must modify the translation script to re-route the file. **Reference** For solutions on processing the sigout.lok file, see the readme file and supplementary white papers for this release of Gentran:Server.

Setting Enveloping Options in a Script or Batch File

Introduction

In production mode, you will probably run the **lftran** command with translation and enveloping options in a script or batch file.

Settings for the translation script

This table describes the lftran option you must set in your script or batch file.

IF you want to…	THEN…
Envelope the EDI transaction sets during translation	Do not set an enveloping option in **lftran**, because this is the default enveloping action.
Collect translated files to envelope later (delay enveloping)	Configure your system to run the **lftran** program twice: ◗ First with the H0 option to generate the output file with the signature lines. You can run this option throughout the day. ◗ Later with the H1 option to lock the Trading Partnership records, generate the envelope segments from the signature lines in the collected output files, and then envelope all the collected data.

Handling the lftran output

The translation scripts supplied with Gentran:Server produce a boxout.lok file. If you want to reprocess the sets in this file, you must modify the translation script to re-route the file through the translation and enveloping process.

For solutions on processing the sigout.lok file, see the readme file and supplementary white papers for this release of Gentran:Server.

Combining like interchange envelopes

If you want to combine like interchange envelope segments into one interchange envelope segment after translation, you must include the **envelope** command in a batch file or script. However, if you are translating with the H1 option and the input files are organized by Trading Partnership code, you can omit the **envelope** command.
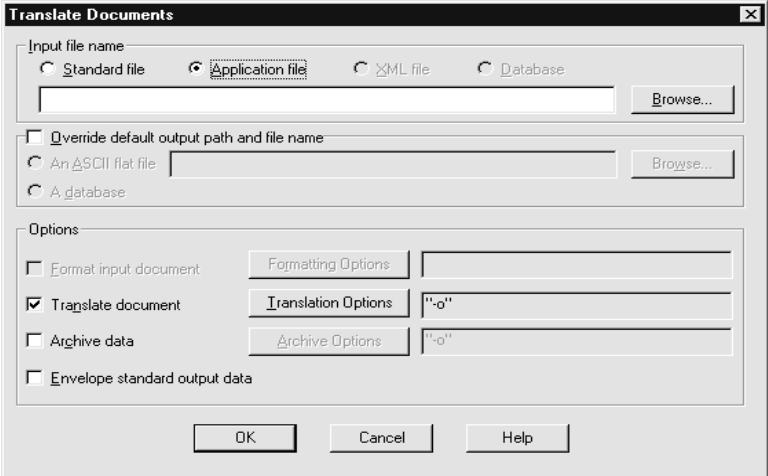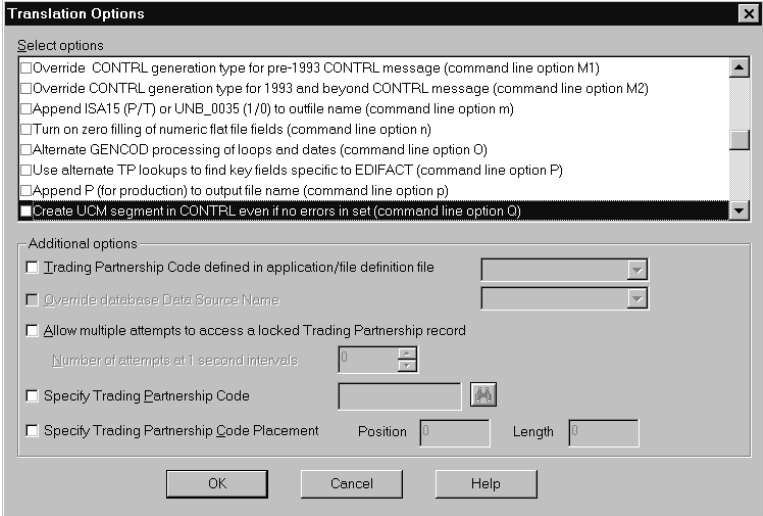
Reference
See the description of the **envelope** command in the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for details.

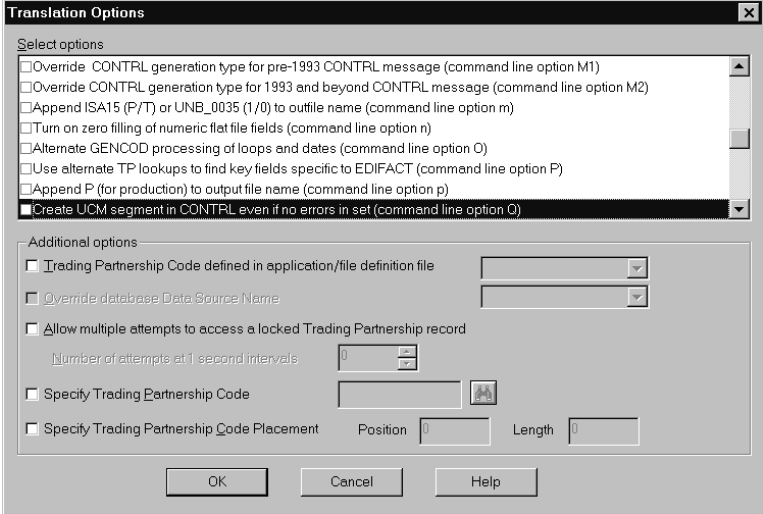# Setting Envelope Options from the Translate Menu

Introduction   If you run translation from the Translate menu, you can set the envelope options along with other translation selections on the Translation Options dialog box.

Procedure   Use this procedure to set enveloping options from the Translate menu.

Step	Action
1	Click **Documents** on the **Translate** menu. **System Response** Gentran:Server displays the Translate Documents dialog box. **Reference** See The Translate Documents Dialog Box for a description of the fields on this dialog box.
2	Select the type of input file. (Continued on next page)

(Contd) Step	Action
3	Select **Translate document** and then click the **Translation Options** button. **System Response** Gentran:Server displays the Translation Options dialog box. **Reference** See The Translation Options Dialog Box for a description of the fields on this dialog box.
4	Select the enveloping options.

IF you want to…	THEN…
Envelope the EDI transaction sets now	Do not select an enveloping option.
Collect translated files to envelope later	Select the H0 option to generate an output file that has Trading Partnership signatures.
Envelope collected output files so that the EDI transactions are ready to send	Select the H1 option to lock the Trading Partnership records and then generate the envelope segments from the signatures in the output files. *(Continued on next page)*

(Contd) Step	Action
3	Select **Translate document** and then click the **Translation Options** button. **System Response** Gentran:Server displays the Translation Options dialog box. **Reference** See The Translation Options Dialog Box for a description of the fields on this dialog box.

4	Select the enveloping options.

IF you want to…	THEN…
Envelope the EDI transaction sets now	Do not select an enveloping option.
Collect translated files to envelope later	Select the H0 option to generate an output file that has Trading Partnership signatures.
Envelope collected output files so that the EDI transactions are ready to send	Select the H1 option to lock the Trading Partnership records and then generate the envelope segments from the signatures in the output files. (Continued on next page)

(Contd) Step	Action
5	Are you running the H1 option with data produced from the H0 option? ◗ If YES, continue with the next step. ◗ If NO, go to Step 7.
6	Do you want to combine like interchanges into a single interchange immediately after translation? ◗ If YES, click the **Envelope standard output data** check box to enable the **envelope** program. ◗ If NO, continue with the next step. **Reference** See the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for information about the envelope command.
7	Select the other translation options that you want the system to use for this translation; then click **OK** to save your changes and return to the Translate Documents dialog box. **CAUTION** **If you set the translation options both here and in the Trading Partnership record,** Gentran:Server **uses those set in the Trading Partnership record if it finds a conflict. If you set nonconflicting options in both places, the lftran program uses both sets of options.**
8	Set any remaining options you want to use for this translation on the Translate Documents dialog box and then click **OK** to start translation.

Setting Up Translation

Introduction

Before you begin　Before you can run a translation in Gentran:Server, you need to be sure that all of the input files for the process are available. Those files include:

◗　Translation object or compiled map

◗　Trading partnership records

◗　Input data file

References　This table provides reference information for the items listed.

For...	See...
Procedures on preparing the compiled map (translation object) that furnishes the specifications for a specific data translation	Designing your Map in this guide.
Procedures for creating a Trading Partnership record for outbound translation	Working with Trading Partnerships in this guide.
Procedures for creating a Trading Partnership record	Working with Trading Partnerships in this guide.

The Translate Documents Dialog Box

Introduction   This topic describes the Translate Documents dialog box. The Translate
Documents dialog box enables you to:

- ◗   Select the input file you want to translate
- ◗   Set formatting options for the input document
- ◗   Set translation options for the input document
- ◗   Set archiving options
- ◗   Enable enveloping when the output file is in a standard format.

Illustration   This illustration shows the Translate Documents dialog box for an input document
in a standard format.
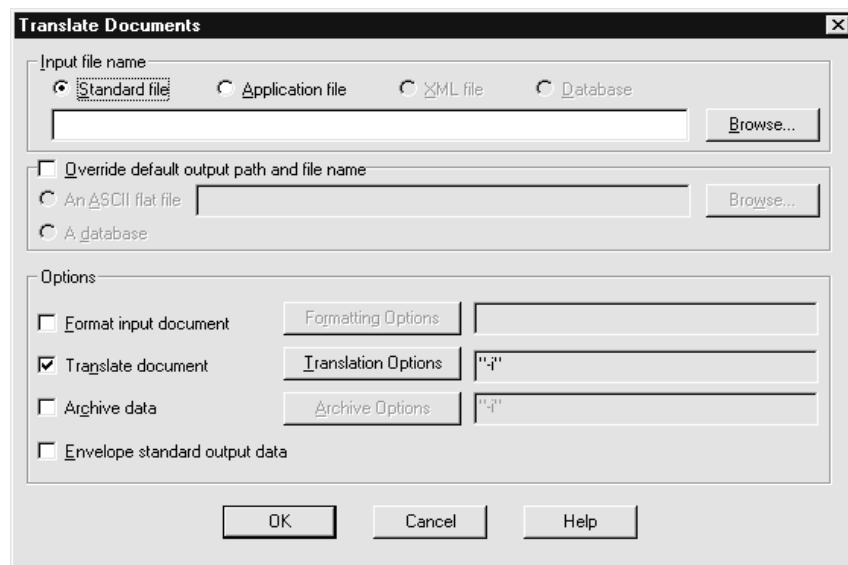
Note
The Translate Documents dialog box retains the parameter settings from the last
translation performed.

Fields and functions

This table describes the Translate Documents dialog box and their functions. The available options depend upon the type of input file selected.

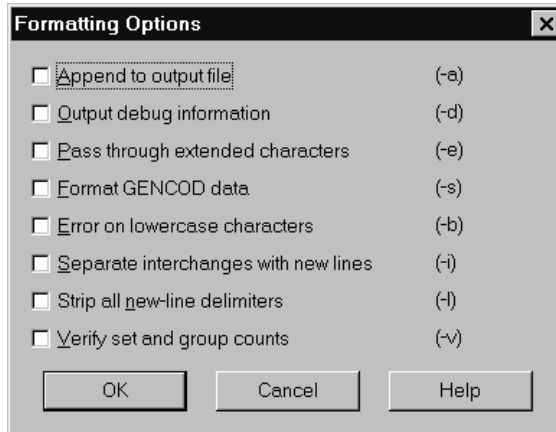| Field	Function
Input file name	
Standard file	Specifies that the input file is in an EDI standard file format (inbound).
Application file	Specifies that the input file is in an application file format (outbound).
XML File	Specifies that the input file is in XML file format. **Note** This option is enabled only if you have the Gentran:Server XML translation option.
Database	Specifies that the input file is in a database format.
Input file name box	Used to enter the path and name of the file you want to translate. **Notes** Do not enter environmental variables in this field. You can use the **Browse** button to locate the file.
Browse	Opens a search dialog box, which enables you to browse the folders to locate the input file name.
Override default output path and file name	Overrides the default location to store the translated data. The default location is specified in the Trading Partnership record. **Note** You should not enter environmental variables in this field.
An ASCII flat file	Writes the result of the translation to an ASCII flat file when this option and the **Override default output path and filename** is selected.
A database	Writes the result of the translation to a database file when this option and the **Override default output path and filename** is selected. (Continued on next page)

(Contd) Field	Function
File name box	When the **An ASCII flat file** option is selected, this box enables you to enter the file and directory where you want Gentran:Server to store the translated data.
Browse	Opens a search dialog box, which enables you to browse the folders to locate the ASCII flat file or database file.
Options	
Format input document	Prepares the inbound EDI data for translation by replacing segment terminators with newline characters. **Notes** Formatting is required for inbound EDI translation. This option is not available if you specified that the input file is in XML format.
Formatting Options	Displays the Formatting Options dialog box to allow you to fully control the formatting process. **Notes** This option is not available if you specified that the input file is in XML format. **Reference** See the <u>Formatting Options dialog box</u> topic.
Translate document	Runs the translation process on the input file. When not checked, you can format, envelope, or archive previously translated data.
Translation Options	Displays the Translation Options dialog box, which enables you to choose from many command line translation options. **Note** The system displays the default translation options in the box next to this button. **References** See <u>The Translation Options Dialog Box</u> topic. See the **Iftran** topic in the Command Reference chapter of the *Technical Reference Guide* for a list of all options.
	(Continued on next page)

(Contd) Field	Function
Archive Data	Archives the EDI data and the status. **Note** To use the functional acknowledgment features, you must archive the data with each translation. This option is not available if you specified that the input file is in XML format.
Archive Options	Displays the Archive Options dialog box, which enables you to choose **ediarc** command line options for the archiving inbound translation. **Notes** The system displays the default archive options in the box next to this button. This option is not available if you specified that the input file is in XML format. **Reference** See The Archive Options Dialog Box topic.
Envelope standard output data	Combines like interchanges for outbound functional acknowledgments created during an inbound translation. Combines like interchanges into one envelope during outbound translation. **Notes** This option is not available if you specified that the input file is in XML format. You can use delayed enveloping instead of this option.

The Formatting Options Dialog Box

Formatting Options dialog box

This illustration shows the Formatting Options dialog box. This dialog box controls how the input file is reformatted for output

Formatting Options fields and functions

This table describes the fields in the Formatting Options dialog box and their functions.

Note
The letters in parentheses are the **edifrmat** command line parameters that correspond to the description for each check box

Field	Function
Append to output file	Appends the results of the format process to previous output instead of overwriting the output.
Output debug information	Creates a list to help resolve issues you are having with inbound translation. The list shows the sets and segments you are translating and is stored in *edifrmat.log* under the reports directory. *(Continued on next page)*

(Contd) Field	Function
Pass through extended characters	Leaves unchanged any extended ASCII characters found in the input file. **Comment** If you do not check this option, **edifrmat** changes all extended ASCII characters to blanks.
Format GENCOD data	Retains original segment terminator instead of substituting new line terminator. Used for routing EDIFACT, GENCOD, or VDA data with product levels Gentran:Server with PCM or higher. **Note** The translator requires new line terminated data. This option is used only with data managers to route data that uses segment terminators.
Error on lowercase characters	Generates an error when lowercase characters are found within the input file.
Separate interchanges with newlines	Places newline characters between interchanges within the input data.
Strip all new-line delimiters	Removes all newline characters from the input file.
Verify set and group counts	Verifies the set and group counts found in the incoming data. The set count is compared to the number contained in the group trailer (for example, in the GE or UNE segments). The group count is compared to the number contained in the interchange trailer (for example, in the IEA, EG, ICE, or UNZ segments).

The Translation Options Dialog Box
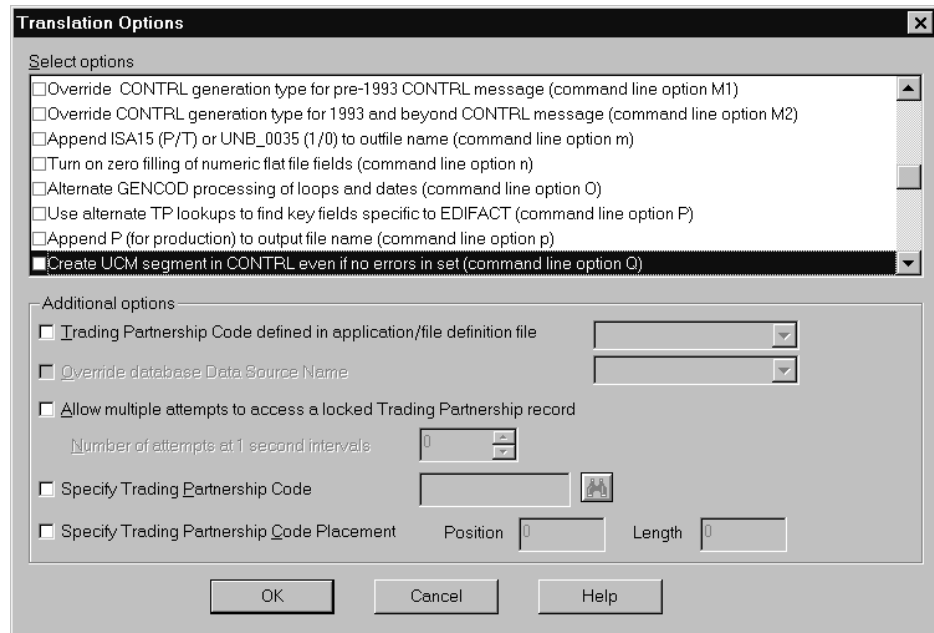
Setting translation options

If you want to use the same translation options for all your Trading Partnerships, you can set translation options as parameters on the translation command. If you want to use different translation options for some Trading Partnerships, set up translation options in the Trading Partnership record. (See Working with Trading Partnerships in this guide.) If you set translation options in both places, the program uses both sets of options unless they conflict. If some options conflict, Gentran:Server uses those set in the Translation Options dialog box.

Note
Gentran:Server saves the options from the last run. You do not have to enter them again for subsequent runs unless you want to make additional changes.

Translation Options dialog box

This illustration shows the Translation Options dialog box. This illustration is for inbound translation. The available options are different for outbound translation.

Translation Options fields and functions

This table describes the fields of the Translation Options dialog box and their functions.

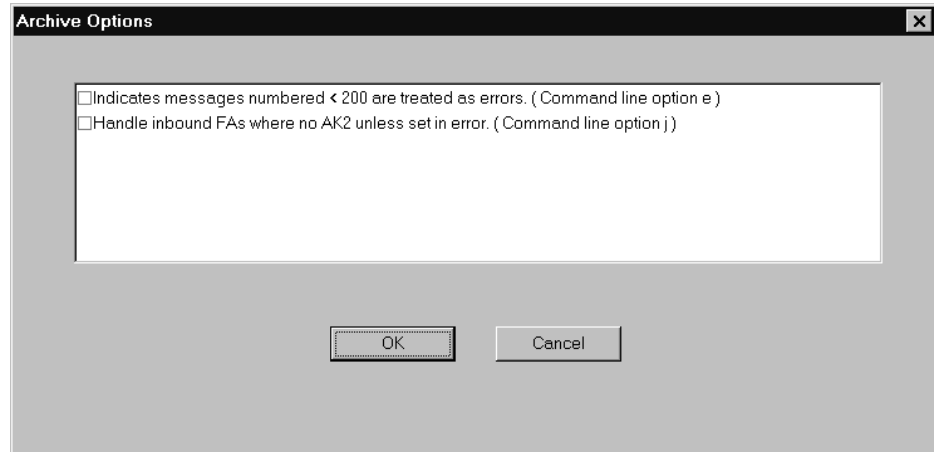| Field	Function
Select Options	Enables you to select parameters for the translation process. **Reference** See the table in the Select Options list topic in the online help system for a description of the available choices.
Trading Partnership Code defined in application/file definition file	Enables you to specify the application file or file definition that defines the location of the Trading Partnership code in the data you are translating. For outbound translation only.
Override database Data Source Name	For Visual Mapper only, enables you to replace the ODBC DSN used to create the application file with the one you want to use for the current translation. **Note** Your Gentran:Server system must have the optional ODBC translation capabilities.
Allow multiple attempts to access a locked Trading Partnership record	Sets the number of attempts Gentran:Server can make to access a locked Trading Partnership record. If the file is still locked after the specified number of attempts, the translation process fails.
Number of attempts at 1 second intervals	Sets the number of lock attempts.
Specify Trading Partnership Code	Specifies the Trading Partnership Code to be passed to the translator during translation. Use this option when the Trading Partnership Code is not present in the application data and the entire file can be translated using one Trading Partnership Code and one map. For outbound translation only.
Specify Trading Partnership Code placement	Specifies the location of the Trading Partnership Code within the input file. For outbound translation only.

Select Options list

For a complete and current description of the choices available in the Select Options list, view the **Select Options list** topic in the online help system.

The Archive Options Dialog Box

Archive Options dialog box

This illustration shows the Archive Options dialog box. This dialog box controls how the input file is archived

Archive Options fields and functions

The options listed on this dialog box are based on the type of input file. The **ediarc** command line parameter is listed in parentheses for each option.

Reference
See the Command Reference chapter of the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for a list of the **ediarc** parameters and variables.

Running Inbound and Outbound Translation

How to Run Inbound Translation

Introduction   This topic explains how to run an inbound translation from the Translate menu.

Note
You must be using a map with either an EDI standard or, if you have the XML translation option, XML as the format of the source document.

Multiple sessions   If you have Gentran:Server for UNIX and are running multiple sessions of the client, you can run translation in only one client session for each host and environment combination. You can translate documents in more than one parallel session as long as each session is for a different host and environment combination.

If you attempt to translate documents when another session that is logged in to the same host and environment is already translating, Gentran:Server displays a message to let you know that the process is currently running and asks that you try again later.

If you have Gentran:Server Workstation and are running multiple sessions on your desktop, you can run translation in only one session.

Procedure   Use this procedure to perform an inbound translation from the Translate menu.

Step	Action
1	Click **Documents** on the Translate menu. **System Response** Gentran:Server displays the Translate Documents dialog box. **Note** The Translate Documents dialog box retains the parameter settings from the last translation performed. (Continued on next page)

(Contd) Step	Action
2	Select the type of input file: ◗ Standard base file ◗ XML file (available only if you have the XML translation option)
3	Use one of the following methods to specify the input file name. ◗ Enter the full path and file name into the text box. ◗ Click **Browse** to display the Translation Input File dialog box, browse through the directories and select a file name, and then click **OK**.
4	Use this table to decide your next step. <table><tr><td>**IF you want to…**</td><td>**AND…**</td><td>**THEN…**</td></tr><tr><td>Store the data in the location specified in the Trading Partnership record</td><td>The input file is in EDI Standard format</td><td>GO TO Step 7.</td></tr><tr><td>Store the data in the location specified in the Trading Partnership record</td><td>The input file is in XML format **Note** The XML format is available only if you have the XML translation option.</td><td>GO TO Step 8.</td></tr><tr><td>Store the data in a different location</td><td>The output file is an ASCII flat file format</td><td>GO TO Step 5.</td></tr><tr><td>Store the data in a different location</td><td>The output file is a database file</td><td>GO TO Step 6.</td></tr></table>
5	Click **Override default output path and filename** and then select the ASCII flat file option. Enter the full path and file name of the ASCII flat file into the text box. **Note** You can also click **Browse** to display the Translation Output File dialog box to select a file name.
6	Click **Override default output path and filename** and then click the database option. (Continued on next page)

(Contd) Step	Action
2	Select the type of input file: ◗ Standard base file ◗ XML file (available only if you have the XML translation option)
3	Use one of the following methods to specify the input file name. ◗ Enter the full path and file name into the text box. ◗ Click **Browse** to display the Translation Input File dialog box, browse through the directories and select a file name, and then click **OK**.
4	Use this table to decide your next step. <table><tr><td>**IF you want to…**</td><td>**AND…**</td><td>**THEN…**</td></tr><tr><td>Store the data in the location specified in the Trading Partnership record</td><td>The input file is in EDI Standard format</td><td>GO TO Step 7.</td></tr><tr><td>Store the data in the location specified in the Trading Partnership record</td><td>The input file is in XML format **Note** The XML format is available only if you have the XML translation option.</td><td>GO TO Step 8.</td></tr><tr><td>Store the data in a different location</td><td>The output file is an ASCII flat file format</td><td>GO TO Step 5.</td></tr><tr><td>Store the data in a different location</td><td>The output file is a database file</td><td>GO TO Step 6.</td></tr></table>
5	Click **Override default output path and filename** and then select the ASCII flat file option. Enter the full path and file name of the ASCII flat file into the text box. **Note** You can also click **Browse** to display the Translation Output File dialog box to select a file name.
6	Click **Override default output path and filename** and then click the database option. (Continued on next page)

(Contd) Step	Action
7	Do you want to format the input file?
	◗ If YES, select **Format input document,** click the **Formatting Options** button, and select the options you want to use.
	Reference See <u>The Translation Options Dialog Box</u> topic for descriptions of the options.
	Comments This step is recommended for all X12 data.
	You must run the formatting process if you run the translation process.
	◗ If NO, continue with the next step.
8	Do you want to set translation options for the input file?
	◗ If YES, select **Translate document** and click the **Translation Options** button to display the Translation Options dialog box. Set the translation options you want to use; click **OK** to return to the Translate Documents dialog box.
	◗ If NO, continue with the next step.
9	Do you want to archive the EDI data?
	◗ If YES, continue with the next step
	◗ If NO, continue with the Step 11.
10	Select **Archive Data** and then click the **Archive Options** button to select the archiving options you want to use.
11	Is the input file in XML format?
	◗ If YES,GO TO Step 13.
	◗ If NO, continue with Step 12.
	Note The XML format is available only if you have the XML translation option.
12	Do you want to combine like interchanges for outbound Functional Acknowledgments created during an inbound translation?
	◗ If YES, select **Envelope standard output data**.
	◗ If NO, continue with the next step.
	(Continued on next page)

(Contd) Step	Action
13	Click **OK** to run the selected processes. **System Response** Gentran:Server translates the inbound data file, stores the resulting data in the appropriate file(s), and displays a process log.
14	Check the process logs for errors. **System Response** Gentran:Server places all the transaction sets that were processed without error into your output file. **Process logs** The process logs are in the rpt directory. ◗ The *boxin.err* log contains data with compliance errors. ◗ The *dtLog.err* file contains the error codes and their descriptions of errors pertaining to data transformation. ◗ The *xlcntl.err* log contains errors and related information pertaining to Trading Partnership records and files used during translation. ◗ The *edifrmat.not* file contains non-EDI characters that appeared before the Interchange header or after the Interchange trailer. ◗ The *edistat.irr* file contains archiving errors for inbound EDI data. ◗ The *edistat.orr* file contains archiving errors for outbound acknowledgments.

After translation   Once you are sure the translation object translates the data correctly, you can check in the map and translation object.

How to Run Outbound Translation

Introduction   This topic explains how to run an outbound translation from the **Translate** menu.

Notes
For outbound translation, you must be using a map with either an application or, if you have the XML translation option, an XML file as the source document.

Multiple sessions   If you have Gentran:Server for UNIX and are running multiple sessions of the client, you can run translation in only one client session for each host and environment combination. You can translate documents in more than one parallel session as long as each session is for a different host and environment combination.

If you attempt to translate documents when another session that is logged in to the same host and environment is already translating, Gentran:Server displays a message to let you know that the process is currently running and asks that you try again later.

If you have Gentran:Server Workstation and are running multiple sessions on your desktop, you can run translation in only one session.

Procedure   Use this procedure to run an outbound translation.

Step	Action
1	Click **Documents** on the Translate menu. **System Response** Gentran:Server displays the Translate Documents dialog box. **Note** The Translate Documents dialog box retains the parameter settings from the last translation performed.
2	Select the type of input file: ◗ Application file ◗ XML file (available if you have the XML translation option) ◗ Database (available if you have the ODBC translation option) (Continued on next page)

(Contd) Step	Action
3	Use one of the following methods to specify the input file name. ◗ Enter the full path and file name into the text box. ◗ Click **Browse** to display the Translation Input File dialog box. Browse through the directories and select a file name. Click **Open**. **System Response** The path and file name display in the input file name text box.
4	Use this table to decide your next step. <table><tr><td>**IF you want to…**</td><td>**THEN…**</td></tr><tr><td>Store the data in the location specified in the Trading Partnership record</td><td>Skip to step 5.</td></tr><tr><td>Store the data in a different location</td><td>Click **Override default output path and filename** and then click either the ASCII flat file or database option. If you select **ASCII flat file,** enter the full path and file name into the text box. **Notes** You can also click **Browse** to display the Translation Output File dialog box to select a file name. You must have the ODBC translation option to use the database option.</td></tr></table>
5	Do you want to set translation options for the input file? ◗ If YES, click the **Translate document option** and then click the **Translation Options** button to set the translation options you want to use. Continue with Step 6. ◗ If NO, GO TO Step 7. (Continued on next page)

(Contd) Step	Action
6	Use this table to determine how to indicate on the Translation Options dialog box the location of the Trading Partnership Code.

IF the Trading Partnership Code is…	**THEN…**
In the first 16 positions of each record in the input data file	The translator can find the Trading Partnership Code. Go to Step 7.
The same for all of the data in the input file	Click **Specify Trading Partnership Code** and then click the binoculars symbol (Browse button) to select the Trading Partnership code.
Marked in the first record of the file definition	Click **Trading Partnership Code defined in Application/File Definition** and then select the file that describes its position from the drop-down list. **Note** When you specify a file definition and translate with the A option, the file definition should either have a field in the first record named TPCODE to mark the Trading Partner Code field, or have application/Trading Partnership rules defined to determine the Trading Partner Code. The name is case-sensitive.
Not marked in the file definition	Click **Specify Trading Partnership Code Placement** and then complete the Position and Length boxes to point to the position within the input data.

(Contd) Step	Action
7	Do you want to archive the EDI data? ◗ If YES, select **Archive Data** and then click the **Archive Options** button to select the archiving options you want to use. ◗ If NO, continue with the next step. (Continued on next page)

(Contd) Step	Action
8	Do you want to combine like interchanges into a single interchange? ◗ If YES, select **Envelope standard output data**. ◗ If NO, continue with the next step.
9	Click **OK** to start the translation. **System Response** Gentran:Server translates the data file, stores the resulting data in the appropriate file(s), and displays a process log.
10	Check the process log for errors. **System Response** Gentran:Server places all the transaction sets that were processed without error into your output file. **Process logs** The process logs are in the rpt directory. ◗ The *boxout.err* log contains data with compliance errors. ◗ The *dtLog.err* file contains the error codes and the descriptions of errors pertaining to data transformation. ◗ The *xlcntl.err* log contains errors and related information pertaining to Trading Partnership records and files used during translation. ◗ The *edifrmat.not* file contains non-EDI characters that appeared before the Interchange header or after the Interchange trailer. ◗ The *edistat.irr* file contains archiving errors for inbound EDI data. ◗ The *edistat.orr* file contains archiving errors for outbound acknowledgments.

After translation   If you used the archive feature during translation, you can correct any errors in the archived file and send the data again. See Archiving Translation Data in this guide for instructions.

After testing a translation object
Once you are sure the translation object translates the input file correctly, you can check in the map and translation object.

How to Modify the Outbound Envelope Segments

Introduction   You can modify the values of the elements in the interchange and group envelope segments that Gentran:Server inserts in the envelope segments of a Trading Partnership's:

◗   Outbound EDI documents

◗   Outbound Functional Acknowledgments.

Standards compliance   The elements in the interchange and group envelope segments depend on the standard, the standard version, or (for EDIFACT) the syntax level. Gentran:Server bases the options it displays for the elements on the information you specify.

Example 1
If you specify X12/ANSI standard version 4020 and later, Gentran:Server displays "Repeating Element Separator" and a drop-down list of valid values for element ISA11 on the Interchange Information dialog box. For standard versions prior to 4020, Gentran:Server displays "Interchange Standards ID" in the ISA11 element field.
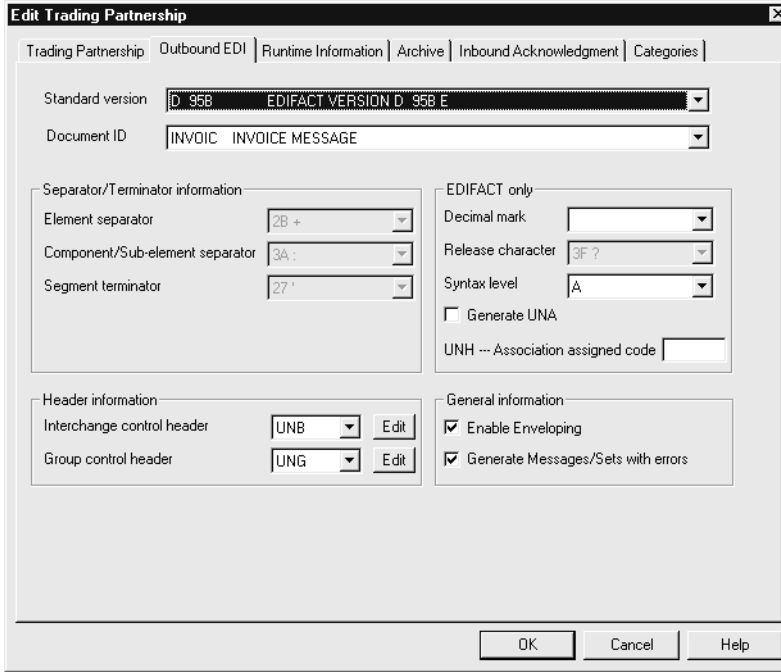
Example 2
If you specify UN/EDIFACT Syntax Version 4 for UNB element 0002 on the Interchange Header Information dialog box, Gentran:Server shows an 8-digit format (for example, 19991129) for element 0017, Date of Preparation, on the UNB and UNG dialog boxes.

Modifying the envelope segments for outbound EDI documents

Use this procedure to modify the envelope segments of outbound EDI documents for a Trading Partnership.

Step	Action
1	Open the Trading Partnership record that you want to modify. **Reference** See How to Search for a Trading Partnership Record or How to Edit Trading Partnership Records for instructions on opening Trading Partnership records.
2	Click the Outbound EDI tab. (Continued on next page)

(Contd) Step	Action
3	Click the **Edit** button of the **Interchange control header** or **Group control header** field. **System Response** The system displays the Information dialog box for the segment you selected from the drop-down box. The standard version and EDIFACT syntax level determine which elements are displayed on the dialog box. This illustration is for the ISA envelope segment for X12/ANSI standard version 4020.
4	Modify the element or elements you want to change. **Reference** See your Standards documentation for the element definitions of the interchange and group control header segments. (Continued on next page)

(Contd) Step	Action
5	Click **OK**. **Warning** You MUST click **OK** on this dialog box to save your changes. **System Response** The system displays the Outbound EDI tab.
6	Click **OK** to save your changes. **Warning** You MUST click **OK** on the Outbound EDI tab to save your changes to the envelope segment.

Modifying the envelope segments for outbound Functional Acknowledgments

Use this procedure to modify the envelope segments of outbound functional acknowledgments for a Trading Partnership record.

Step	Action
1	Open the Trading Partnership record that you want to modify. **Reference** See How to Search for a Trading Partnership Record or How to Edit Trading Partnership Records for instructions on opening Trading Partnership records.
2	Click the Outbound Acknowledgment tab.
3	Click the **Edit** button of the **Interchange control header** or **Group control header** field. **System Response** The system displays the Information dialog box for the segment
4	Modify the element or elements you want to change. **Reference** See your Standards documentation for the element definitions of the interchange and group control header segments. (Continued on next page)

| (Contd) Step	Action
5	Click **OK**. **System Response** The system displays the Outbound Acknowledgment tab.
6	Click **OK** to save your changes. **Warning** You MUST click **OK** on the Outbound Acknowledgment tab to save your changes to the envelope segment.

Generating a Translation Summary

Translation Summary Report

Introduction   If you archive your translation data, you can generate a Translation Summary of data about selected translations.

Translation Summary   This illustration shows an example Translation Summary.
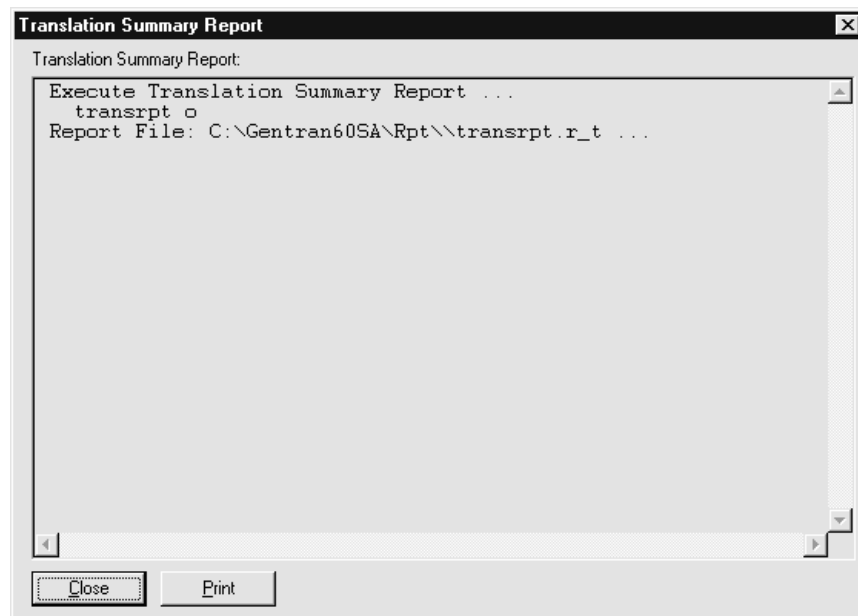
**Translation
Summary
contents**

The Translation Summary contains the following information for the selected interchanges, one functional group to a line:

◗ Trading Partnership Code

◗ Trading partner's Interchange ID for this interchange

◗ Interchange control number

◗ Group control number

◗ Set ID

◗ Translation date

◗ Number of sets translated

◗ Number of characters translated

The summary also shows grand totals for the number of sets and the number of characters.

Translation Summary Ranges Dialog Box

Illustration This illustration shows the Translation Summary Ranges dialog box.
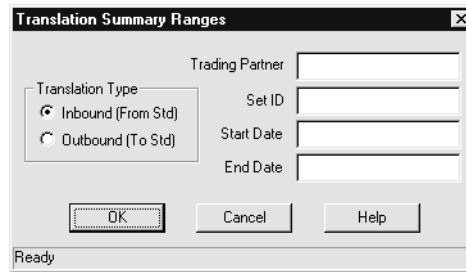
Fields and functions This table describes the fields of the Translation Summary Ranges dialog box and their functions.

Field	Function
Inbound (From Std)	Specifies that the summary will include only inbound translations.
Outbound (To Std)	Specifies that the summary will include only outbound translations.
Trading Partner	Specifies the Trading Partner Code for which to display translations. **Comment** If you enter a Trading Partner Code, do not specify a Set ID.
Set ID	Specifies the Translation Set ID for which to display translations. **Comment** If you leave this field blank, all sets are included. If you enter a Set ID, do not specify a Trading Partner Code. (Continued on next page)

(Contd) Field	Function
Start Date	Specifies the earliest date for which to display translations. **Comment** If you leave this field blank, all translations are included back to the earliest one.
End Date	Specifies the last date for which to display translations. **Comment** If you leave this field blank, all translations are included up to the most recent.

How to Generate a Translation Summary

Procedure   Use this procedure to generate a Translation Summary.

Step	Action
1	Click **Translation Summary** from the Translate menu. **System Response** The Translation Summary Ranges dialog box appears.
2	Click either **Inbound** or **Outbound**.
3	Use this table to specify the translations to include in the report. <table><tr><td>**IF you want to include…**</td><td>**THEN…**</td></tr><tr><td>Translations from only one particular Trading Partnership</td><td>Enter the Trading Partner Code.</td></tr><tr><td>Translations of only one particular document</td><td>Enter the Set ID.</td></tr><tr><td>Translations after a particular date</td><td>Enter the Starting Date.</td></tr><tr><td>Translations up to a particular date</td><td>Enter the Ending Date.</td></tr></table>
4	Click **OK**. **System Response** Gentran:Server displays the Translation Summary. If Gentran:Server finds no data to report, it creates an empty file and displays a report containing only a title. If Gentran:Server finds data, you can scroll from left to right and from top to bottom as necessary to see the entire report.
5	Do you want to print the Translation Summary? ◗ If YES, click **Print**. ◗ If NO, continue with next step.
6	Click **Close** to close the Translation Summary dialog box.

Archiving Translation Data

Contents

Overview

Introduction

About archiving

The Gentran:Server archiving feature for EDI translation enables you to store a copy of inbound and outbound EDI document status information that is created during translation, and the actual EDI documents. Through the archive setup windows, you choose the types of records to be archived.

Once data is archived, you can:

◗   Examine archived inbound and outbound data

◗   Edit the archived outbound data

◗   Extract archived outbound data and prepare it to resend it to the trading partner

◗   Reconcile functional acknowledgments

◗   Move data to media or to another machine.

In this chapter

This chapter shows you how to:

◗   Choose the default archiving options and run the archiving program

◗   Work with archived data, including reconciling functional acknowledgments

◗   Move archived data with the archive utilities.

Key terms

This table describes key terms used in this chapter.

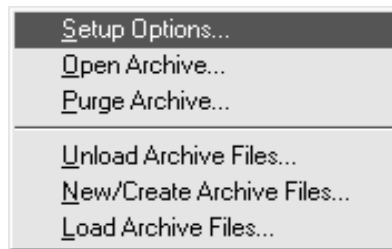| Term	Description
archive	The process of copying and saving EDI data that has been processed.
data file	The file that contains the inbound or outbound data.
EDI History Audit directory	The directory used to store permanent history files and EDI data.
EDI Status Audit directory	The directory used to store temporary translation audit files (*edistat.i* and *edistat.o*).

(Contd) Term	Description
ediarc	The Gentran:Server program that controls archiving.
functional acknowledgment (FA)	The standard transaction set used to acknowledge receipt of a transmission. The functional acknowledgment tells you if a document you sent was received and processed by your trading partner. It also can tell you if your document contained EDI standards compliance errors.
historical record	The index files named *edihist.dat* and *edihist.idx* that store the location of the archived data. The files are built from the temporary translation audit files. Together, they form the permanent audit file.
reconciliation	The process of viewing functional acknowledgments, viewing data with certain statuses, such as Error, Late, or Waiting, and correcting the EDI data.
status record	A temporary audit file named *edistat.i* or *edistat.o*, created during translation, which includes the directory path and file name for the EDI data received or generated. The record does not contain the actual EDI data.

The Archive Menu

Archive menu This illustration shows the **Archive** menu.

```
Setup Options...
Open Archive...
Purge Archive...

Unload Archive Files...
New/Create Archive Files...
Load Archive Files...
```

Archive menu options and functions

This table describes the options of the **Archive** menu and their functions.

Option	Function
Setup Options	Sets the archive and retrieval system defaults for EDI translation, which may be overridden by the Trading Partnership Record.
Open Archive	Accesses archived EDI data for viewing, editing, or extracting.
Purge Archive	Removes EDI historical records and stored data that is older than the retention period.
Unload Archive Files	Creates an ASCII file (*edihist.unl*) from the contents of an EDI historical file. You can then move the ASCII file to another machine or to media.
New/Create Archive Files	Creates two historical files named *eidhist.dat* and *edihist.idx*. These files are used to prepare for moving archived data from another machine or from media. You populate these files by archiving EDI data or by loading an ASCII file created from the contents of an Archive historical file.
Load Archive Files	Loads the EDI historical files (*edihist.dat/.idx*) from ASCII file data (*edihist.unl*).

The Archive Process

Introduction   The **ediarc** program controls archiving for EDI translation data. Archiving begins when the translation process produces status records. The subsequent stages in the process depend on how you set up archiving for your system.

Archiving process   This table describes the archive process when all archiving options are activated.

Stage	Description
1	During translation, Gentran:Server creates temporary audit files named *edistat.i* (inbound) and *edistat.o* (outbound) and places them in the EDI Status Audit directory. For each set translated, these audit files include pointers to the directory path and file name for the EDI data received or generated.
2	If you configure your archiving system to save historical records, **ediarc** posts the status records to the historical files, *edihist.dat* and *edihist.idx*, which reside in the EDI History Audit directory.
3	If you configure your archiving system to save EDI data files, **ediarc** checks the archive switches in the Trading Partnership record to determine which sets to archive and their purge dates. **ediarc** then copies the EDI data files into a directory hierarchy in the EDI History Audit directory. **Reference** See the topic <u>How archived files are named</u> in this chapter.
4	If functional acknowledgments were created during the translation, **ediarc** posts their status to the *edihist.idx* and *edihist.dat* files and copies the EDI data file to the EDI History Audit directory.
5	If the data was archived successfully, Gentran:Server deletes the temporary status records *edistat.i* and *edistat.o*. If an error occurred during archiving, the temporary files are kept for debugging under the names *edistat.irr* and *edistat.orr*.
6	Gentran:Server writes error messages to the file *xlcntl.err*, which is in the wrktemp subdirectory under the directory specified for User Files.

How archived files are named

The file name format for archived files is as follows:

> edihist\yyyy\mm\dd\hh\nnnnnnnn.ext

Where:

- ◗ "edihist" represents the name of the EDI History directory.
- ◗ "yyyy\mm\dd\hh" represents the year, month, day, and hour the data was translated.
- ◗ "nnnnnnnn" is the system-generated 8-character file name.
- ◗ ".ext" represents the file name extension.
    - — If you do not select a data compression utility during setup, the extension is *.arc*.
    - — If you select a data compression utility during setup, the extension is determined by the compression utility. (*.zip* for Pkzip files, *.z* for packed, and *.Z* for compressed.)

Example

Assume that the EDI History Audit directory is named *edihist*. The data file was translated on January 3, 1997, at 11 a.m. and compressed with Pkzip.

Ediarc copies the file and names it:

> *edihist\1997\01\03\11\00000001.zip*

Setting Up EDI Translation Archiving

Introduction

About this section

This section explains how to set up your archiving system. It contains these topics:

- ◗ Specifying the audit directories
- ◗ Choosing EDI archive options
- ◗ Selecting a data compression utility.

Before you begin

To ensure the accuracy of archiving and functional acknowledgment reconciliation, you must have unique control numbers across all documents for a trading partner. This is because the archiving program does not archive duplicate records; if two different records have the same control numbers, the archiving program rejects the second as a duplicate.

The best way to ensure unique control numbers is to use Organization records and maintain control numbers globally. This causes Gentran:Server to automatically assign unique, sequential control numbers to all documents for a single trading partner.

Reference

See How to Create a Group Organization Record in this guide for information about using Organization records.

WARNING

If you do not use Organization records and maintain control numbers globally, you must check each Trading Partnership record with the same Interchange IDs and edit the control numbers. Add a unique, sequential prefix (for example, 1, 2, 3, ..., *n*) to each control number so the control numbers are unique.

How to Specify the Translation Audit Directories

Introduction

How your organization specifies the EDI translation audit directories depends on whether you are running Gentran:Server for UNIX (client/server) or the Gentran:Server Workstation version.

Decision table for Creating Audit Directories

Use this decision table to determine the number and types of audit directories you need to create for the different versions of Gentran:Server.

IF you work in a...	THEN create...
Single-user environment in which one user runs translation and archives data	One audit directory that serves as both the EDI Status Audit directory and the EDI History Audit directory.
Multi-user environment in which several users create maps, but only one user runs translation and archives data	One audit directory that serves as both the EDI Status Audit directory and the EDI History Audit directory.
Multi-user environment in which several users run translation and archive data	One EDI History Audit directory that is the same for all users. A private EDI Status Audit directory for each user.

Audit directories for the UNIX version

If you are running Gentran:Server for UNIX, your system Security Administrator specified the EDI Status and EDI History Audit directories on the host when your security file was established. Check with your Security Administrator to make sure these directory paths are in place before you run **ediarc**.

The EDI Status Audit directory is your working directory. The EDI History Audit directory stores the path and file name to every archived file.

You can view the paths to these directories by selecting Setup Directories from the Settings menu, but you cannot change the paths. The Security Administrator can change the paths with the Security Administration software installed on the UNIX host.

Reference
See the *Gentran:Server for UNIX Maintenance and Troubleshooting Guide* or the *Gentran:Server Workstation Maintenance and Troubleshooting Guide* for more information.

Audit directories for the Workstation version

The Workstation version provides for two types of audit directories:

◗ EDI Status Audit stores your audit files, *edistat.o* and *edistat.i,* temporarily.

◗ EDI History Audit stores the full path and file name to every archived file.

If you are running the Workstation version of Gentran:Server, you must designate the paths to these directories before you run **ediarc**.

How to specify the audit directories in the Workstation version

Use this procedure to specify the audit directories in the Workstation version.

Step	Action
1	Click **Settings** on the menu bar.
2	Click **Setup Directories** on the **Settings** menu. **System Response** Gentran:Server displays the Setup Directories dialog box.
3	Type the full path name for the EDI Status Audit directory in the **EDI Status Audit** box; then type the full path name for the EDI History Audit directory in the **EDI History Audit** box.
4	Click **OK** to save your changes.

Archive/Retrieval Setup Dialog Box

Introduction   The Archive/Retrieval Setup dialog box is used to set the archiving options for EDI translation data and to select a data compression utility to compress your archived data. The dialog box has two tabs:

◗   EDI tab

◗   Compression tab

EDI tab   This illustration shows the EDI tab.
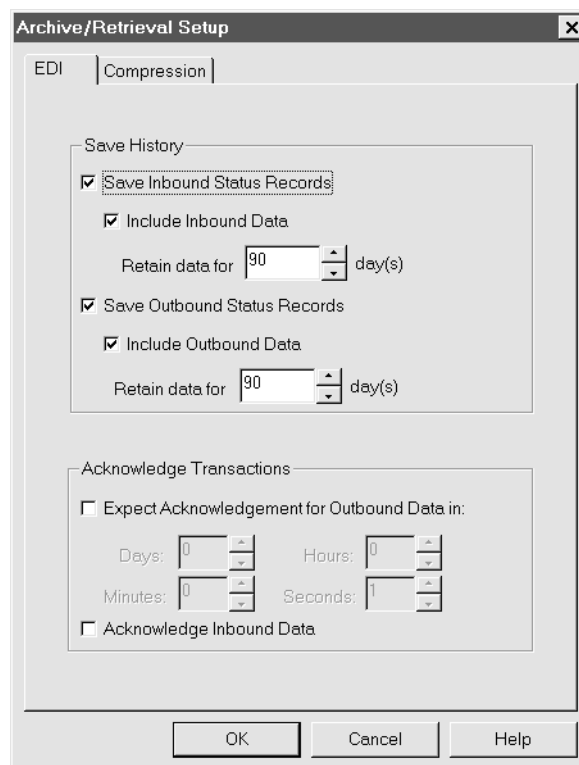
EDI tab fields and functions

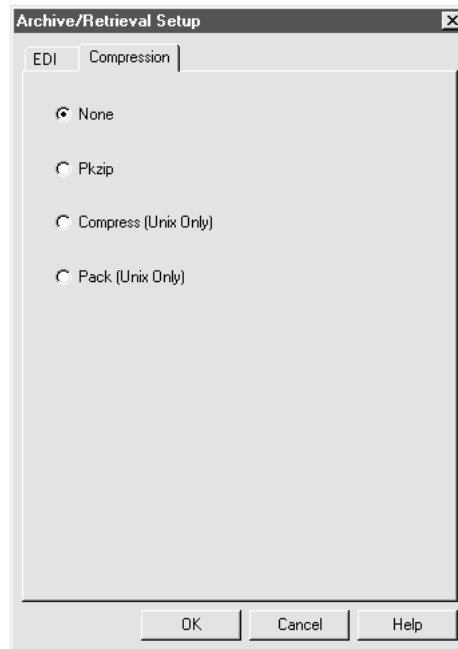This table describes the EDI tab fields and their functions.

Field	Function
Save History	
Save Inbound Status Records	Archives inbound EDI status records by converting them to historical records and adding them to the permanent audit history files, *edihist.dat* and *edihist.idx*.
Include Inbound Data	Archives inbound data files.
Retain data for ___ days	Sets the minimum number of days after which inbound data files are eligible for purging. After this number of days, the inbound data files and the associated historical records i the permanent audit file are eligible for purging.
Save Outbound Status Records	Archives outbound status records by converting them to historical records and adding them to the permanent audit files, *edihist.dat* and *edihist.idx*.
Include Outbound Data	Archives outbound data files.
Retain data for __days	Sets the minimum number of days outbound data files are kept. After this number of days, the outbound data files and associated historical records in the permanent audit file are eligible for purging.
Acknowledge Transactions	
Expect Acknowledgment for Outbound Data in	Signals system to expect acknowledgments for outbound data in the specified days, hours, minutes, and seconds.
Days	Sets the number of days in which to expect acknowledgment for outbound data.
Hours	Sets the number of hours in which to expect acknowledgment for outbound data.
Minutes	Sets the number of minutes in which to expect acknowledgment for outbound data.

(Contd) Field	Function
Seconds	Sets the number of seconds in which to expect acknowledgment for outbound data.
Acknowledge Inbound Data	Causes the system to send acknowledgments to trading partners when it receives inbound data.

Compression tab

This illustration shows the Compression tab on the Archive/Retrieval Setup dialog box.

Compression tab fields and functions

This table describes the fields of the Compression tab and their functions.

Field	Function
None	Indicates whether compression is on or off.
Pkzip (Workstation Only)	Specifies PKWARE's PKZIP as the compression utility. You must include PKZIP in your PATH statement. *(Continued on next page)*

(Contd) Field	Function
Compress (Unix Only)	Specifies Compress as the compression utility. Compress is available on every UNIX system.
Pack (Unix Only)	Specifies Pack as the compression utility. Pack is available only on some UNIX systems and compresses data tighter than Compress.

How to Set EDI Translation Archiving Options

Introduction

To set up EDI translation archiving and retrieval options to fit your business needs, you begin by selecting **Setup Options** from the **Archive** menu to display the Archive/Retrieval window. You use the EDI tab on this window to choose the records you want the system to archive.

Archiving options

On the EDI tab, you choose to have the system perform one or more of the following tasks:

◗ Create status records *(edistat.i* and *edistat.o*)

◗ Archive inbound status records, outbound status records, or both (create the *edihist.dat* and *edihist.idx* files from the *edistat.i* and *edistat.o* records)

◗ Archive inbound data files, outbound data files, or both

◗ Keep data files for a specified number of days

◗ Look for functional acknowledgments for outbound data in a specified number of days, hours, minutes, and seconds

◗ Acknowledge inbound data files.

Option dependencies

Archiving starts with the creation of status records, *edistat.o* and *edistat.i*, during translation. These status records are eventually posted to *edihist.dat* and *edihist.idx*, the historical records. The historical records store the location of the archived data. Therefore, to archive and retrieve data files, you must activate both the status record and historical record option.

Where default settings are stored

The Archive/Retrieval system default settings for EDI translation are stored in the configuration file *arcopt.cfg*. This file is in the edihist directory of each environment.

Relationship to Trading Partnership record	Gentran:Server uses the EDI Archiving settings as default settings when you set up a new Trading Partnership record, but enables you to override them for any Trading Partnership record. To determine which records to archive during processing, Gentran:Server checks the archive options of the Trading Partnership record. **Reference** See How to Create a Trading Partnership Record in this guide for information.

Selecting archiving options	Use this procedure to select the default archiving and retrieval options for your system.

Step	Action
1	Click **Setup Options** on the **Archive** menu. **System Response** Gentran:Server displays the EDI tab on the Archive/Retrieval Setup dialog box.
2	Click **inbound** and/or **outbound** in the **Save History** section to select the type of status records you want to save. You may select both types. **Example** To archive inbound status records, click **Save Inbound Status Records**. **System Response** The **Include Data** field for the type of status record you selected becomes available.
3	Do you want to archive the data as well as the status record? ◗ If YES, click **Include Inbound Data** or **Include Outbound Data** and type the number of days to keep the data in the Retain data for ___ day(s) field. ◗ If NO, continue with Step 5. (Continued on next page)

(Contd) Step	Action
4	Do you want your system to expect functional acknowledgments for outbound data? ◗ If YES, click **Expect Ack for Outbound Data in** and type the number of days, minutes, hours and seconds. ◗ If NO, continue with Step 6. **Comment** When you create a new Trading Partnership, Gentran:Server uses the choice you make in this step as the default setting.
5	Do you want to acknowledge inbound data? ◗ If YES, click **Acknowledge Inbound Data** and then continue with the next step. ◗ If NO, continue with Step 6. **Comment** When you create a new Trading Partnership, Gentran:Server uses the choice you make in this step as the default setting.
6	Click **OK** to save the settings.

How to Select a Data Compression Utility

Introduction

Archived data can consume large amounts of disk space. To help control disk space, you may use a compression utility to reduce the size of your archived documents.

Before you begin

Gentran:Server does not include compression utilities. If you want to compress your archived data, you must have installed one of the compression utilities that Gentran:Server supports:

◗ Pkzip

◗ Compress

◗ Pack

Important note
After the utility is installed, you must add the utility to your user path.

How compression works

When you select a compression utility, Gentran:Server records your selection in a configuration file. During archiving, Gentran:Server checks this file to determine which compression utility to use and then starts the compression utility.

File extensions of compressed data

These are the file extensions for compressed and non-compressed data:

Utility	File Extension
Pkzip	.zip
Pack	.z
Compress	.Z
Not compressed	.arc

Selecting a compression utility

Use this procedure to select a compression utility.

Step	Action
1	Click **Setup Options** from the **Archive** menu. **System Response** Gentran:Server displays the EDI tab of the Archive/Retrieval Setup dialog box.
2	Select the Compression tab.
3	Select the compression utility you have installed.
4	Click **OK** to save your selection.

Changing the compression utility

You may choose a different compression utility at any time or turn compression off. If you choose a different utility, Gentran:Server uses the new selection from the point at which you made the change; it does not change previously compressed data.

How to Run the Archive Program

Introduction

You can start **ediarc**, the program that controls EDI translation archiving, from:

◗ The command line

◗ A batch file or script you run from the Task Scheduler

◗ The **Archive Data** option on the Translate Outbound Documents and Translate Inbound Documents dialog boxes.

When to archive data

The best time to archive EDI translation data is immediately after all inbound and outbound translations, before the data file is purged or moved.

WARNING

To prevent loss of data, make sure that you run the archive program before you delete or move a data file.

Starting ediarc from a batch file or script

You may run **ediarc** from a UNIX or DOS batch file and use the Task Scheduler to run **ediarc** on a schedule.

Reference

See the chapter Using the Task Scheduler in this guide for information about using the Task Scheduler.

Starting ediarc from the command line

Use this procedure to start **ediarc** from the command line.

Step	Action
1	Go to the UNIX command line.
2	Type the **ediarc** command, a space, parameter to indicate inbound or outbound, a space, the -cp parameter, and the full path to the configuration file, *envprim.cfg*. Then press ENTER. **Command statement** ediarc -i/o -cp[configuration path] **Example: UNIX version** ediarc -i -cp[configuration path] **Example: Workstation version** ediarc -o -cp[configuration path] **System Response** If you configured your archiving system to save EDI data files, **ediarc**: ▮ Checks the archive switches in the Trading Partnership record to determine which sets to archive and their locations ▮ Copies the EDI data files into a directory hierarchy in the EDI History Audit directory ▮ Posts the *edistat.i* and/or *edistat.o* record to the permanent audit history files, *edihist.dat* and *edihist.idx* ▮ Deletes the *edistat.i* and/or *edistat.o* record.

Reference

See Command Reference in the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for more information about the **ediarc** command.

Working with Archived Data

Introduction

In this section	This section explains how to retrieve and work with archived EDI translation data. It includes these topics:

- ◗ How to search for archived information
- ◗ How to view information about archived data
- ◗ How to view and reconcile functional acknowledgment errors
- ◗ How to extract archived documents (for resending to a trading partner or for editing)
- ◗ How to purge data.

Relationship of topics	You must conduct a search for archived information before you can view archived data, reconcile functional acknowledgments, or extract archived documents so that you can resend them or edit them.

How to Search for an Archived Document

Introduction

Before you can view or retrieve archived EDI translation data, you must first search for the records and files by entering search criteria on the EDI Document Search dialog box.

EDI Document Search dialog box

This illustration shows the EDI Document Search dialog box.

(Continued on next page)

EDI Document Search fields and functions

This table describes the fields on the EDI Document Search dialog box and their functions.

Field	Function
Trading Partner Code	Specifies the Trading Partner Code to use in the search. To search for a Trading Partner Code, click the binoculars icon to display the Trading Partner Search dialog box.
Doc. Reference No.	Specifies the document reference number to search for, if the data came from Gentran:Server product level PCM or higher. Leave this field blank if you are searching for documents that came from Gentran:Server Workstation.
Message/Set ID	Specifies the message or set ID to use in the search (such as 104 Air Shipment Information or BANSTA Banking Status Message). Leave blank to view all sets.
Starting (CCYYMMDD)	Specifies the first translation date to use in the search. Leave blank to see a range of archived data that begins with the earliest translation date in the file.
Ending (CCYYMMDD)	Specifies the last translation date to use in the search. This field defaults to the current date.
Document status	Specifies the functional acknowledgment status for documents with a functional acknowledgment.
Interchange IDs	
Yours	Specifies your company's interchange ID.
Partner's	Specifies your trading partner's interchange ID.
Data Control Numbers	
Interchange Control No.	Specifies the interchange control number to use in the search.
Group Control No.	Specifies the group control number to use in the search.
Transaction Set Control No.	Specifies the transaction set control number to use in the search. (Continued on next page)

(Contd) Field	Function
Restrict Search To	
Inbound EDI Documents	Selects Inbound EDI Document as the type of documents to search.
Outbound EDI Documents	Selects Outbound EDI Document as the type of documents to search.
Display Results With	
Summary Information Only	Selects collapsed mode as the display mode. **Reference** For information about display modes, see the How to View Set-Level Information topic.
Full Document Details	Selects expanded mode as the display mode. **Reference** For information about display modes, see the How to View Set-Level Information topic.
Search	Starts the search.
Cancel	Cancels the search request.

Searching for EDI Documents

Use this procedure to search for an EDI document or documents.

Step	Action
1	Click **Open Archive** on the **Archive** menu. **System Response** The system displays the EDI Document Search dialog box.
2	Type the search criteria for the document or documents. **Reference** For field descriptions, see the EDI Document Search fields and functions topic in this chapter. (Continued on next page)

(Contd) Step	Action
3	Click **Search**. **System Response** Gentran:Server finds records in the *edihist.dat* and *edihist.idx* files that match the search criteria and displays the results on the **Document Search Results** window. If you select Summary Information only, the window displays the first record for each interchange/group combination that meets the search criteria. If you select Full Document details, the window displays a list of all records that meet the search criteria.
4	Use this table to determine your next step. <table><tr><td>**IF you want to…**</td><td>**THEN go to…**</td></tr><tr><td>Sort the search results by TP Code, Interchange Date, or another column</td><td>How to Sort Document Search Results in this chapter.</td></tr><tr><td>View record information</td><td>How to View Set-Level Information in this chapter.</td></tr><tr><td>View functional acknowledgment errors or change a document's status</td><td>How to View Functional Acknowledgment Errors in this chapter.</td></tr><tr><td>Extract data to: ◗ View it ◗ Edit it and prepare it to resend to your trading partner</td><td>How to Extract Archived EDI Documents in this chapter.</td></tr></table>

How to Sort Document Search Results

Introduction

When you search for an archived document, the results displayed on the Document Search Results dialog box are sorted by the trading partner's interchange ID. This topic explains how to change the sorting order.

View modes

You can view Gentran:Server document search results in two modes:

◗ Summary Information Only, which displays only the first records for each interchange/group combination that meet the search criteria

◗ Full Document Details (expanded), which displays all the records that meet the search criteria.

**Document
Search Results
dialog box**

This illustration shows the Document Search Results dialog box in full document detail (expanded) mode.
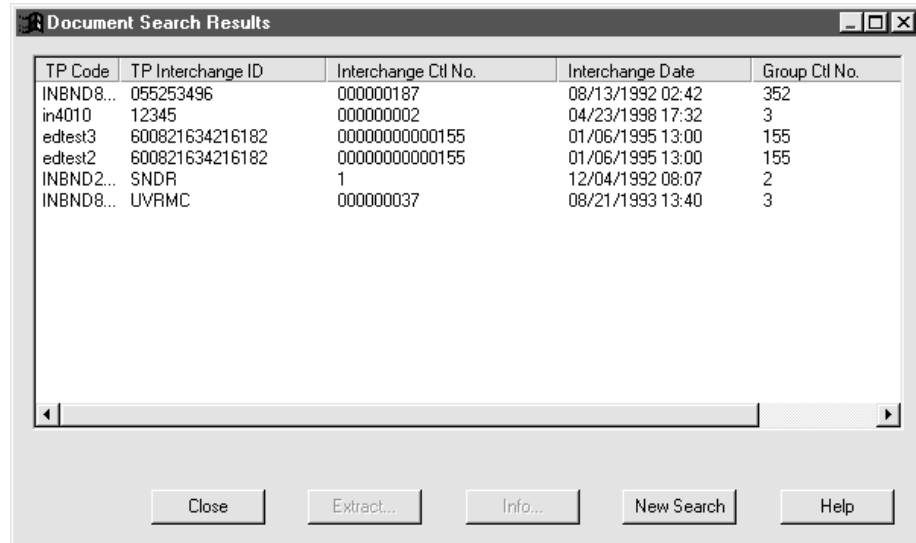
Document Search Results fields and functions

This table describes the fields on the Document Search Results dialog box and their functions.

Field	Function
TP Code	Displays the Trading Partner Code associated with the record.
TP Interchange ID	Displays the trading partner's interchange ID.
Interchange Ctl No.	Displays the interchange control number associated with the record.
Interchange Date	Displays the date and time of the interchange.
Group Ctl No.	Displays the Group Control Number associated with the record.
No. of Sets	Displays the number of transaction sets in the record.
Close	Closes the Document Search Results dialog box.
Extract	Starts the data extraction process. **Reference** See How to Extract Archived EDI Documents for instructions.
Info...	Displays information about the selected record. If the selection is a transaction set, the system displays the history window for the record. If the selection is a group, the system displays the Group Information window. **Reference** See How to View Set-Level Information for instructions.
New Search	Displays the EDI Document Search dialog box, which enables you to start a new search.
Help	Displays help text for the Document Search Results dialog box.

(Continued on next page)

Sorting search results

Use this procedure to change the sorting order of the records displayed on the Document Search Results dialog box.

Step	Action
1	Conduct a document search by following the instructions in the How to Search for an Archived Document topic. **System Response** Gentran:Server displays the **Document Search Results** window in either summary mode or expanded mode.
2	Click on the column header that you want to use to sort the records. **System Response** The system sorts the records in ascending order of the column you selected.
3	Do you want to sort the records in reverse order? ◗ If YES, click the same column header again. ◗ If NO, continue with Step 4. **Note** Each subsequent click on a header reverses the sorting order, toggling between ascending and descending order.
4	*Use this table to determine your next step.* <table><tr><td>**IF you want to…**</td><td>**THEN…**</td></tr><tr><td>Extract archived data</td><td>See How to Extract Archived EDI Documents in this chapter.</td></tr><tr><td>View archived data</td><td>See How to View Set-Level Information in this chapter.</td></tr><tr><td>Start a new search</td><td>Click **New Search**.</td></tr><tr><td>Close the Document Search Results dialog box</td><td>Click **Close**.</td></tr></table>

How to View Set-Level Information

Introduction   You can view this information about archived data at the set-level:
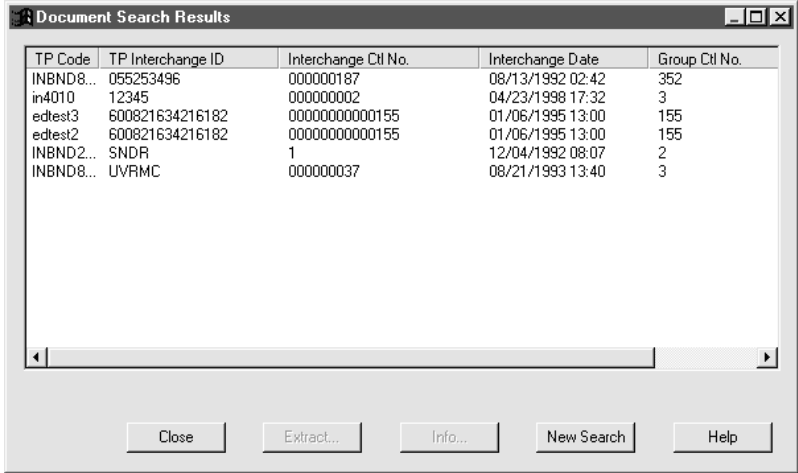
◗   Identification

◗   Translation

◗   Processing

◗   Archive

◗   Acknowledgment

Reference
To view information at the group level, see How to View Group Information in this chapter.

Viewing set-level information   Use this procedure to view archived EDI translation data at the set level.

Step	Action
1	Conduct a document search in Full Document Details (expanded) mode by following the instructions in the How to Search for an Archived Document topic. **System Response** Gentran:Server displays the Document Search Results window. **Example** (Continued on next page)

(Contd) (Contd)	**Action**
2	Move the focus to the record you want to view and then click **Info**. **System Response** Gentran:Server displays the history window for the selected record.
3	Click the tab for the type of information you want to see.

Tab	*Description*
Identification	The basic information used to identify the document, such as the Trading Partner Code, control numbers, document reference number, and the interchange and group IDs.
Acknowledgment Information	Information about a functional acknowledgment transaction, such as the date and time the functional acknowledgment was received, the document status, message type, and error data.
Translation information	Information about the transaction, such as the date and time, status, result, segment terminator, and number of segments and characters.
Processing Information	The date after which the document is eligible for purging.
Archive Information	The file name of the archived file, the archive directory path to the file, and (if you are viewing set information) a number that tells you where the set starts in the data file.

Identification tab   This illustration shows the Identification tab.

Identification tab fields and functions table   This table describes the Identification tab fields and their functions.

Field	Function
Trading Partner	Specifies the Trading Partner code.
Interchange Ctl	Specifies the document's interchange control number.
Group Ctl	Specifies the document's group control number.
Trans. Set Ctl	Specifies the document's transaction set control number.
Document Ref	Specifies the document reference number (only if from Gentran:Server product level PCM and higher).
Name	Specifies the trading partner's name.
Set/document ID	Specifies the set or document identifier.
Interchange created	Specifies the date and time the interchange was translated.
Group created	Specifies the date and time the group was translated.

Acknowledgment Information tab

This illustration shows the Acknowledgment Information tab.

Acknowledgment Information tab fields and functions table

This table describes the Acknowledgment Information tab fields and their functions.

Field	Function
Ack. expected	Indicates whether a functional acknowledgment is expected. Yes indicates one is expected. No indicates that a functional acknowledgment is not expected.
Ack. date/time	Specifies the date and time at which the functional acknowledgment was expected.
Document Status	Specifies the document status. Valid responses include Accepted with No Errors, Partially Accepted with Errors, Rejected, and Waiting.
Message type	Specifies the transaction type: 997, 999, or CONTRL.

(Contd) Field	Function
Change document status	Indicates whether the Document Status field is updated when the functional acknowledgment is verbal. ◗ Verbal OK - A verbal acknowledgment arrived; no errors in document ◗ Verbal Error - A verbal acknowledgment arrived; document contains errors
View Errors	Indicates whether the system extracts a functional acknowledgment record from the archive and displays the related error information in the Errors section of the window
First	Prompts the system to display information about the first error in the archive
Next	Prompts the system to display information about the next error in the archive
Previous	Prompts the system to display information about the previous error in the archive
Last	Prompts the system to display information about the last error in the archive
Errors	Prompts the system to display information about an error. The type of information displayed depends on the type of error. **Example** If the error is at the segment level, the Errors section displays the Segment ID in addition to the Error Number and Error Text.

Translation Information tab

This illustration shows the Translation Information tab.

Translation Information tab fields and functions table

This table describes the Translation Information tab fields and their functions.

Field	Function
Date/time	Specifies the translation date and time.
Translation Result	Indicates the status of translation, such as "OK" or an error type number.
Seg. Terminator	Specifies the character used to separate segments.
No. Segments	Specifies the number of segments in the set.
No. Characters	Specifies the number of characters in the set.

Processing Information tab

This illustration shows the Processing Information tab.

Processing Information tab fields and functions table

This table describes the Processing Information tab fields and their functions.

Field	Function
Purge Date	Specifies the date on which the document is eligible for purging.
User Defined Info	Currently unused. Reserved for future use.

Archive Information tab

This illustration shows the Archive Information tab.

Archive Information tab fields and functions table

This table describes the Archive Information tab fields and their functions.

Field	Function
Archive File Name	Specifies the archived file.
Archive Directory	Specifies the archive directory.
Archive Offset	If you are viewing information for a set, this field Specifies the number that indicates where the set starts in the data file. The EDI Extract program uses this information to extract the archived data.

How to View Group Information

Introduction   The Group Information dialog box displays basic information used to identify a document. You can access this dialog box from the Document Search Results dialog box.

Group Information dialog box   This illustration shows the Group Information dialog box.

Group Information dialog box fields and functions   This table describes the Group Information fields and their functions.

Field	Function
Trading Partner	Displays the Trading Partner code.
Interchange Ctl	Displays the document's interchange control number.
Group Ctl	Displays the document's group control number.
Interchange created	Displays the date and time the interchange was translated.
Group created	Displays the date and time the group was translated.
Interchange IDs	Displays your interchange ID and your trading partner's interchange ID.

(Contd) Field	Function
Group IDs	Displays your group ID and your trading partner's group ID.
No. of Transaction Sets	Displays the number of transaction sets in this Group document.
Change doc. status	Updates the Document Status field for all transaction sets in this group when the functional acknowledgment is expected. ◗ Verbal OK - A verbal acknowledgment arrived. No errors in document. ◗ Verbal Error - A verbal acknowledgment arrived. Document contains errors. **Note** This field is disabled for inbound documents.

Viewing group information

Use this procedure to view group information.

Step	Action
1	Conduct a document search (in Summary mode) by following the instructions in the How to Search for an Archived Document topic. **System Response** Gentran:Server displays the Document Search Results window.
2	Move the focus to the group-level record you want to view and then click **Info**. **System Response** Gentran:Server displays the Group Information window.
3	Do you want to set the status for all transaction sets in this group to Verbal Ok or Verbal Error? ◗ If YES, continue with Step 4. ◗ If NO, go to Step 6. (Continued on next page)

(Contd) Step	Action
4	Select the new status from the **Change document status** choices and then click **OK** to save your changes. **Note** If you try to change document status on a group for which an acknowledgment was not expected, the system ignores the status change. **System Response** Gentran:Server displays a confirmation prompt.
5	Click **Yes** to confirm your changes.
6	Click **Cancel** to close the Group Information dialog box. **System Response** Gentran:Server returns to the Document Search Results dialog box.

Reconciling Functional Acknowledgments (FAs)

Overview

Definitions

A **functional acknowledgment** (FA) that you receive is an inbound document that contains the status of outbound data you sent to a trading partner.

A **functional acknowledgment** that you send is an outbound document in response to a document you received from a trading partner.

Purpose

The purpose of functional acknowledgments is to provide you with:

◗ The status of a sent or received document (rejected, accepted, or accepted with errors).

◗ The information necessary to reconcile the historical records maintained in *edihist.dat*.

◗ Error information related to a sent or received document.

Types of FAs

This table lists the types of functional acknowledgments and describes each type.

◗ The receiving company uses one of these functional acknowledgment transaction sets to communicate the results of a transmission back to the sending company.

◗ For each functional group sent, the acknowledgment details what transactions were received and if there were any segment or data element syntax errors according to the related standard.

Type	Description
997	The X12 functional acknowledgment transaction set.
999	The UCS functional acknowledgment transaction set.
CONTRL	The Edifact and Odette functional acknowledgment transaction set.

(Contd) Type	Description
Network	For TRADACOMS-formatted messages only, Gentran:Server can receive and reconcile both IBM® and INS® network acknowledgments.

Functional acknowledgment levels

Functional acknowledgments are always associated with a transaction group (a collection of transactions).

The functional acknowledgments that Gentran:Server handles have three different levels of detail:

◗ **Group**, which indicates if a group of like documents contained errors
◗ **Set**, which indicates if an individual transaction set contained errors
◗ **Segment/Element**, which indicates the exact segment and element containing an error.

Control numbers in a functional acknowledgment

The archiving program does not archive duplicate records, so if two different records have the same control numbers, the archiving program rejects the second as a duplicate. See the Setting Up EDI Translation Archiving topic.

Reconciliation

If you send or receive functional acknowledgments, you need to reconcile errors regularly.

The process of reconciling functional acknowledgments is similar to the process of reconciling your bank statement: you review the status of documents that you have sent and received, and then resolve any discrepancies.

You must archive EDI data (including any functional acknowledgments) after each translation to use the functional acknowledgment reconciliation feature.

Setting Up Your System to Handle Functional Acknowledgments

Introduction

You must request functional acknowledgments from your trading partner before sending EDI data. If you receive documents from a trading partner, you should ask if they want functional acknowledgments from you and at what level they want to receive them.

Configuring your system

To configure your system to send acknowledgments to all trading partners when you receive inbound data, select the **Acknowledge Inbound Data** option on the EDI tab of the Archive/Retrieval window.

To configure your system to expect acknowledgments from all trading partners for outbound data, complete the **Expect Ack for Outbound Data in** option on the EDI tab of the Archive/Retrieval window.

References
See the EDI tab for a description of the acknowledgment options.

Configuring a Trading Partnership record

You can configure the functional acknowledgment settings at the Trading Partnership level. These settings override your system settings for the individual trading partner.

To configure your system to send acknowledgments to a specific trading partner when you receive inbound data from that trading partner, select the **Create acknowledgment for inbound document** option from the Outbound Acknowledgment tab of the Edit Trading Partnership dialog box.

To configure your system to expect acknowledgments from a specific trading partner for outbound data, complete the **Expect acknowledgment for outbound document in** option on the Inbound Acknowledgment tab of the Edit Trading Partnership dialog box.

Reference
See Outbound Acknowledgment tab for a description of the inbound document acknowledgment options on this tab of the Edit Trading Partnership dialog box.

See Inbound Acknowledgent tab for a description of the outbound document acknowledgment options on this tab of the Edit Trading Partnership dialog box.

Reconciliation Process

Stage table   This table describes the reconciliation process.

Stage	Description
1	When you receive a functional acknowledgment from a trading partner, the reconciliation program collects these items: ◗ Sender's Interchange ID ◗ Receiver's Interchange ID ◗ Sender's Group/Application ID ◗ Receiver's Group/Application ID ◗ Group control number ◗ Set/message control number ◗ Set/Message ID.
2	Gentran:Server searches the files *edihist.dat* and *edihist.idx* for a record that matches the acknowledgment.

	IF a match is…	**THEN Gentran:Server…**
	Found	Reconciles the functional acknowledgment and assigns the appropriate status to the document. Valid responses include Accepted with No Errors, Partially Accepted with Errors, or Rejected.
	Not found	Reports a reconciliation error in the xlcntl.err file. **Note** The inability to find a match may be because the original document was not archived, the original document was purged, or a problem exists with the functional acknowledgment. (Continued on next page)

(Contd) Stage	Description
3	Click **View Errors** on the Acknowledgment Information tab of the document history window. **System Response** Gentran:Server retrieves the first functional acknowledgment error associated with the selected document. The system then displays information about the error in the Errors section of the tab. The tab has scroll buttons that enable you to view other errors associated with the document.

How to View Functional Acknowledgment Errors

Introduction   You view functional acknowledgment information, including error information, on the Acknowledgment Information tab of the document history window.

Acknowledgment Information tab   This illustration shows the Acknowledgment Information tab.

Procedure   Use this procedure to view functional acknowledgment errors for a document.

Step	Action
1	Display the document history window by following the procedure in the Viewing set-level information topic.
2	Click the Acknowledgment Information tab. (Continued on next page)

(Contd) Step	Action
3	Click **View Errors**. **Note** This button is disabled if the received functional acknowledgment was at the group or set level. **System Response** Gentran:Server displays information about the first functional acknowledgment error associated with the document in the **Errors** box.
4	To scroll through information about other errors for this document, click **Next**, **Previous**, and/or **Last**.

How to Change the Document Status

Introduction   If you receive a verbal acknowledgment, you can change the status of an outbound document to "Verbal OK" or "Verbal Errors" from the:

◗ Document Search Results window

◗ Acknowledgment Information tab of the document history window

◗ Group Information window.

You can change the status of:

◗ All transaction sets in a group from either the Document Search Results window or the Group Information window

◗ Individual transaction sets from either the Document Search Results window or the Acknowledgment Information tab.

Using the Document Search Results window   Use this procedure to enter a verbal acknowledgment from the Document Search Results window. You can change the status of all transaction sets in a group or the status of individual transaction sets.

Step	Action
1	Conduct a search for outbound documents by following the instructions in the How to Search for an Archived Document topic. **System Response** Gentran:Server displays the Document Search Results window.
2	Move the focus to the record for which you want to enter a verbal acknowledge status and right-click. **System Response** Gentran:Server displays a popup window. **Note** At the group level, the **Set Verbal Error** and **Set Verbal OK** items are always enabled in the popup window. At the set level, if the current status is either Set Verbal Error or Set Verbal OK, the current status is disabled in the popup menu. If the current status is Not Expected, both items are enabled. *(Continued on next page)*

(Contd) Step	Action
3	Select the new status from the menu. ◗ **Verbal OK** - A verbal acknowledgment arrived; no errors in document ◗ **Verbal Error** - A verbal acknowledgment arrived; document contains errors **CAUTION** **If the record is a group-level entry, the system will apply the status you select to all transaction sets in this group.** **System Response** Gentran:Server displays the Change status confirmation dialog box.
4	Click **Yes** to confirm the change in status.

Using the Acknowledgment Information tab

Use this procedure to change the status of an individual transaction set in an outbound document.

Step	Action
1	Display the document history window for an outbound transaction set by following the procedure in the <u>How to View Set-Level Information</u> topic.
2	Click the Acknowledgment Information tab.
3	Select the new document status by clicking on one of the following options in the Change Doc. Status section of the tab: ◗ **Verbal OK** - A verbal acknowledgment arrived; no errors in document ◗ **Verbal Error** - A verbal acknowledgment arrived; document contains errors **System Response** Gentran:Server displays a confirmation dialog box.
4	Click **Yes** to save your changes. **System Response** Gentran:Server applies the status you selected and displays the Document Search Results dialog box.

Using the Group Information window	Use this procedure to enter a verbal acknowledgment for all transaction sets in a group from the Group Information window.

Step	Action
1	Conduct a search for outbound documents by following the instructions in the <u>How to Search for an Archived Document</u> topic. **System Response** Gentran:Server displays the Document Search Results window.
2	Move the focus to the group-level record you want to view and click **Info**. **System Response** Gentran:Server displays the Group Information window.
3	Select the new status from the Change document status choices. ◗ **Verbal OK** - A verbal acknowledgment arrived; no errors in document ◗ **Verbal Error** - A verbal acknowledgment arrived; document contains errors
4	Click **OK** to save your changes. **System Response** Gentran:Server displays the Change status confirmation dialog box.
5	Click **Yes** to confirm the changes. **System Response** Gentran:Server applies the status you selected to all transaction sets in this group and displays the Document Search Results dialog box.

EDI errors	When you reconcile outbound documents, you may want to correct errors in the EDI data and resend the document to your trading partner.

Reference
See the <u>Editing an archived EDI document</u> in this chapter.

Miscellaneous Procedures

How to Extract Archived EDI Documents

Introduction   There are three reasons to extract archived EDI translation data files:

◗   View the data

◗   Edit erroneous EDI data and prepare it to send it to your trading partner

◗   Prepare EDI data that your trading partner did not receive so that you can resend it.

The Extract EDI Archive tab enables you to extract archived documents.

Extract EDI Archive dialog box   This illustration shows the Extract EDI Archive dialog box.
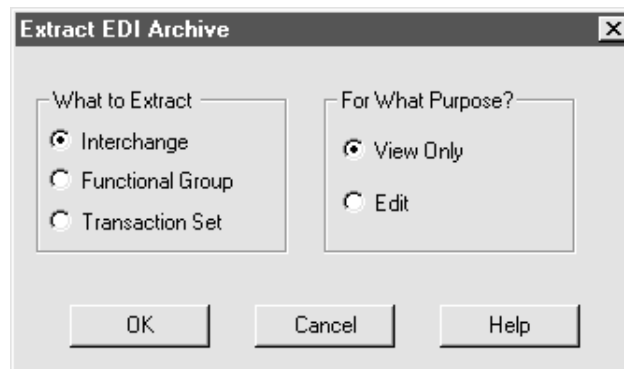
Extract EDI Archive fields and functions table   This table describes the **Extract EDI Archive** fields and their functions.
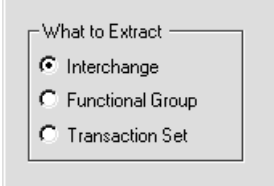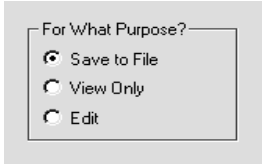
Field	Function
What to extract	Specifies the type of data to extract.
Interchange	Prompts the system to select interchange data.
Functional Group	Prompts the system to select functional group data.
Transaction set	Prompts the system to select transaction set data.
	(Continued on next page)

(Contd) Field	Function
For what purpose	Indicates the reason for extracting the data.
View only	Prompts the system to extract data only for viewing.
Edit	Specifies that the extracted data will be edited.

Extracting archived data

Use this procedure to extract archived data.

Step	Action
1	Conduct a document search for the document you want to extract. **Reference** See How to Search for an Archived Document in this chapter.
2	Place the focus on the record you want to extract and then click **Extract**. **System Response** Gentran:Server displays the Extract EDI Archive dialog box.
3	In the What to Extract area, click the type of data to extract: **Interchange**, **Functional Group**, or **Transaction Set**. What to Extract ⊙ Interchange ○ Functional Group ○ Transaction Set
4	In the **For What Purpose** section, click the reason for extracting the data. For What Purpose? ⊙ Save to File ○ View Only ○ Edit (Continued on next page)

(Contd) Step	Action	
5	Click **OK**. **System Response**	
	IF you selected…	**THEN…**
	View only	The system displays the archived document in the File Browser so that you can view it.
	Edit	The system displays the archived document in the File Browser so that you can edit it. To continue, see <u>How to Prepare Documents to Resend</u> in this chapter.

How to Prepare Documents to Resend

Introduction

The resend program repackages archived outbound or inbound data so that you can send it. The program replaces new lines with character terminators and increments the control numbers to those you specify.

Resending Outbound Data

You can edit archived outbound data that contains errors and then send the corrected data to your trading partner. In this case, you want to increment the control numbers so that the archiving program does not reject the document as a duplicate.

You may also resend archived outbound data that your trading partner never received. In this case, you may use the same control numbers as the original document.

Resending Inbound Data

You may also resend inbound data. This data is saved as a newline-terminated file. The **Override Control Numbers** option is unavailable when you are resending inbound data.

When resending inbound data, you must use the File Browser to edit the control numbers before saving the file. This prevents the duplication of audit records in the archive when you retranslate the file inbound.

Resend EDI Document dialog box

This illustration shows the Resend EDI Document dialog box.

Resend EDI Document dialog box fields and functions

This table describes the fields of the Resend EDI Document dialog box and their functions.
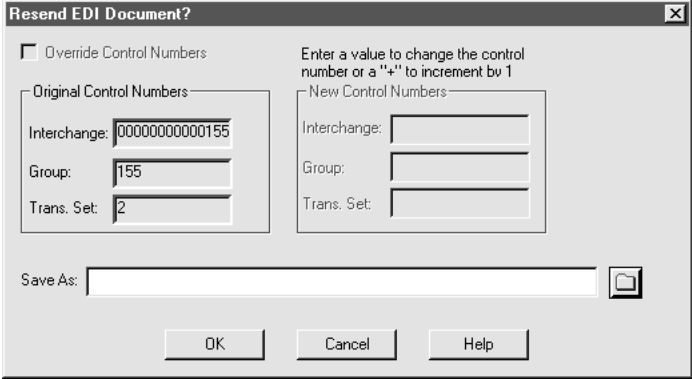
Field	Function
Override Control Numbers	Prompts the system to activate the New Control Numbers fields.
Original Control Numbers	
Interchange	Specifies the current interchange control number of the archived data.
Group	Specifies the current group control number of the archived data.
Trans. Set	Specifies the current transaction set control number of the archived data.
New Control Numbers **Note** You must increment the original control numbers to archive new, corrected data.	
Interchange	Defines the new interchange control number.
Group	Defines the new group control number.
Trans. Set	Defines the new transaction set control number.
Save as	Defines a file name for a copy of the archived data.

Editing an archived EDI document

Use this procedure to correct EDI data that contains errors so that you can send the corrected version to your trading partner.

Step	Action
1	Extract the document for editing. **Reference** See Extracting archived data in this chapter. **System Response** Gentran:Server displays the document in the File Browser.
2	Use these instructions to edit the document. <table><tr><td>**IF you want to…**</td><td>**THEN…**</td></tr><tr><td>Insert characters</td><td>Click at the insertion point and type the characters.</td></tr><tr><td>Copy selected text to the clipboard</td><td>Click **Copy** on the **Edit** menu.</td></tr><tr><td>Paste text on the clipboard into the document</td><td>Click at the insertion point and then click **Paste** on the **Edit** menu.</td></tr><tr><td>Select the entire document</td><td>Click **Select All** on the **Edit** menu.</td></tr><tr><td>Search for characters, words, or phrases</td><td>Click **Find** on the **Edit** menu. Then, enter the search string.</td></tr><tr><td>Search for the next occurrence of a search item</td><td>Click **Find Next** on the **Edit** menu.</td></tr><tr><td>Replace selected text</td><td>Click **Replace** on the **Edit** menu. Then type the replacement text.</td></tr><tr><td>Print the document</td><td>Click **Print** on the **File** Menu.</td></tr><tr><td>Save the changes to the document</td><td>Click **Save** on the **File** menu.</td></tr><tr><td>Save a copy of the document under a new path or name</td><td>Click **Save As** on the **File** menu. and enter the new path and file name.</td></tr></table>
3	Click **Save** on the **File** menu to save your changes.

(Contd) Step	Action
4	Click **Close** on the **File** menu. **System Response** Gentran:Server displays the Resend EDI Document dialog box.
5	Do you want to assign new control numbers to the extracted outbound data? ◗ If YES, click **Override Control Numbers**. Then, type the new control numbers in the **New Control Numbers** boxes. To reuse an original number, leave a box blank. ◗ If NO, continue with Step 6. **WARNING** **The Override Control Numbers option is unavailable for inbound data.** **Note** If you do not assign new control numbers to outbound data and it has already been sent, the recipient's archive program views the edited document as a duplicate and does not archive it.
6	Type a destination path and file name for the data. **Note** If you type only a filename, the program uses *$ENV_ROOT* as the default directory. (Continued on next page)
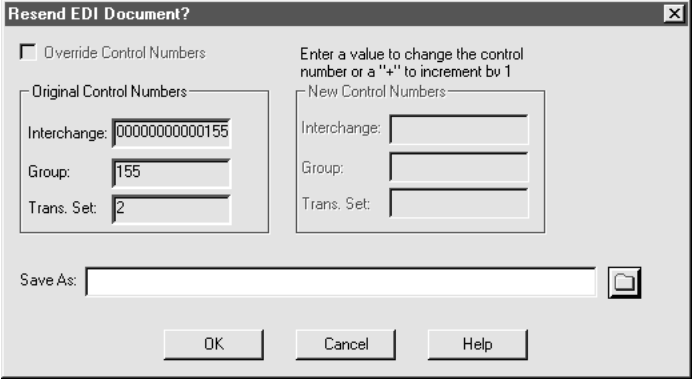
(Contd) Step	Action
7	Click **OK**. **System Response** If you resent outbound EDI data, the resend program modifies the archived outbound data, replacing new lines with character terminators and replacing the original control numbers with the new ones you specified. If you resent inbound data, the system creates a new file using the name typed in step 6.
8	If you want to archive this data, run **ediarc**. **Reference** For more information, see the topic How to Run the Archive Program earlier in this chapter.
9	Resend the document by following the procedures your company has established for sending EDI documents.

Preparing an unedited archived EDI document for resending

Use this procedure to prepare to resend an archived document that your trading partner did not receive.

Step	Action
1	Extract the document for editing. **Reference** See <u>Extracting archived data</u> in this chapter. **System Response** Gentran:Server displays the document in the File Browser window.
2	Click **Close** on the **File** menu. **System Response** Gentran:Server displays the Resend EDI Document dialog box. ![Resend EDI Document? dialog box showing Override Control Numbers checkbox, Original Control Numbers section with Interchange: 00000000000155, Group: 155, Trans. Set: 2, New Control Numbers section with empty Interchange, Group, and Trans. Set fields, a Save As field, and OK, Cancel, Help buttons]
3	Do you want to assign new control numbers to the extracted data? ◗ If YES, click **Override Control Numbers**. Then, type the new control numbers in the **New Control Numbers** boxes. To reuse an original number, leave a box blank. ◗ If NO, continue with Step 5. **Note** If you do not assign new control numbers, the archive program views the unedited document as a duplicate and does not archive it. This is fine if you have not made changes. (Continued on next page)

(Contd) Step	Action
4	Do you want to save the data to a different path or file name? ◗ If YES, type the new path and file name in the **Save As** box. ◗ If NO, continue with Step 6. **Note** If you do not type a new path and file name, the program uses *edixtrct.tmp* as the default file name.
5	Click **OK**. **System Response** The resend program modifies the archived data, replacing new lines with character terminators and replacing the original control numbers with the new ones you specified.
6	Resend the document by following the procedures your company has established for sending EDI documents.

How to Purge Data

Introduction
Leaving large amounts of data in the archive files slows down the retrieval process when you view or edit data. You should remove the stored data from the audit/archive directory regularly. Determine the interval based on management of your disk space and your company's policies.

Running the purge program
The purge program enables you to purge data based on criteria you select. You may start the purge program from:

◗ The archive menu

◗ The command line

◗ A batch file or script through the Task Scheduler.

In this section, you learn to run the purge program from the **Archive** menu.

Comment
The data purge process may require a long time to run. You may want to run it from a batch file or script during off hours through the Task Scheduler.

When you schedule **edipurge** to run during the off hours, you must verify that the system will not be running **ediarc** at the same time.

References
To select data for purging with a command line program, use the **edipurge** command. See the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for more information about **edipurge**.

To run the purge program from the Task Scheduler, see the chapter in this guide.

The Archive Purge dialog box

This illustration shows the Archive Purge dialog box for archived translation data.

Archive Purge dialog box fields and functions

This table describes the fields of the Archive Purge dialog box and their functions.

Field	Function
Trading Partner Code	Determines the Trading Partner Code associated with the data you want to purge. Leave blank to purge data from all trading partners.
Interchange Control Number	Specifies the Interchange Control Number of the data you want to purge. Leave blank to select all interchanges.
Group Control Number	Specifies the Group Control Number of the data you want to purge. Leave blank to select all groups.
Set ID	Specifies the Set ID of the data you want to purge. Leave blank to select all transaction sets.
FA Status	Specifies the functional acknowledgment status of the data you want to purge. Leave blank to purge all sets.

(Contd) Field	Function
Search Date Range	
Starting (CCYYMMDD)	Specifies the earliest translation date of the data to be purged. Leave blank to purge a range of data beginning with the earliest data available.
Ending (CCYYMMDD)	Specifies the latest translation date of the data you want to purge.
Restrict Search To	
Inbound EDI Document	Restricts purging to only inbound EDI documents.
Outbound EDI Document	Restricts purging to only outbound EDI documents.
After Purging	
Reduce EDI History File Size	Compresses the EDI History file after documents are purged. **Note** The EDI History file is not available to the **ediarc** program during the compression process.

Purging data   Use this procedure to purge archived data.

Step	Action
1	Verify that the **ediarc** program is not running.
2	Click **Purge Archive** on the **Archive** menu. **System Response** Gentran:Server displays the Archive Purge dialog box.
3	Type the criteria of the data you want to purge. Leave a field blank to select all the documents in that criteria category. **Example** Leave the Set ID text box blank to purge all sets. (Continued on next page)

(Contd) Step	Action
4	Click **OK**. **System Response** Gentran:Server purges the records from the EDI History Audit file, and also purges the archived data files if no other records point to the data file. When the process is complete, Gentran:Server displays a list of the records and data files that were purged.
5	Use this table to determine your next step. | IF you want to…	THEN…
Print the list of purged records and data files	Click **Print**.
Exit the Purge Archive Files window	Click **OK**.	

Using the Task Scheduler

Contents

Overview

Task Scheduler Basics

Introduction
You can use Gentran:Server's Task Scheduler to run tasks (DOS batch files or UNIX scripts) automatically on a schedule you specify.

You may find it convenient to run translations and other Gentran:Server functions from a batch file or script. You can schedule batch files or scripts to perform functions unattended during off-hours.

The Task Scheduler is available in both the Gentran:Server for UNIX (client/server) version and the Workstation version. However, each version handles a scheduled file differently.

Task Scheduler example
You might create a batch file or script to run outbound translation, and then use the Task Scheduler to run it every night at 11:00 p.m.

Gentran:Server for UNIX version
If you are running Gentran:Server for UNIX, the Task Scheduler sends the scheduled files to the host and executes the UNIX **crontab** command. The UNIX **crontab** command adds the file to the **cron** spool area. The UNIX **cron** program activates the timer.

The location of crontab and the UNIX program **cron** varies among the types and version of the UNIX operating system.

Example
Here is a sample path:

/var/spool/cron/crontab/muser

Note
The *muser* file, which represents the server where **cron** runs, needs permission to run **cron**.

Workstation version

The Workstation version of Gentran:Server has its own timer, **mtimer**. You may activate this executable file by:

◗ Clicking **Run Task Scheduler** on the **Tools** menu

◗ Configuring *mtimer.exe* to start when your computer starts.

When you use the Task Scheduler to schedule a file, the Task Scheduler saves the file to *crontab.dat*. Then, when you activate **mtimer**, **mtimer** reads the *crontab.dat* file and processes the files based on the schedule you established in the Task Scheduler.

In this chapter

This chapter contains relevant points about creating batch files or scripts containing the Gentran:Server command-line and operating-system commands you want to perform. You will also learn how to use the Task Scheduler to specify the date and time you want the batch file or script to run and important points about running the batch files and scripts.

Key terms

This table lists the key terms used in this chapter.

Term	Description
batch file	A file that contains command-line commands and DOS commands.
cron	A UNIX operating-system utility that runs commands and scripts according to a preset schedule. This is the executable file that activates the timer in Gentran:Server for UNIX.
ediarc	The Gentran:Server program that controls archiving.
edifrmat	The utility that removes segment terminator and replaces it with a new-line terminator.
envelope	The utility that combines like interchanges.
mtimer	A feature you can use to start, stop, or resume unattended processes in the Workstation version of Gentran:Server.
script file	A file that contains command-line commands and UNIX commands.

The Flow of Work

This table describes the process of scheduling a batch file or script.

Stage	Description
1	Create a batch file or script file that contains the Gentran:Server commands and operating-system commands that you want to run on a schedule.
2	Use the Task Scheduler to schedule execution of the batch file or script by specifying the path and name of the batch file or script and the date and time you want it to run.
3	Use this table to determine the next action.

<table>
<tr><td>Stage</td><td colspan="2">Description</td></tr>
<tr>
<td rowspan="3"></td>
<td>IF operating environment is...</td>
<td>THEN...</td>
</tr>
<tr>
<td>Gentran:Server for UNIX</td>
<td>Gentran:Server automatically uses cron to activate the process unless prohibited from doing so by your system administrator.</td>
</tr>
<tr>
<td>Gentran:Server Workstation</td>
<td>You must use the Run Task Scheduler command to start the timer (mtimer).</td>
</tr>
</table>

DOS Batch Files and UNIX Script Files

Introduction

For use with Gentran:Server, DOS batch files or UNIX scripts generally contain a combination of operating-system commands and Gentran:Server command-line commands. When you create a batch file or script, you enter the commands in the order you want to execute them.

Reference
See the chapter Command Reference in the *Gentran:Server for UNIX and Workstation Technical Reference Guide* for information on command syntax and options.

Used for unattended translation

Perhaps one of the most common uses for batch files or scripts within Gentran:Server is for unattended inbound and outbound translation.

About unattended inbound translations

To run inbound translations unattended, you create a batch file or script that includes commands to do some or all of the following:

◗ Delete files created in the previous translation (mandatory unless you want Gentran:Server to append these results to the results of the previous translation).

◗ Remove non-EDI data from the input file (mandatory; this is the **edifrmat** utility).

◗ Translate the EDI data (mandatory).

◗ Envelope the outbound functional acknowledgment or CONTRL (optional).

◗ Archive the EDI data exactly as it was received (optional).

◗ Archive the outbound functional acknowledgments (optional).

◗ Create and print a Translation Summary Report (optional).

◗ Delete or rename intermediate files created in this run (mandatory).

About unattended outbound translations

To run outbound translations unattended, you create a batch file or script that includes commands to do some or all of the following:

◗ Delete files created in the previous translation (mandatory unless you want Gentran:Server to append these results to the results of the previous translation).

◗ Translate the data into EDI or XML data (mandatory).

Note
If you have the XML translation option, then the data can also be translated into XML.

◗ Combine like interchanges for EDI data (optional; this is the envelope utility).

◗ Archive the EDI data exactly as it was sent (optional).

◗ Create and print a Translation Summary Report (optional).

◗ Delete or rename intermediate files created in this run (mandatory).

Sample Gentran:Server Batch Files (Workstation Version)

Introduction	If you are running the Workstation version of Gentran:Server, Gentran:Server batch files run from the Task Scheduler and **mtimer**.

How Gentran:Server batch files differ from DOS files	Gentran:Server batch files differ from DOS files in the following ways:

◗ Gentran:Server batch files can execute only Gentran:Server and DOS command-line commands, working from the top of the file to the end.

◗ Gentran:Server batch files cannot contain conditions or pointers (for example, **GOTO**). However, you can nest DOS batch files in Gentran:Server batch files. Nested DOS batch files execute properly, including conditions and pointers. You cannot, however, include Windows Gentran:Server commands (**Iftran** and **mtimer**) in DOS batch files that are nested in Gentran:Server batch files.

◗ To adhere to the syntax of the **mtimer** scripting language, for the Workstation version you must precede each DOS command in unnested or top-level batch files with 'DOS'.

About these samples	The following batch files are intended as examples only. File names used here suggest possible naming conventions. In your work, select one file naming convention and use it consistently.

Keep in mind that you can create one batch file to perform multiple tasks or handle multiple standard types.

Sample 1: Batch file for inbound purchase orders	The sample batch file below deletes files previously created, runs inbound translation on X12 purchase orders, archives the received EDI data and the functional acknowledgment sent, and then deletes the **edifrmat** input file.

Batch file line	Function
DOS del \server\temp\xlcntl.err	Deletes the translation log file *xlcntl.err.*
DOS del val4po.dat	Deletes previous **Iftran** translation output. (Continued on next page)

Batch file line	Function
edifrmat \server\data\val4.in boxin.edi -I	Runs the **edifrmat** program. This program prepares the data for translation.
lftran boxin.edi -if val4po.dat	Runs the **lftran** program. This program translates the data. **Comment** **lftran** uses the six key fields in the inbound document to locate the Trading Partnership record.
envelope	Runs the **envelope** program. This program combines like interchanges.
ediarc -I	Archives the inbound data using the **ediarc** program.
ediarc -O	Archives the outbound data using the **ediarc** program.
DOS del \server\data\val4.in	Deletes the temporary file(s) that the **edifrmat** program creates.

Sample 2: Batch file for inbound purchase orders

This sample deletes previously created files, runs inbound translations on EDIFACT purchase orders, archives the received EDI data and the CONTRL sent, and then deletes the **edifrmat** input file.

Batch file line	Function
DOS del \server\temp\xlcntl.err	Deletes the translation log file *xlcntl.err*.
DOS del boxin.po	Deletes previous **lftran** translation output.
edifrmat \server\data\val4.in boxin.edi -I	Runs the **edifrmat** program. This program prepares the data for translation. (Continued on next page)

(Contd) Batch file line	Function
lftran boxin.edi -i	Runs the **lftran** program. This program translates the data. **Comment** **lftran** uses the 6 key fields in the inbound document to locate the Trading Partnership record.
envelope	Runs the **envelope** program. This program combines like interchanges.
ediarc -I	Archives the inbound data using the **ediarc** program.
ediarc -O	Archives the outbound data using the **ediarc** program.
DOS del \server\data\val4.in	Deletes the temporary file that the **edifrmat** program creates.

Sample 3: Batch file for outbound invoices

The sample batch file below deletes files from previous executions, runs outbound translations on X12 invoices, and archives the EDI data sent.

Batch file line	Function
DOS del \server\temp\xlcntl.err	Deletes the translation log file *xlcntl.err*.
DOS del boxin.edi	Deletes previous **lftran** translation output.
lftran \server\data\boxin.udf -oT[TPCODE]	Runs the **lftran** program. This program translates data.
envelope	Runs the **envelope** program. This program combines like interchanges.
ediarc -O	Archives the outbound data using the **ediarc** program.

Sample 4: Batch file for outbound invoices

This sample deletes files from previous executions, runs outbound translations on EDIFACT invoices, and archives the EDI data sent.

Batch file line	Function
DOS del \server\temp\xlcntl.err	Deletes the translation log file *xlcntl.err*.
lftran \server\data\boxin.udf -oT[TPCODE]f edat	Runs the **lftran** program. This program translates the data.
envelope	Runs the **envelope** program This program combines the interchanges.
ediarc -O	Archives the outbound data using the **ediarc** program.

Sample UNIX Scripts

Introduction
If you are running Gentran:Server for UNIX, you use UNIX scripts for inbound and outbound translation. This topic contains samples of UNIX scripts you can schedule using the Task Scheduler.

Note
As in UNIX, commands in scripts are case-sensitive.

About these samples
The following scripts are intended as examples only. File names used here suggest possible naming conventions. In your work, you should select one file naming convention and use it consistently.

Keep in mind that you can create one script to perform multiple tasks or handle multiple standard types.

Sample 1: UNIX script for inbound purchase orders
The sample UNIX script below deletes previously created files, runs inbound translation on X12 purchase orders, archives the received EDI data and the functional acknowledgment sent, and deletes the **edifrmat** input file.

Script line	Description
rm /usr/server/temp/xlcntl.err	Removes the translation log file *xlcntl.err*.
rm value4po.dat	Removes previous **lftran** translation output.
edifrmat /usr/server/data/ value_4.in boxin.edi -l	Runs the **edifrmat** program. This program prepares the data for translation.
lftran boxin.edi -if value4po.dat -cp	Runs the **lftran** program. This program translates the data. **Comment** **lftran** uses the 6 key fields in the inbound document to locate the Trading Partnership record.
envelope	Runs the **envelope** program. This program combines like interchanges. (Continued on next page)

(Contd) Script line	Description
ediarc -I	Archives the inbound data using the **ediarc** program.
ediarc -cp /user/server/temp	Archives the outbound data using the **ediarc** program.
rm /usr/server/data/val4.in	Removes the temporary file that the **edifrmat** program creates.

Sample 2: UNIX script for inbound purchase orders

This sample deletes previously created files, runs inbound translations on EDIFACT purchase orders, archives the received EDI data and the CONTRL sent, and deletes the **edifrmat** input file.

Script line	Description
rm /usr/server/temp/xlcntl.err	Removes the translation log file *xlcntl.err*.
rm boxin.po	Removes previous **lftran** translation output.
edifrmat /usr/server/data/value_4.in boxin.edi -l	Runs the **edifrmat** program. this program prepares the data for translation.
lftran boxin.edi -i boxin.in -cp /user/server/temp	Runs the **lftran** program. This program translates the data. **Comment** **lftran** uses the 6 key fields in the inbound document to locate the Trading Partnership record.
envelope	Runs the **envelope** program This program combines like interchanges.
ediarc -I -cp /usr/server/temp	Archives the inbound data using the **ediarc** program.
ediarc -O -cp /usr/server/temp	Archives the outbound data using the **ediarc** program.
rm /usr/server/data/val4.in	Removes the temporary file that the **edifrmat** program creates.

Sample 3: UNIX script for outbound invoices

The sample UNIX script below deletes files from previous executions, runs outbound translation on X12 invoices, and archives the EDI data sent.

Script line	Description
rm /usr/server/temp/xlcntl.err	Removes the translation log file *xlcntl.err.*
rm boxin.edi	Removes previous **lftran** translation output.
lftran /usr/server/data/boxin.udf -oT[TPCODE]f boxin.in -cp /user/server/temp	Runs the **lftran** program. This program translates the data.
envelope	Runs the **envelope** program. This program combines like interchanges.
ediarc -O -cp /usr/server/temp	Archives the outbound data using the **ediarc** program.

Sample 4: UNIX script for outbound invoices

This sample UNIX script deletes files from previous executions, runs outbound translations on EDIFACT invoices, and archives the EDI data sent.

Script line	Description
rm /usr/server/temp/xlcntl.err	Removes the translation log file *xlcntl.err.*
rm dat.edi	Removes previous **lftran** translation output.
lftran /usr/server/data/boxin.udf -oT[TPCODE]f dat.edi -cp /user/server/temp	Runs the **lftran** program. This program translates the data.
envelope	Runs the **envelope** program. This program combines like interchanges.
ediarc -O -cp /usr/server/temp	Archives the outbound data using the **ediarc** program.

Task Scheduler Dialog Box

Introduction

You use the Task Scheduler to specify the batch file or script you want to run and the date and time at which you want it to execute.

Opening the Task Scheduler

To open the Task Scheduler, click **Schedule Tasks** on the **Tools** menu.

The Task Scheduler dialog box

This illustration shows the Task Scheduler dialog box.
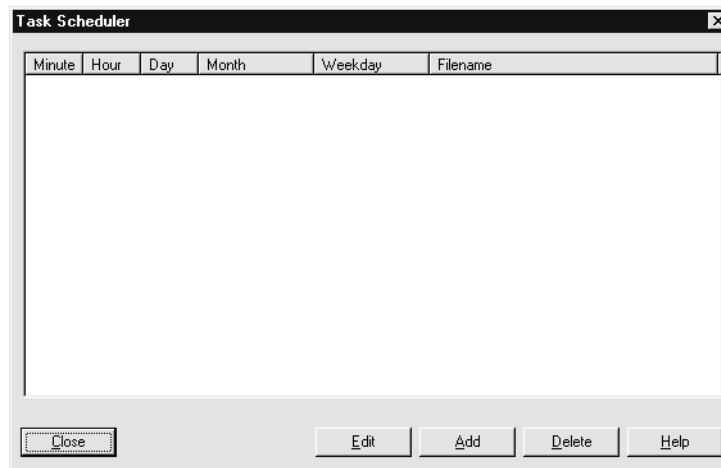
Task Scheduler dialog box fields, functions, and values

This table describes the Task Scheduler dialog box text fields that you use to enter information about the batch file or script and the execution date and time.

Field	Function	Values
Minute	Specifies the minute	0 through 59
Hour	Specifies the hour (24-hour clock)	0 through 23
Day	Specifies the day of the month	1 through 31
Month	Specifies the month (Continued on next page)	JAN through DEC

(Contd) Field	Function	Values
Weekday	Specifies the day of the week	SUN through SAT
Filename	Specifies the path and batch file or script name	
Edit	Opens the Edit Task Scheduler Entry dialog box so that you can change the date or time value or the filename.	
Add	Opens the Add Task Scheduler Entry dialog box so that you can add a task to the scheduler	
Delete	Removes a selected entry from the Task Scheduler	

Note

Use the asterisk (*) to include all values in a field. For example, if you want to run a batch file every day of the week, enter an asterisk into the Weekday box. To run the task every month of the year, enter an asterisk into the Month box.

Examples of scheduling a task

This table shows schedules for example batch files A, B, and C.

Batch File	Minute	Hour	Day	Month	Weekday
A	15	*	*	*	*
B	0	20	*	*	MON
C	30	8	30	*	*

◗   A runs everyday, at 15 minutes past every hour.

◗   B runs every Monday at 8 p.m.

◗   C runs on the thirtieth (30th) of each month at 8:30 a.m.

About scheduling tasks in the Gentran:Server for UNIX version

When you close the Task Scheduler after scheduling tasks, Gentran:Server sends the file to the host and executes the UNIX crontab command. This adds the file to the cron spool area. This UNIX operating system **cron** program activates the timer.

CAUTION

Closing the Task Scheduler overwrites any existing *crontab* file and deletes tasks previously scheduled outside of Gentran:Server. You should check for previously scheduled tasks and, if necessary, schedule Gentran:Server tasks outside of Gentran:Server.

Reference
See the topic How to Run Scripts (UNIX Version) in this chapter for instructions on how to use **crontab** to check scheduled tasks and how to schedule Gentran:Server tasks outside of Gentran:Server.

Add Task Scheduler Entry Dialog Box

Introduction

The Add Task Scheduler dialog box is used to add a batch file or script to the scheduler.

Illustration

This illustration shows the Add Task Scheduler dialog box.
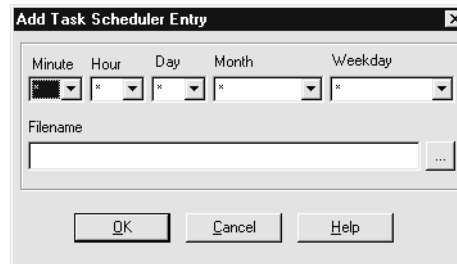
Fields and functions

This table lists the fields of the Add Task Scheduler dialog box dialog box and their functions.

Field	Function	Values
Minute	Specifies the minute	0 through 59
Hour	Specifies the hour (24-hour clock)	0 through 23
Day	Specifies the day of the month	1 through 31
Month	Specifies the month	JAN through DEC
Weekday	Specifies the day of the week	SUN through SAT
Filename	Specifies the path and batch file or script name	alphanumeric characters

Procedures
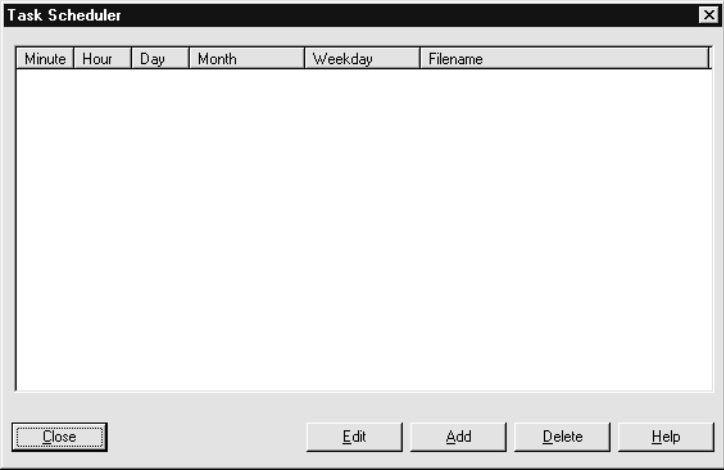
How to Add a Batch File or Script to the Scheduler
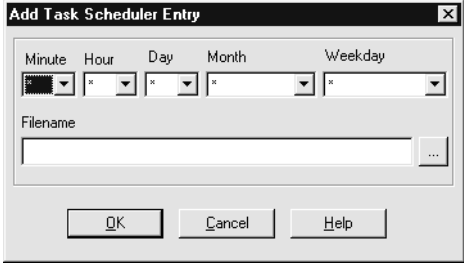
Introduction   This topic describes how to use the Task Scheduler to indicate the day and time you want to execute a batch file or script.

Scheduling a batch file or script   Use this procedure to schedule a batch file or script.
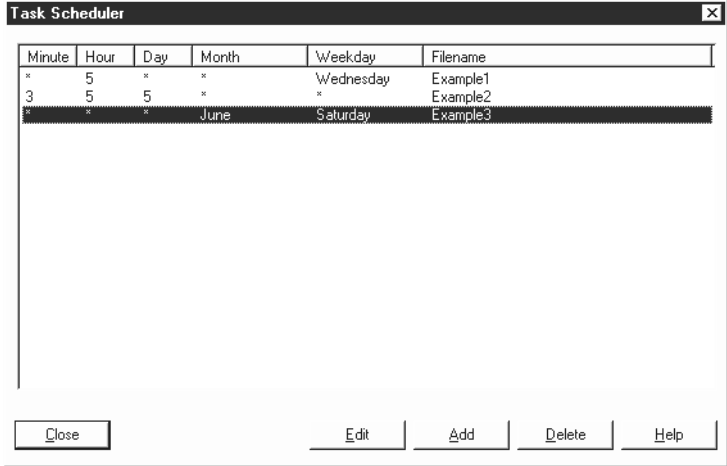
Step	Action
1	Click **Schedule Tasks** on the **Tools** menu. **System Response** The system displays the Task Scheduler dialog box. **Reference** See the topic <u>Task Scheduler Dialog Box</u> in this chapter for instructions and a description of features. (Continued on next page)
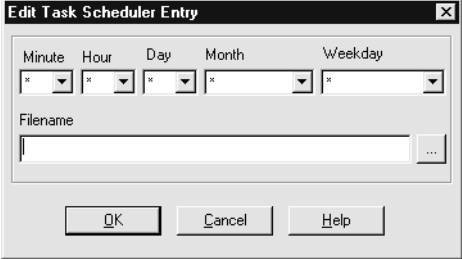
(Contd) Step	Action
2	Click the **Add** button. **System Response** The system displays the Add Task Scheduler Entry dialog box.
3	From the drop down list, select the appropriate time and date value.
4	Click the **Filename** text box and enter the full path and file name of the batch file or script. **Note** Alternatively, you can click **Browse** to display the Select File dialog box that enables you to search for the file.
5	Click **OK** to save your changes.

How to Edit a Scheduled Entry

Introduction   You can use the Task Scheduler to change the date or time values or the file name entered.

Changing an entry   Use this procedure to change a schedule entry.
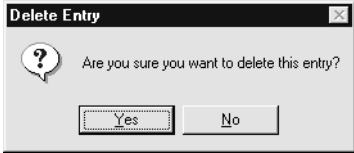
Step	Action
1	Open the Task Scheduler. **System Response** The system displays the Task Scheduler dialog box. **Reference** See the topic Task Scheduler Dialog Box in this chapter for instructions and a description of features.
2	Select the entry you want to edit. (Continued on next page)

(Contd) Step	Action
3	Click the **Edit** button. **System Response** The system displays the Edit Task Scheduler Entry dialog box. **Note** See the topic <u>Add Task Scheduler Entry Dialog Box</u> for a description of the fields for the Edit Task Scheduler Entry dialog box.
4	Select a value from the drop down menu in the field you want to edit.
5	Type the full path and filename of the batch file or script. **Note** You can also click the **Browse** button to display the Select File dialog box that enables you to search for the file.
6	Click **OK** to save your changes.

How to Delete an Entry from the Scheduler

Introduction This topic explains how to delete a scheduled task from the Task Scheduler.

Deleting a task Use this procedure to delete a scheduled task.

Step	Action
1	Open the Task Scheduler. **Reference** See the topic Task Scheduler Dialog Box in this chapter for instructions and a description of features.
2	Click anywhere on the row that contains the task you want to delete.
3	Click **Delete**. **System Response** The system displays the Delete Entry dialog box. ![Delete Entry dialog box — "Are you sure you want to delete this entry?" with Yes and No buttons]
4	Do you want to delete the entry? ◗ If YES, click Yes. **System Response** The system removes the task from the Task Scheduler. ◗ If NO, click No. **System Response** The systems returns to the Task Scheduler dialog box.

How to Run Scripts (UNIX Version)

Introduction

Gentran:Server for UNIX automatically uses **cron** to run a scheduled script at the specified time.

Note
Your system administrator can prohibit a scheduled script from running.

Using crontab to check scheduled tasks

To check the tasks you currently have scheduled outside of Gentran:Server, enter the following command at a UNIX prompt from any directory:

    crontab -l

Scheduling Gentran:Server tasks outside of Gentran:Server

Use this procedure to schedule Gentran:Server tasks outside of Gentran:Server.

Step	Action
1	Copy the current list of your tasks by entering the following UNIX command from any directory: **crontab -l> jobs.txt** Where *jobs.txt* is a temporary file for storing tasks.
2	Use any text editor to edit the file, adding your Gentran:Server tasks to the current task list.
3	Save the file.
4	To overwrite the current task list, enter the following command: **crontab jobs.txt**

MTimer and Workstation Batch Files

Introduction   This topic discusses the use of **mtimer** in processing Gentran:Server Workstation batch files with the Run Task Scheduler tool.

Reference
After you read this topic, see the topic <u>How to Run Scheduled Tasks (Workstation Version)</u> in this chapter for step-by-step instructions.

About mtimer   For Windows platforms, you must have **mtimer** running at the time you want to run the unattended processes specified in the Task Scheduler.

Example
To run batch files daily, you must either activate **mtimer** daily or leave it on. You can leave **mtimer** on only if you leave your computer on with Windows running. You can close Gentran:Server without affecting **mtimer.**

The system displays the **mtimer** window whenever **mtimer** is on. This window displays currently scheduled jobs. The system displays only the title bar if no jobs are scheduled.

The Task Scheduler mtimer dialog box   This illustration shows an example of the Task Scheduler mtimer window.

Click here to display menu

Double-click to stop **mtimer**

```
Task Scheduler - mtimer                          _ □ ×
  ID    RC    DATE     TIME  BATCHFILE    Show Log  ▲
  --    --    ----     ----  ---------



◄                                              ►  ▼
Ready                              08/24/2001 12:08:17 PM
```
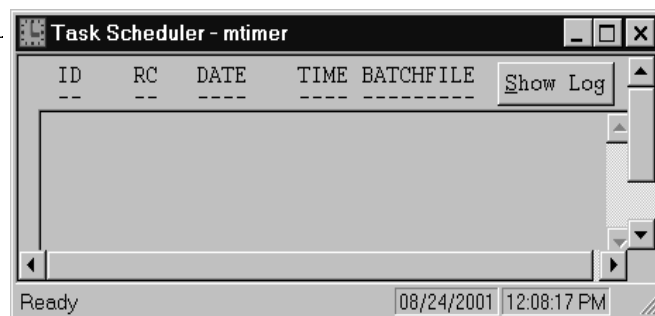
The **mtimer** window displays each scheduled task on a separate line. Each line contains the date and time as well as the batch file or script scheduled to run.

About mtimer files

When you activate **mtimer**, Gentran:Server creates three files. This table describes the files.

File	Description
unop.ran	Contains a list of tasks completed during this active session. Unless you elect to suspend processing, Gentran:Server deletes this file when you stop **mtimer**.
unop.log	Contains the list of completed tasks. Gentran:Server appends to this file with each new **mtimer** session, instead of deleting or overwriting it. This means it can grow very large. You should watch its size and periodically delete this log file manually. You can view this log on the MTimer Log dialog box while mtimer is running.
unop.err	Contains operating-system error messages in response to errors in the DOS commands included in your batch file.

MTimer Log dialog box

This illustration shows the MTimer Log dialog box.
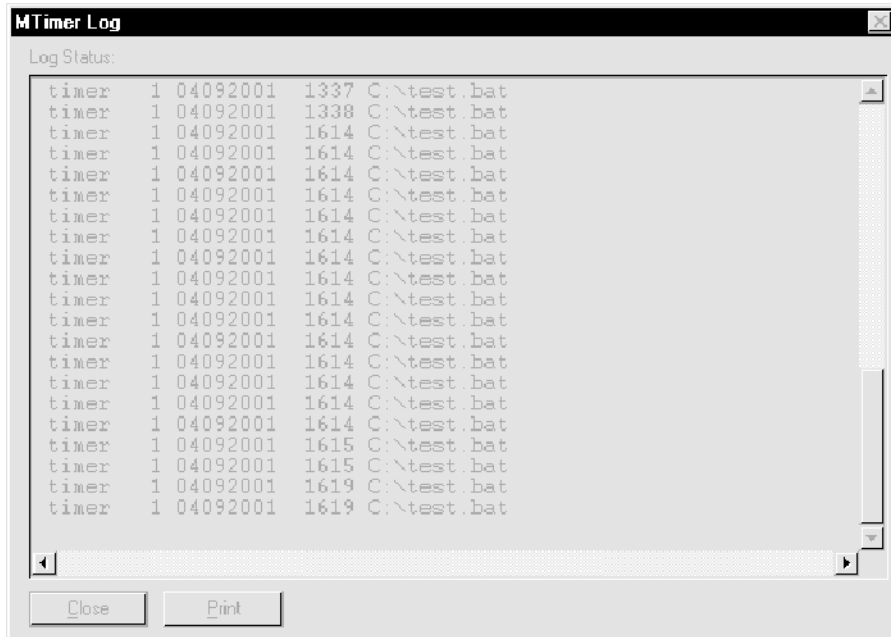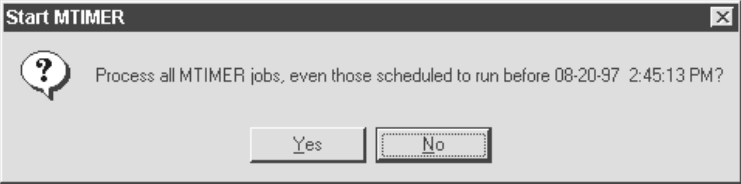
How to Run Scheduled Tasks (Workstation Version)
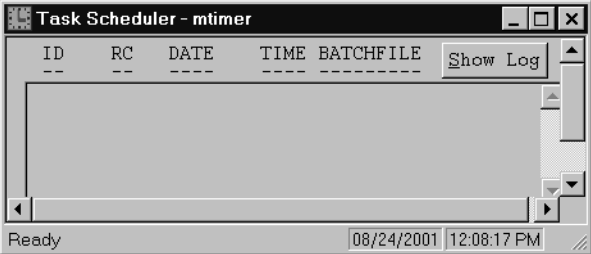
Introduction   This topic contains step-by-step instructions for running scheduled tasks from the Gentran:Server Tools menu in the Workstation version of Gentran:Server.

CAUTION

When you close Windows, mtimer automatically suspends job processing.

Procedure   Use this procedure to run scheduled tasks and activate **mtimer**.

Step	Action
1	Click **Run Task Scheduler** on the **Tools** menu.
	System Response
	Gentran:Server displays a message box that asks whether you want to run the tasks scheduled to run on the current date, but prior to the current time.
	(Continued on next page)

(Contd) Step	Action
2	Use this table to determine your action.

IF you want to...	THEN select...
Run all jobs, including those scheduled to run prior to the current time	Yes
Run only those jobs scheduled to run from the current time forward	No

System Response
The system displays the Tasks Scheduler - mtimer window.

3	Do you want to display the unop.log file in the MTimer Log dialog box?

◗   If YES, click the Show Log button.

◗   If NO, continue with the next step.

4	Do you want to stop mtimer?

◗   If YES, GO TO How to Stop and Resume mtimer.

◗   If NO, you are finished.

How to Stop and Resume mtimer

Introduction

This topic contains instructions for stopping (or suspending), and resuming **mtimer** in the Workstation version of Gentran:Server.

About stopping or suspending mtimer

At any time during processing, you can either stop **mtimer** or suspend job processing.
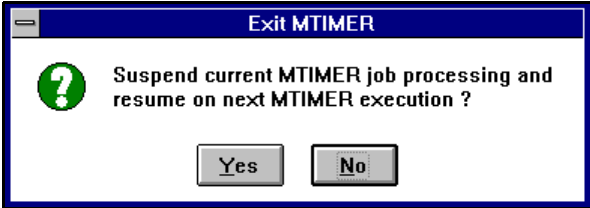
If you suspend job processing, Gentran:Server retains a list of the completed tasks. You have the option of resuming **mtimer** at any time in the future. When you resume **mtimer**, processing begins where it left off.

Stopping or suspending mtimer

Use this procedure to stop or suspend **mtimer**.

CAUTION

When you close Windows, mtimer automatically suspends job processing.

Step	Action
1	Double-click the control box bar in the **mtimer** window. **System Response** If there are jobs to be processed, Gentran:Server displays the following message. **Exit MTIMER** Suspend current MTIMER job processing and resume on next MTIMER execution ? [Yes] [No] **Note** The system does not display this message if there are no jobs to be processed. *(Continued on next page)*
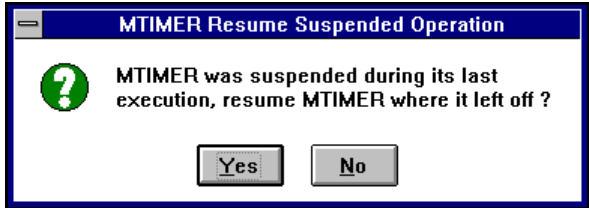
(Contd) Step	Action
2	Use this decision table to determine your action.

IF you want to…	THEN click…
Suspend job processing. Suspending job processing retains the list of completed tasks. When you activate **mtimer** again, the system only runs tasks not completed previously.	Yes
Stop **mtimer** and delete the list of completed tasks. The next time you activate **mtimer**, the system runs all tasks. This includes tasks run previously.	No
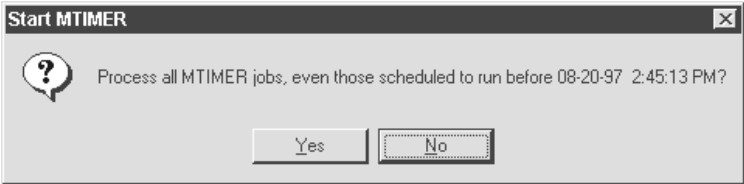
Note
When you suspend **mtimer**, the system retains the *unop.log* and *unop.err* files. However, if you stop **mtimer** and delete the list of completed tasks, Gentran:Server deletes the *unop.ran* file. This means that, the next time you activate **mtimer**, the system runs jobs that were already run.

Resuming mtimer

Use this procedure to resume **mtimer** after you have suspended processing.

Step	Action
1	Click **Run Task Scheduler** on the **Tools** menu. **Note** You can also type **mtimer** at the command line. **System Response** Gentran:Server displays a message box asking whether you want it to resume the schedule at the point at which it stopped. **MTIMER Resume Suspended Operation** MTIMER was suspended during its last execution, resume MTIMER where it left off ? Yes No (Continued on next page)

(Contd) Step	Action		
2	Use this decision table to determine your action.		
	IF you want to…	**THEN select…**	**AND Gentran:Server…**
	Have Gentran:Server read the *unop.ran* file and resume job processing where it last stopped	Yes	resumes processing, omitting the files listed in the *unop.ran* file. You are finished.
	Run the job processing from the beginning, even if it means running completed jobs again	No	displays a message box that asks whether you want to run the tasks scheduled to run prior to the current time. Continue with the next step.
3	When this dialog box is displayed, use the decision table below to determine your action. **Start MTIMER** ☒ (?) Process all MTIMER jobs, even those scheduled to run before 08-20-97 2:45:13 PM? [Yes] [No]		
	IF you want to…	**THEN select…**	
	Run the jobs scheduled to run today prior to the current time.	Yes	
	Run only those jobs scheduled to run from the current time forward.	No	
	System Response The system opens the Task Scheduler mtimer window.		

Error Messages

Contents

Overview

Introduction

The Gentran:Server error messages and other informational messages are noted in the Compile Errors dialog box when you compile the translation object or in the Error section of the Extended Rules dialog boxes when you compile an extended rule containing errors prior to compiling the translation object, and when you commit an erroneous action in Gentran:Server.

The informational messages are dependent on the context of the program, and are intended to be self-explanatory. The error messages are described in the topics accessed below, along with the actions you can take to correct the problem.

In this appendix

This appendix explains how to determine the action you should take when you receive an error message.

Map Editor Messages

Introduction   Gentran:Server errors can occur when you are creating a map. The system uses pop ups to notify you of an error, such as invalid or missing data.

Messages   The GENTRAN: Server error messages are listed below in alphabetical order by the first letter of the error message text. The table also includes an explanation of the error and an action you can take to correct the error.

When the system displays an error message, acknowledge the message by clicking **OK** and then take the appropriate action.

Message Text	Explanation/Your Action
A code list entry must have a code value	**Explanation** You attempted to add a code without an associated code value.
A code list must have an element id	**Explanation** You attempted to create a code list without an element ID.
A code list for this element already exists. You must use another element id or delete the original code list first	**Explanation** You attempted to create a code list with the same element ID as an already existing code list. **Your action** Either change the element ID or delete the original code list.
A condition field is required	**Explanation** You established a conditional relationship that requires a condition field, but did not specify a condition field.
A data type is required.	**Explanation** You did not specify the Data Type of a field. (Continued on next page)

(Contd) Message Text	Explanation/Your Action
A field must have a name that is unique within its parent group	**Explanation** You tried to give a field a name that is already in use by another field within the same group (not necessarily the same record).
A group must have a unique name	**Explanation** You tried to give a group a name that is already used by another group or record.
A group or record with a maximum usage of 1 cannot be split or promoted	**Explanation** You attempted to split or promote a single-occurrence group or record. This is not a valid action.
A key field has been selected but no key value has been specified.	**Explanation** You established a key field without specifying a key value.
A Memory Exception has occurred.	**Explanation** A system error occurred. **Your action** Exit Gentran:Server and restart Windows.
A name must be entered	**Explanation** You did not specify a name for an object.
A problem occurred while attempting to close loop .... This is probably due to incorrect standards.	**Explanation** An error occurred when reading from the standards.
A record must have a unique name	**Explanation** You tried to give a record a name that is already used by another group or record.
A Resource Exception has occurred.	**Explanation** A system error occurred. **Your action** Exit Gentran:Server and restart Windows. *(Continued on next page)*

(Contd) **Message Text**	**Explanation/Your Action**
A serious error was encountered while accessing the clipboard and the action was abandoned	**Explanation** The system aborted a cut, copy, or paste operation because a serious error occurred. **Your action** Perform the cut, copy, or paste operation again.
A system error was encountered while compiling the translation object	**Explanation** While compiling a translation object, a system-related problem, such as lack of memory, prevented the compile from completing. **Your action** Close unnecessary applications to free memory, and reboot your computer (if necessary), and then recompile the translation object.
A TDF could not be generated due to a lack of available memory. Either restart Windows or close down other applications, then try again	**Explanation** While generating a TDF, the system ran out of memory. **Your action** Exit Gentran:Server, restart Windows and regenerate the TDF.
An accumulator must be chosen for this entry	**Explanation** You did not specify the accumulator to be used in a Use Accumulator standard rule.
An invalid usage count has been entered	**Explanation** You entered an invalid usage count for a record or group.
No actions have been chosen for this accumulator entry	**Explanation** You selected an accumulator but did not specify any actions for it in a Use Accumulator standard rule.

(Contd) Message Text	Explanation/Your Action
No alternate accumulator has been selected	**Explanation** In a Use Accumulator standard rule, you attempted to create an accumulator entry that used an alternate accumulator, but did not specify which alternate accumulator should be used.
No character ranges have been specified for this token.	**Explanation** You did not specify the characters allowed for a syntax token.
No fields have been used in this relationship	**Explanation** You established a conditional relationship but did not specify the fields involved.
No more accumulator entries can be created	**Explanation** You attempted to create more than six accumulator entries in a Use Accumulator standard rule.
No more field mappings can be created	**Explanation** You attempted to create more than eight field mappings for a Select standard rule.
The constant ID must be unique.	**Explanation** You entered a literal constant identifier in the ID field that is already used in an existing constant. **Your action** Type a unique literal constant identifier in the ID field.
The contents of the clipboard cannot be pasted into the translation object at this point	**Explanation** Either the Clipboard does not contain copied or cut data, or you are trying to paste XML information into an application file format or paste application information into an XML file format. (Continued on next page)

(Contd) Message Text	Explanation/Your Action
The file format cannot be deleted	**Explanation** You tried to delete a file format object.
The Maximum Usage must be greater then zero and not less than the Minimum Usage.	**Explanation** You entered an invalid maximum usage count.
The number of entries to be split must be greater than zero and less than the maximum number of entries in the original	**Explanation** You entered an invalid number in the Split dialog box. The number to be split must be greater than zero and less than the maximum number of entries in the original.
The translation object could not be compiled	**Explanation** The translation object is not ready to be compiled.
The token code must be unique.	**Explanation** You attempted to create a syntax token with the same token identifier as an existing syntax token. **Your action** Type a unique token identifier in the Token field.
These fields cannot be linked because the input field is deeper than the output field. The translator would be unable to address the input field.	**Explanation** You are attempting to create an invalid link. A valid link involves an Input field and an Output field that are at the same level.
This code already exists. You must use another code or delete the original code first	**Explanation** You attempted to add a code to a code list that already contained that code.
This field cannot be linked to another field	**Explanation** Due to an unknown error, Gentran:Server was unable to begin creating the link. (Continued on next page)

(Contd) Message Text	Explanation/Your Action
This file is not a valid GENTRAN translation object	**Explanation** You attempted to open (load) a file that is not a translation object.
You have entered an invalid character or character code in Record Delimiter 1	**Explanation** You entered an invalid Record Delimiter 1 on the Positional File Format Properties dialog box.
You have entered an invalid character or character code in Record Delimiter 2	**Explanation** You entered an invalid Record Delimiter 2 on the Positional File Format Properties dialog box.
You have entered an invalid decimal separator or character code.	**Explanation** You entered an invalid decimal separator.
You have entered an invalid element delimiter character or character code	**Explanation** You entered an invalid element delimiter.
You have entered an invalid Pad character or character code	**Explanation** You entered an invalid pad character in a field.
You have entered an invalid release character or character code	**Explanation** You entered an invalid release character.
You have entered an invalid sub-element delimiter character or character code	**Explanation** You entered an invalid sub-element delimiter.
You have entered an invalid tag delimiter character or character code	**Explanation** You entered an invalid tag delimiter. (Continued on next page)

(Contd) Message Text	Explanation/Your Action
You must enter a description of the translation object.	**Explanation** You attempted to save the translation object without entering the translation object description on the Translation Object Details dialog box. **Your action** Type the translation object description in the Description field on the Translation Object Details dialog box. This field must be unique because it is used by the system to identify the map.
You must enter a subtable name.	**Explanation** For a Select or Update standard rule, the selected table requires you to specify a table name in the Sub Table field.
You must enter a valid character range.	**Explanation** You entered an invalid character range for a syntax token.
You must enter a valid syntax token.	**Explanation** You did not enter a valid token identifier in the Token field when creating or editing a syntax token.
You must enter a value for this constant.	**Explanation** You did not specify a value for the constant in the Value field when creating or editing a syntax token.
You must enter an ID.	**Explanation** You did not specify a literal constant identifier for the constant in the ID field when creating or editing a syntax token.
You must enter the name of the author of this translation object	**Explanation** You attempted to save the translation object without entering the name of the translation object author on the Translation Object Details dialog box. (Continued on next page)

(Contd) Message Text	Explanation/Your Action
You must select a constant type.	**Explanation** You did not select a constant type from the Type list when creating or editing a constant.
You need to specify Input and Output File Types for the new translation object	**Explanation** You clicked **OK** on the Create New Translation Object dialog box without selecting both the Input Format Type and Output Format Type.

Compiler Messages

Introduction   The compile errors can occur when you compile a map or Extended Rules. If there are errors, then the system lists them either in the Compile Error dialog box or in the error section of the Extended Rules dialog box.

Messages   The compile error messages are listed below in numerical order based on their Message ID. The table also includes an explanation of the error and an action you can take to correct the error.

After you correct the cause of the errors, click **Compile** again to ensure that the rule is error-free.

Msg ID	Message Text	Explanation/Your Action
1000	expected '.'	**Explanation** The rule does not have the required "." between a group name and a field name. **Your action** Insert a "." between the group name and field name.
1001	no statement to compile	**Explanation** The rule is does not contain any statements. **Your action** Add a statement or statements to the rule.
1002	unexpected end of program	**Explanation** The rule was not complete. **Your action** Finish the rule.
1003	expected ','	**Explanation** The rule does not have the required "," between parameters. **Your action** Insert a "," between parameters. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
1004	expected ','	**Explanation** The statement was not terminated properly. **Your action** Terminate the statement.
1006	no statements to compile	**Explanation** The body of an IF/ELSE or WHILE condition was empty. **Your action** Complete the body of the unfinished condition.
1007	syntax error ...	**Explanation** The rule contains a syntax error. **Your action** Correct the syntax error.
1008	expected '#'	**Explanation** The rule does not have the required "#" before a field name. **Your action** Insert a "#" before the field name.
2000	group ... undefined	**Explanation** The rule references a group that does not exist. **Your action** Change the reference to an existing group or delete the reference.
2001	... is not a member of ...	**Explanation** The rule references a field that does not belong to the specified group. **Your action** Change the reference to an existing field in the specified group. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
2002	insufficient indices to access group ...	**Explanation** The rule does not give the full addressing for a group. **Your action** Complete the addressing for the group.
2003	too many indices to access group...	**Explanation** The rule uses too many addresses for the group. **Your action** Address the group correctly.
2004	... has not been defined	**Explanation** The rule references an undefined variable. **Your action** Define the variable in the declarations section.
2005	out of temporary variables	**Explanation** The rule could not be compiled because some expressions are too complex. **Your action** Simplify the expressions and compile the rule again.
2008	field type unknown	**Explanation** The compiler was unable to determine the type of a field. **Your action** Verify that a Data Type is selected for this field. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
2009	cannot reference a local field with no current group	**Explanation** The rule (probably pre- or post-session) references a local field, but is not associated with any group. **Your action** Reference the field using the proper addressing.
2010	instances of '%1' are transient and cannot be accessed	**Explanation** The rule references an output group using full addressing. This form of addressing is only appropriate for input groups. **Your action** Reference the output group with the proper address.
2011	no field specified	**Explanation** The rule omits a field reference. **Your action** Add the field reference to the rule.
2012	group ... is promoted and cannot be accessed in this manner	**Explanation** You tried to access a promoted group with the COUNT or DELETE function. **Your action** Do not use COUNT or DELETE with a promoted group.
20700	only one binary data and one binary length field are permitted	**Explanation** You have more than one binary data and/or more than one binary length element in one segment. **Your action** Remove the additional binary data and/or binary length elements from the segment.

(Contd) Msg ID	Message Text	Explanation/Your Action
20701	binary length must precede binary data	**Explanation** The binary length element must be sequenced *before* the binary data element in the segment so the translator knows how much data to expect. **Your action** Move the binary length element to before the binary data element.
20702	incomplete binary data	**Explanation** You marked a segment as binary, but did not include either a binary length element or a binary data element (or both). **Your action** Add a binary length element and/or a binary data element to the segment.
20703	group [group] has no active child objects	**Explanation** You tried to compile the translation object, but the specified group is empty. **Your action** Activate at least one child object in the group.
2100	... is not an array variable	**Explanation** The rule uses array indexing for a variable that is not an array. **Your action** Use the proper indexing for the variable.
2101	... : array index required	**Explanation** The rule uses an array variable without using the necessary array indexing. **Your action** Add the necessary indexing to the array variable. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
2102	... : array overflow	**Explanation** The rule uses an invalid array index. **Your action** Use the proper array index for the array variable.
2103	only one wildcard index is permitted	**Explanation** The rule uses more than one wildcard index. **Your action** Only one wildcard index is permitted per rule.
2104	a wildcard index must be specified	**Explanation** The rule does not specify a wildcard index when required. **Your action** Add a wildcard index where necessary.
2200	expected a string	**Explanation** A required string or string variable is not supplied. **Your action** Add the required string or string variable.
2201	string overflow	**Explanation** A string overflow occurred. **Your action** Verify that the size of a target string is equal to or greater than the source string.
3000	array size expected in declaration of ...	**Explanation** Invalid array declaration.
3001	declaration of ... missing ']'	**Explanation** Invalid array declaration.
3002	string size expected	**Explanation** Invalid string declaration. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
3003	string size missing ']'	**Explanation** Invalid string declaration.
3004	variable name expected	**Explanation** Invalid variable declaration.
3005	... already defined	**Explanation** Two variables with the same name are defined at the same scope. **Your action** Rename one of the two variables.
3006	expected an accumulator number, found ...	**Explanation** Invalid accumulator reference.
3007	... is not a valid accumulator number	**Explanation** Invalid accumulator reference.
4000	expected a numeric expression, found ...	**Explanation** You specified something other than the expected numeric expression. **Your action** Specify the correct numeric expression.
4001	expected a term, found ...	**Explanation** You specified something other than the expected term. **Your action** Specify the correct term.
4002	expected '+' or '-'	**Explanation** You specified something other than the expected "+" or "-". **Your action** Specify "+" or "-". (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
4003	expected '*' or '/'	**Explanation** You specified something other than the expected "*" or "/". **Your action** Specify "*"or"/".
4004	expected ')'	**Explanation** You specified something other than the expected ")". **Your action** Specify ")".
4005	expected a factor, found ...	**Explanation** Invalid numeric expression.
4006	expected '('	**Explanation** You specified something other than the expected "(". **Your action** Specify "(".
4007	... is of incorrect type	**Explanation** The specified expression is of an incorrect type. **Your action** Specify the correct type for the expression.
4008	expected a relational operator	**Explanation** You specified something other than the expected relational operator. **Your action** Specify the correct relational operator.
4009	missing argument	**Explanation** A required parameter was omitted. **Your action** Add the required parameter. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
4010	assignment expected	**Explanation** The assignment operator was omitted from an assignment statement. **Your action** Add the correct assignment operator to the statement.
4011	operator ... requires two arguments	**Explanation** Only one parameter was supplied for a binary operator. **Your action** Supply a second parameter for the binary operator.
4012	converting real to integer may lose significant digits	**Explanation** A real number is being converted to an integer. **Your action** Verify that loss of the decimal places is acceptable.
4013	expression too complex, use sub-expressions	**Explanation** The mathematical expression contains too many terms. **Your action** Group some of the terms in parentheses.
4014	expected an integer, found ...	**Explanation** The compiler expected to find an integer, but instead it found ... **Your action** Replace the found value with the correct value.
4015	no valid condition specified	**Explanation** The statement does not have a valid IF or WHILE condition. **Your action** Supply a valid condition in the IF or WHILE statement. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
4100	expected a date, found ...	**Explanation** A date is required but not supplied. **Your action** Specify a date.
4101	expected a date modifier, found ...	**Explanation** A date modifier was required but not supplied. **Your action** Specify a date modifier.
4102	... is not a date	**Explanation** Invalid date expression.
5000	THEN expected	**Explanation** The compiler expected a THEN condition.
5001	DO expected	**Explanation** The compiler expected a DO condition.
5002	END expected	**Explanation** The compiler expected a END condition.
5004	FROM expected	**Explanation** The compiler expected a FROM condition.
5005	INTO expected	**Explanation** The compiler expected a INTO condition.
5006	END without BEGIN	**Explanation** An END statement was found without a corresponding BEGIN statement. **Your action** Insert a BEGIN statement in the correct location. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
5016	misplaced 'BREAK'	**Explanation** The compiler found a BREAK function outside of a loop. **Your action** Remove the BREAK function or place it inside a loop.
5017	misplaced 'CONTINUE'	**Explanation** The compiler found a CONTINUE function outside a loop. **Your action** Remove the CONTINUE function or place it inside a loop.
5018	too many parameters	**Explanation** Too many parameters were supplied for a function. **Your action** Remove the unnecessary parameters.
5019	too few parameters	**Explanation** Too few parameters were supplied for a function. **Your action** Add the necessary parameters.
7000	... is not a valid seek type	**Explanation** The seek type in FSEEK is invalid. **Your action** Use BEGIN, END, or CURRENT for the seek type..
20001	Record ..., Field ...:date field missing date format	**Explanation** The specified field does not have a date format. **Your action** Edit the field and choose a date format. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
20003	Field ... : constant used in standard rule does not exist	**Explanation** The field's standard rule uses an invalid constant. **Your action** Correct the standard rule.
20004	Field ... : code list used in standard rule does not exist	**Explanation** The field's standard rule uses an invalid code list. **Your action** Correct the standard rule.
20005	Field ... : the qualifier field specified in a use constant standard rule is invalid	**Explanation** The field's standard rule uses an invalid qualifier field. **Your action** Correct the standard rule.
20006	Field ... : the field specified to store the code description in a use code standard rule is invalid	**Explanation** The field's standard rule uses an invalid code description field. **Your action** Correct the standard rule.
20007	Record ... : the specified key field ... : uses an undefined constant	**Explanation** The record's key field uses an invalid constant. **Your action** Correct the key field.
20008	Record ... : the specified key field ... : uses an undefined code list	**Explanation** The record's key field uses an invalid code list. **Your action** Correct the key field.
20010	Record ... : the specified key field ... : is inactive	**Explanation** The record's key field is inactive. **Your action** Activate the key field. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
20704	Element ..., Attribute ... : enumerated attribute declared without accompanying standard rule	**Explanation** The XML attribute specified is configured to use an enumeration but there is no code list standard rule to define allowed values **Your action** Define a use code list standard rule for this attribute.
20705	Element ..., Attribute ... : code list used in enumerated attribute does not exist	**Explanation** The code list specified for this enumerated XML attribute does not exist. **Your action** Define the code list.
20706	Element ..., Attribute ... : default value is not valid ...	**Explanation** The specified default is not in the code list for this XML attribute. **Your action** Make sure that the default value is in the code list.
20707	Codelist ..., Attribute ... : value used in enumerated attribute code list does not match XML NMTOKEN production	**Explanation** A value in the enumeration code list is not legal in XML. **Your action** Make sure that all the values are legal for an attribute in XML.
20708	Entity ... : Entity value is invalid	**Explanation** The specified entity's value is not legal in XML. **Your action** Correct the entity's value.
20709	... : Illegal use of reference character	**Explanation** An entity reference was not properly terminated. **Your action** Correct the entity reference. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
20710	... : Malformed character reference encountered.	**Explanation** An XML character reference was not properly terminated. **Your action** Correct the character reference.
20711	... : Invalid character referenced.	**Explanation** An XML character reference was invalid. **Your action** Correct the character reference.
20712	... : Reference to undefined entity encountered	**Explanation** The referenced entity is not defined. **Your action** Define the correct entity.
20713	... : Circular entity references encountered.	**Explanation** An entity references an entity which references the original entity. **Your action** Do not reference the original entity in this circular manner.
20714	... : default does not match the attribute type.	**Explanation** The default value is the wrong type for the attribute. **Your action** Correct the type of the attribute or the default value.
20715	... : Contains illegal character ('<')	**Explanation** The default value contains '<', which is illegal. **Your action** Correct the default value.

Translator Report Error Messages

Introduction

The translator report error messages can occur when the translator cannot translate a document, read input data, or write output data.

Types of messages

This table lists the types of translation messages that can occur and their corresponding message ID range.

Type of translation messages	Message ID range
General translation messages	100-499
XML translation messages	600-699
ODBC translation messages	700-799
NCPDP translation messages	800-899

References

See the cerror topic in the Using Extended Rules chapter of this guide for more information about these errors.

See the NCPDP Translator Messages topic in this guide for information on resolving translator messages that occur when translating NCPDP data.

NCPDP Translator Messages

Introduction

All errors that occur during translation are listed in the Translator Report. There are three types of errors that can occur when translating NCPDP data.

◗ NCPDP messages

◗ Input error messages that result from data that the translator is reading

◗ Output error messages that result when the translator attempts to write to the output file

NCPDP Messages

NCPDP messages can occur when the translator cannot translate a document that is in an NCPDP format.

The messages are listed below in numerical order based on their Message ID. The table also includes an explanation of the error and an action you can take to correct the error.

Msg ID	Message Text	Explanation/Your Action
800	Batch File Header Section Missing	**Explanation** The system cannot find a header section record within the batch file. **Your action** Contact your Trading Partner.
801	Unexpected Header Segment ID	**Explanation** A value other than "00" is in the header section record segment Identifier field (701). **Your action** Contact your Trading Partner.
802	Unexpected Transmission Type	**Explanation** A transmission value other than T (Transaction), R (Response), or E (Error) is in the transmission type (880-K6) field of the header section record. **Your action** Contact your Trading Partner. *(Continued on next page)*

(Contd) Msg ID	Message Text	Explanation/Your Action
803	Sender ID Missing	**Explanation** The system cannot find a value in the sender ID (880-K1) field in the Header Section Record. **Your action** Contact your Trading Partner.
804	Invalid Batch Number	**Explanation** The batch number field (806-5C) contains a format other than the YYDDD format, where: ‣ YY=Year ‣ DDD=Julian date **Your action** Contact your Trading Partner.
805	Unexpected File Type	**Explanation** A value other than P (Production) or T (Test) is in the File Type field (702). **Your action** Contact your Trading Partner.
806	Receiver ID Missing	**Explanation** The system cannot find a value in the receiver ID field (880-K7) of the header section. **Your action** Contact your Trading Partner.
807	Batch File Detail Data Record Missing	**Explanation** The system cannot find a Detail Data Record within the batch file. **Your action** Contact your Trading Partner. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
808	Unexpected Data Segment ID	**Explanation** A value other than "G1" is in the segment identifier field of the Detail Data Record (701). **Your action** Contact your Trading Partner.
809	Transaction Reference Number Missing	**Explanation** The system cannot find a value in the transaction reference number field of the Detail Data Record (880-K5). **Your action** Contact your Trading Partner.
810	Batch File Trailer Record Missing	**Explanation** The system cannot find the trailer record within the batch file. **Your action** Contact your Trading Partner.
811	Invalid Trailer Segment ID	**Explanation** A value other than "99" is in the segment identifier field (701) of the trailer record. **Your action** Contact your Trading Partner.
812	Unexpected Batch Number	**Explanation** The batch number field (806-5C) of the trailer record does not match the batch number field (806-5C) of the header record. **Your action** Contact your Trading Partner.
813	Invalid Record Count	**Explanation** The record count field (751) of the trailer record contains an invalid number. **Your action** Contact your Trading Partner. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
814	Processed DDRs Mismatch Record Count	**Explanation** The actual count of Detail Data Records is different from the value in the record count field (751) of the trailer record. **Your action** Contact your Trading Partner.
815	Duplicate Segment Within Transaction	**Explanation** A segment occurs more than one time within the transaction. **Your action** Contact your Trading Partner.
816	Unexpected Transaction Code	**Explanation** A value other than a value listed below is in the transaction code field (103-A3) of the request transaction header segment or response header segment. ▸ E1 = Eligibility Verification ▸ B1 = Billing ▸ B2 = Reversal ▸ P1 = Rebill ▸ P2 = P.A. Request & Billing ▸ P3 = P.A. Inquiry ▸ P4 = P.A. Request Only ▸ N1 = Information Reporting ▸ N2 = Information Reporting Reversal ▸ N3 = Information Reporting Rebill ▸ C1 = Controlled Substance Reporting ▸ C2 = Controlled Substance Reporting Reversal ▸ C3 = Controlled Substance Reporting Rebill **Your action** Contact your Trading Partner. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
817	Invalid Transaction Count	**Explanation** A value other than a value listed below is in the transaction count field (109-A9) of the request transaction header segment or the response transaction header segment. ◗ Blank = not specified ◗ 1 = One Occurrence (Default) ◗ 2 = Two Occurrence (Except Transaction Code E and P) ◗ 3 = Three Occurrence (Except Transaction Code E and P) ◗ 4 = Four Occurrence (Except Transaction Code E and P) **Your action** Contact your Trading Partner.
818	Processed Transactions Mismatch Transaction Count	**Explanation** The actual count of the transactions is different from the value in the transaction count field (109-A9) of the response header segment. **Your action** Contact your Trading Partner.

(Contd) Msg ID	Message Text	Explanation/Your Action
819	Unexpected Header Response Status Code	**Explanation** A value other than A (Accepted) or R (Rejected) is in the header response status field (501-F1) of the response header segment. **Your action** Contact your Trading Partner.
820	Unexpected Transaction Response Status Code	**Explanation** A value other than a value listed below is in the transaction response status field (112-AN). ◗ A = Approved ◗ C = Captured ◗ D = Duplicate of Paid ◗ F= PA Deferred ◗ P = Paid ◗ Q = Duplicate of Capture ◗ R = Rejected ◗ S = Duplicate of Approved **Your action** Contact your Trading Partner.

Input error messages

Input error messages can occur when the translator is reading input data.

The input error messages are listed below in numerical order based on their Message ID. The table also includes an explanation of the error and an action you can take to correct the error.

Msg ID	Message Text	Explanation/Your Action
830	Null Character	**Explanation** The data file contains binary data. **Your action** Contact your Trading Partner.
833	Unexpected End of File	**Explanation** A field of a specified length is incomplete. **Your action** Contact your Trading Partner.
835	Unexpected Start of Text	**Explanation** The record does not contain End of Text character. **Your action** Contact your Trading Partner.
836	Unexpected End of Text	**Explanation** The record does not contain Start of Text character. **Your action** Contact your Trading Partner.
837	Missing Stream Separator1	**Explanation** The system is reading non-streamed data with a template from a map where "stream segments" box is unchecked. **Your action** Contact your Trading Partner. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
838	Missing Stream Separator2 (Windows Only)	**Explanation** The system reading non-streamed data with a template from a map where "stream segments" box is unchecked. **Your action** Contact your Trading Partner.
839	Unexpected Separator	**Explanation** A group separator, segment separator, field separator, STX, or ETX is present in the data. **Your action** Contact your Trading Partner.
840	Unexpected Field Separator	**Explanation** Based on the map, two field separators occur consecutively or a different separator is expected. **Your action** Contact your Trading Partner.
841	Unexpected Group Separator	**Explanation** Based on the map, a group separator occurs when none was specified. **Your action** Contact your Trading Partner.
842	Missing Group Separator	**Explanation** Based on the map, a group separator is missing. **Your action** Contact your Trading Partner.
843	Unexpected Stream Separator1	**Explanation** Based on the map, a stream separator1 is occurs when none was specified. **Your Action** Contact your Trading Partner. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
845	Positional Tag Defined Beyond EOS	**Explanation** The position of the transaction code (Positional Record Tag) is specified to start beyond the end of the segment. **Your action** Change the value of the positional segment under the NCPDP Properties tag to start before the end of the segment.
846	Missing Positional Record Separator1	**Explanation** Based on the map, an expected positional record separator1 is missing. **Action** Contact your Trading Partner.
847	Missing Positional Record Separator2	**Explanation** Based on the map, an expected positional record separator2 is missing. **Action** Contact your Trading Partner.
848	Positional Segment Has No Fields	**Explanation** The positional segment must have data. **Action** Contact your Trading Partner.
849	Batch Segment Has No Fields	**Explanation** The batch segment must have data. **Action** Contact your Trading Partner.
855	Delimited Segment Has No Fields	**Explanation** The delimited segment must have at least the mandatory fields. **Your action** Contact your Trading Partner. *(Continued on next page)*

(Contd) Msg ID	Message Text	Explanation/Your Action
856	Missing Delimited FS After SS	**Explanation** The delimited field separator is not present after the segment separator. **Your action** Contact your Trading Partner.
857	Field ID Data Does Not Match Map Field ID	**Explanation** The field ID data is different from the defined delimited field ID tag. **Your action** Contact your Trading Partner.
858	Unexpected Char(s) in Delimited Field ID	**Explanation** A separator was found in the delimited field ID data. **Your action** Contact your Trading Partner.
859	Null Character in Delimited Field ID	**Explanation** The delimited field ID contains binary data. **Your action** Contact your Trading Partner.
860	Incomplete Delimited Field ID	**Explanation** The field ID data is shorter than expected. **Your action** Contact your Trading Partner.
861	Missing Delimited Field Data	**Explanation** No data is found after a delimited field ID. **Your action** Contact your Trading Partner.
862	Null Character in Delimited Field Data	**Explanation** The delimited field data contains binary data. **Your action** Contact your Trading Partner. (Continued on next page)

(Contd) Msg ID	Message Text	Explanation/Your Action
865	Segment ID Data Does Not Match Map Segment ID	**Explanation** The segment ID data is different from the defined delimited segment ID field. **Your action** Contact your Trading Partner.
866	Unexpected Char(s) in Delimited Segment ID	**Explanation** A separator was found in the delimited segment ID. **Your action** Contact your Trading Partner.
867	Null Character in Delimited Segment ID	**Explanation** The delimited segment ID contains binary data. **Your action** Contact your Trading Partner.
868	Incomplete Delimited Segment ID	**Explanation** The delimited segment ID is shorter than expected. **Your action** Contact your Trading Partner.
869	Field ID Size Differs From Global Field ID Size	**Explanation** The computed size of the segment/field ID is different from the value of the global field ID length. **Your action** Contact your Trading Partner.
885	No Data after STX in Batch Segment	**Explanation** The batch record does not contain any data. **Your action** Contact your Trading Partner. *(Continued on next page)*

(Contd) Msg ID	Message Text	Explanation/Your Action
886	Unexpected STX in Batch Segment	**Explanation** The batch record does not contain an ETX character. **Your action** Contact your Trading Partner.
887	Missing ETX in Batch Segment	**Explanation** The batch record does not contain an ETX character. **Your action** Contact your Trading Partner.

Output error messages

Output error messages can also occur when the translator tries to write to the output file. Many times the error results because the Batch File string is missing a portion of the Batch File or a value in the string is incorrect.

To prevent errors from occurring verify that the Batch File string follows the detail data record (DDR) format:

<separator>filename<separator>StartOffset<separator>EndOffset<separator>.

The output error messages are listed below in numerical order based on their Message ID. The table also includes an explanation of the error and an action you can take to correct the error.

Msg ID	Message Text	Explanation/Your Action
888	Could Not Open NCPDP Data File	**Explanation** The NCPDP data file either does not exist or it is not readable. **Your action** Verify that the file you are selecting is the correct filename or supply the information that is missing from the string field.
890	Missing Separator in DDR Info String	**Explanation** A separator is missing from the DDR string field. **Your action** Insert a ";" in the DDR string field.
891	Missing Filename in DDR Info String	**Explanation** The filename is not in the DDR string field. **Your action** Add the filename to the DDR.
892	Invalid Start byte Offset in DDR Info String	**Explanation** The system found non-numeric data at start of set. **Your action** Add a valid start of set (numeric) value. *(Continued on next page)*

Msg ID	Message Text	Explanation/Your Action
893	Invalid End Byte Offset in DDR Info String	**Explanation** The system found non-numeric data at end of set. **Your action** Add a valid end of set (numeric) value or remove the data if data up to end of file should be processed.
894	Missing Start/End Byte Offset in DDR Info String	**Explanation** The Start/End Byte Offset is not in the DDR string field. **Your action** Add the Start/End Byte Offset to the DDR string field or remove extra separators.
895	Extra Data in DDR Info String	**Explanation** The DDR string contains extra data. **Your action** Remove the extra data from the DDR string.
896	Total Bytes Read Mismatch Total Bytes to Read	**Explanation** The actual number of bytes read is less than the number of bytes specified in the set. **Your action** Verify the Start/End Byte Offsets or remove the end byte offset if data up to end of file should be processed. (Continued on next page)

Msg ID	Message Text	Explanation/Your Action
897	Start Byte Offset is Beyond End of File	**Explanation** The position of the Start Byte Offset is specified to start after the end of the file. **Your action** Change the start position of the Start Byte Offset.
898	Start Byte Offset is Beyond End Byte Offset	**Explanation** The position of the Start Byte Offset starts after the position of the End Byte Offset. **Your action** Change the position of the Start Byte Offset to start before the End Byte Offset, or change the position of the End Byte Offset to start after the Start Byte Offset.

Glossary

acknowledgment	The ANSI 997 functional acknowledgement, the EDIA 999 acceptance/rejection advice, and the EDIFACT CNTRL document.

activation	The process of turning on groups, segments, composites, and elements that a standard does not define as mandatory, but that you have determined that you need to use in a map.

application file	A *fixed-format, variable-length file,* or a *positional file* that contains all the information that you need to extract from your partner's document or send to your partner so that your system can accurately process the data.

Application Integration subsystem	The Gentran:Server for UNIX mapping software that enables you to create maps that translate:

◗ your application files to EDI standard formats for documents you send to your trading partners (outbound mapping), and

◗ EDI standard formats to your application format for documents that you receive from your trading partners (inbound mapping).

Note
If you are using the XML or ODBC translation options, then you can also create maps that translate XML and ODBC data.

application system	Computer systems, outside of EDI, designed to fulfill specific business functions. These include accounting, purchasing, materials control, human resources, shipping, and other systems.

Application-to-application map	A map that describes how to translate an application document to another application format.

Application-to-EDI map	A map that describes how to translate your application document to an EDI document destined for your trading partner.
Application-to-XML map	A map that describes how to translate your application document to an XML document destined for your trading partner.
archive	The process of copying and saving EDI standard data that has been processed.
Auto trim	The Application Integration feature that enables you to automatically activate and deactivate map components on the EDI side of a map by using a sample EDI file as a model.
batch file	A file that contains command-line commands and DOS commands.
category	A user-definable record that enables you to group Trading Partnerships.
code list	A roster of code list tables that have been created or loaded in Gentran:Server.
code list tables	A table that contains a list of valid codes for an element in an EDI standard.
colours	The Application Integration feature that enables you to select foreground and background colours to visually define the various map components.
command	A keyword that performs actions on variables. Commands are responsible for moving data.
compile	The process used to convert a map into a translation object.

component data element	A data element that belongs to a composite data element. Component data elements are also called sub-elements.
composite	A data element that contains two or more component data elements or sub-elements.
composite data element	A data element that contains two or more component data elements or sub-elements. Composites are defined by the EDI standards that use them (EDIFACT, TRADACOMS, and certain ANSI X12 standards).
conditions	See *relational conditions*.
constant standard rule	A standard rule that enables you to move a literal constant value to a specified field or generate qualifiers to indicate a qualifying relationship with another field.
contact record	A record containing the name, address, and telephone numbers of an individual at your trading partner's business.
cron	A UNIX operating-system utility that runs commands and scripts according to a preset schedule. This is the executable file that activates the timer in Gentran:Server for UNIX.
cross reference table	A table that associates strings with specific Trading Partnerships.
data file	The file that contains the inbound or outbound data.
data definition format	A file that defines a file format used in a map. It includes the hierarchical and looping structure of the data, the map objects, and the attributes of the objects.

date/time type

A type of field that contains a date or time. If you specify that a field is a date/time type, you must specify exactly how the date or time must be formatted.

deactivation

The mapping function that enables you to make non-mandatory map components unavailable for use by the system.

default

A predefined value. Programs use these built-in values *unless* you specifically override them.

delimiters

Flags that you define to the system as separators for specific EDI components.

document

One transaction set containing actual data and treated as a single entity. The amount of data does not affect whether something is a document or not, only the fact that it is treated as a single entity. For example, regardless of whether a purchase order contains one item or ten thousand, if it is one purchase order, it is one document.

If a trading partner sends you a communication containing 10 purchase orders, you have received 10 documents. If the communication contained 15 invoices, you have received 15 documents.

See also transaction set.

EDI

Electronic Data Interchange (EDI). The process by which companies can exchange business documents directly from application to application by computer without paper documents being produced.

EDI History Audit directory

The directory used to store permanent history files and EDI data.

EDI standard

The rules for turning a business document into an EDI document.

EDI Status Audit directory	The directory used to store temporary translation audit files (*edistat.i* and *edistat.o*).
EDI-to-Application map	A map that defines how to move data from the EDI standard-formatted documents that your partners send you to your application file.
EDI-to-EDI map	A map that defines how to translate data that is in an EDI standard format into another EDI standard format.
EDI-to-XML map	A map that defines how to translate data that is in an EDI standard format into XML format.
ediarc	The Gentran:Server program that controls archiving.
edifrmat	The utility that removes segment terminator and replaces it with a new-line terminator.
element	The smallest piece of usable information defined by the standards. An element is the EDI map component that is mapped (linked) to a corresponding application field to move data to and from the EDI file. 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. See composite data element and component data element.

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

There are three levels of envelopes, as follows:

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

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

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

equalize
The mapping function that enables you to reinstate the two sides of the map with focus in equal dimensions.

expression
A logical unit of information that the system evaluates.

extended rules
The Application Integration feature that enables you to apply complex mapping operations to fields. These rules give you access to the Gentran:Server proprietary programming language.

field
The smallest piece of information defined in an 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.

file format definition
A file that defines how fixed-format data needs to be formatted for an application to process it. These files have a .DDF extension.

File format definition files contain a layout of the records, fields, and groups in an application file.

fixed-format file	See [application file]().
Font feature	The mapping feature that enables you to globally change the font type, style, and point size that is used in the display of all maps.
functional acknowledgment (FA)	The standard transaction set used to acknowledge receipt of a transmission. The functional acknowledgment tells you if a document you sent was received and processed by your trading partner. It also can tell you if your document contained EDI standards compliance errors. Likewise, the functional acknowledgment you send to your trading partner relays the same information.
functional group	A group of transaction sets that the standards-setting body (such as ANSI) has defined as fitting together with other related transaction sets. An example might be a Functional Group defined as containing all purchasing transaction sets.
global scope	A variable in an extended rule that can be accessed from any other extended rule in the map.
group	A looping structure that contains related records/segments and/or groups that repeat in sequence until either the group data ends or the maximum number of times that the loop is allowed to repeat is exhausted.
Group Organization record	A record containing all information specific to a single division or department in your trading partner's organization.
historical record	The index files named *edihist.dat* and *edihist.idx* that store the location of the archived data. The files are built from the status records that translation produces. Together they form the permanent audit file.
history audit directory	The EDI History Audit (*hisaud*) directory used to store permanent history files.

identification string	The value that results from applying a set of Trading Partnership rules to an application or XML document.

inbound mapping	The process of creating a map that defines:

◗ The EDI standard format in which your trading partner sends documents to you

◗ Your application file format (how the data needs to be formatted for your application to process it).

Note
If you have the XML or ODBC translation option, then the process of creating maps also defines:

◗ The XML format in which your trading partner sends documents to you

◗ The ODBC format in which your trading partner sends documents to you

inbound translation	A translation in which the input format of the source document is either an EDI standard format or an XML format that you use to exchange documents with a trading partner.

Note
Inbound translation using the XML format is available only if you have the XML translation option.

integer	A number that can have an implied decimal point (e.g., "2.01" is formatted as "201").

interchange	All the functional groups of documents (transaction sets) sent from one sender to one receiver in the same transmission.

Interchange Organization record	A record containing all information specific to a single company.

Interchanges browser	The Gentran:Server browser that enables you to view a hierarchical list of all the interchanges that were sent or received. It contains status information about the interchanges, (e.g., whether or not a functional acknowledgement was received and the status of that acknowledgement). This browser also enables you to access the documents within the interchanges. The Interchanges browser can always be accessed.
keyword	A fixed defined use of a word that indicates how the programming language should be interpreted.
link	The function that connects or maps a field from the Input side of the map to a field on the Output side of the map. The link between two map components is visually represented with a line connecting the two fields.
literal constant	A constant that the system uses as a repository to store information that is used at a later point in the map. Typically, constants are used in an outbound map to generate a qualifier.
loop	A mapping structure in which the records or groups repeat in a sequence until either the group data ends or the maximum number of times that the loop is allowed to repeat is exhausted.
loop count standard rule	The function that enables you to track the number of times a loop is repeated. If the field is part of a loop. If the loop is a nested loop, you can track the current loop *or* the outer loop. For example, if the Y loop is nested within the X loop, and the Y loop has cycled through 15 iterations and the X loop has cycled through 3 iterations, you can choose to count either the "15" (Y loop) or the "3" (X loop).
loop start/loop end segments	Loop Start (LS) and Loop End (LE) Segments, used by certain EDI standards, differentiate between two or more loops of the same type.
many-to-many mapping relationships	A mapping relationship that has a looping structure.

map A set of instructions that indicates the corresponding relationship between your application file input and output, and defines how the system should translate data.

map object A component object of a map. For example:

◗ Positional or Delimited EDI file

◗ group

◗ segment

◗ record

◗ element

Note
If your company purchased the XML translation, other map objects include:

◗ pcdata

◗ attribute

◗ attribute container

◗ content particle

message See transaction set.

mtimer A feature you can use to start, stop, or resume unattended processes in the Workstation version of Gentran:Server.

nested looping structure A mapping structure that has a loop within another loop.

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

one-to-one mapping relationship A mapping relationship without a looping structure.

outbound mapping	The process of creating a map that translates your application file format to EDI standard formats, so you can send documents to your trading partners.

Note
If you have the XML translation option, you can create a map that translates your application file to an XML format to send to your trading partners.

outbound translation	A translation in which the input format of your source document is either an application format or an XML format that you defined for your internal application documents.

Note
Outbound translation using an XML format is available only if your company purchased the XML translation option.

partner key	A set of values that the translator uses to find the partner relationship.
positional file	See application file.
post-session rule	An extended rule that the system processes after it processes the translation object.
predefined	On a data entry translation object, a default value for a particular field.
pre-session rule	An extended rule that the system processes before it processes the translation object.
promote	To extract one iteration (instance) of a group or repeating record from a loop.
qualifier	A field that has a value expressed as a code that gives a specific meaning to the function of another field. A qualifying relationship is the interaction between a field and its qualifier. The function of the field changes depending on which code the qualifier contains.

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

reconciliation   The process of viewing functional acknowledgments, viewing data with certain statuses, such as Error, Late, or Waiting, and correcting the EDI data.

Reconciliation ID record   A record containing a set of interchange and group IDs used with functional acknowledgments.

record   A map item that contains a group of related fields. A record can occur once or can repeat multiple times.

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

scope   The range of an extended rule. The scope determines which variables in the rule are accessible from other extended rules. The scope depends on the current state of the map.

script file   A file that contains command-line commands and UNIX commands.

segment   A group of related elements or composite data elements that combine to communicate useful data. Segments are defined by the EDI standards. A segment can occur once or can repeat multiple times. For example, a catalog price segment might consist of elements for item description, volume, price, lead time, etc. By themselves, none of these elements would communicate useful information. Together, they provide the information necessary for someone to tell if the item is what is desired, whether the price is reasonable, etc. A number of segments together form a transaction set.

select standard rule	A mapping function that enables you to map definition and table information from the Gentran:Server Partner Editor to a selected element or field in a map. This standard rule enables you to select entries from the parent group on the current side of the map. 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.
simple mapping	See *link*.
split (map component	A function used to split (break) a group or repeating record into two loops. You typically use this function when you need more than one instance of the same map component that still occurs multiple times.
split (XML file)	A configuration process that enables the system to break an XML file into smaller documents or sets.
Standard Cross-Reference Table	The Gentran:Server feature that enables you to build a table of standard values. Gentran:Server can use these values to find the Trading Partner record for certain inbound documents.
standard rules	An Application Integration feature that enables you to apply basic or standard mapping operations to fields. These rules give you access to 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 audit directory	The EDI Status Audit directory (*stataud*) used to store temporary audit files (*edistat.i* and *edistat.o*) and EDI data.
status record	A temporary audit file named *edistat.i* or *edistat.o*, created during translation, which includes the directory path and file name for the EDI data received or generated. The record does not contain the actual EDI data.

string	A type of field or element that contains one or more printable characters. If you specify that a field is a string type, you must designate the format by specifying a syntax token.
sub-element	See *component data element*.
syntax token	A symbol or expression that defines ranges of characters and numbers that are allowed to be used for a string-type field. You use the syntax tokens in the Format field of the Field Properties dialog to define what type of characters the system uses while compliance checking each field.
system variable standard rule	A function that enables you to set a variable that maps the current date and time to the selected field.
TRADACOMS record	A record containing all of the supplementary Trading Partnership information for use with TRADACOMS standards.
TDF	Transaction Data File. A flat file representation of the EDI standard document. This file serves as a filter between your document files and the Gentran:Server translator. For outbound processing, data is imported from a TDF file and translated to EDI format using a TDF Import translation object. The data is then ready to be posted and sent to your trading partner. For inbound processing, EDI documents that are received by Gentran:Server can be exported to a TDF-formatted file using a TDF Export translation object. This data file is then ready to be processed or converted to your internal application files.
trading partner	Another firm with which your company trades documents.
Trading Partner Administration	The Gentran:Server subsystem that enables you to define, edit, and delete all partner information for your company and all of your trading partners.

Trading Partner record	One of the records maintained in trading partner files:

- ◗ Interchange Organization record
- ◗ Group Organization record
- ◗ Trading Partnership record
- ◗ Contact record
- ◗ Reconciliation ID record
- ◗ Category record
- ◗ TRADACOMS record.

Trading Partnership	An arrangement with a specific trading partner to exchange information in a specific document type and using a particular standard version.
Trading Partnership code	A user-defined code that uniquely identifies a Trading Partnership record. Normally, the Trading Partnership record describes a unique trading partner and document combination. The Trading Partnership Code must contain from 1-15 alphanumeric characters. Spaces and special characters are NOT allowed.
Trading Partnership record	A record containing information about one of the Trading Partnerships you have established.
transaction set	A business form as defined by the standards. 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. Examples include an ANSI 850 purchase order or an UCS 880 invoice. Transaction sets are also known as messages and documents.
translation	The conversion of data from one format to another.
translation object	A compiled map file containing information that instructs the translator how to convert a file from one format to another.

translator	The Gentran:Server subsystem that translates data from one format to another.
update standard rule	A standard rule that enables you to update a specific field on the current side of the map with the contents of a field.
use accumulator standard rule	A standard rule that gives you access to a set of numeric variables that you can manipulate via numeric operations, and then transfer to and from fields. Accumulators are used generally for counting the occurrences of a specific field or generating increasing or sequential record or line item numbers.
use code standard rule	A standard rule that enables you to match a field against a predefined code table, specify whether or not a compliance error is generated if the field does or does not contain one of the values in the code table, and store a code's description in another field.
user exit	An extended rule that enables the map to temporarily exit translation, run a shell script, return to translation, and use the results of the shell script in translation.
user translation object	A translation object that controls data entry, Application-to-EDI processing, EDI-to-Application processing, EDI-to-EDI processing, and the creation of printed reports. The Application Integration subsystem enables you to generate user translation objects.
variable	An item that you define in the declarations section of an extended rule to store values. Variables consist of a name and a data type.
workspace	The system's working area for an import file.

Index