Gentran:Director ®

for Windows ®

Version 5.4

Application Integration TRADACOMS Tutorial Guide

Sterling Commerce
An IBM Company

4109-540-TUTR03-0000

April 2005

Contents

CHAPTER

1

Getting Started

Welcome to Gentran:Director® for Windows® Application Integration subsystem, Sterling Commerce’s Electronic Data Interchange (EDI) mapping software for the Windows operating system. Gentran:Director provides you with the tools you need to translate your application files to EDI standard formats for documents you send to your partners (outbound mapping), and to translate EDI standard formats to your application format for documents that you receive from your partners (inbound mapping).

What’s in this Manual

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

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

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

- **The terminology used to describe mouse and cursor actions.**
- **The terminology used to describe Windows-specific attributes, including dialogue boxes, icons, windows, and buttons.**

This manual also assumes that you are familiar with EDI concepts, EDI standards structure, your internal application format, data mapping concepts, and Gentran:Director.

The *Tutorial Guide* is organised into chapters. A brief description of the chapter contents follows.

- *Chapter 1, Getting Started* explains the content, organization, conventions, and system requirements in this guide. This chapter briefly describes how to build maps using Gentran:Director Application Integration and how to customize global display options. It also introduces mapping and Gentran:Director for Windows.
- *Chapter 2, Inbound Purchase Order (ORDERS)* uses a step-by-step approach to explain the tasks and procedures necessary to create an export map for a TRADACOMS inbound (ORDERS) purchase order.
- *Chapter 3, Outbound Invoice (INVOIC)* uses a step-by-step approach to explain the tasks and procedures necessary to create a system import map and an import map for a TRADACOMS outbound (INVOIC) invoice.
- The *Glossary* contains definitions of mapping and Application Integration subsystem terms and concepts.

Conventions in this Manual

The different typefaces used in this manual are as follows:

Creating a new map This typeface is used for emphasis or to indicate a chapter or section in this manual or other Gentran:Director manuals.

NEW... This typeface indicates a button in the program. Click the button with your mouse, or press **TAB** until the button is highlighted, then press **ENTER** to execute that command.

F1 This typeface indicates a key that you need to press on your keyboard. A plus sign (+) between key names indicates a combination of keys to be pressed together (e.g., **ALT** + **TAB**).

The different icons used in this manual are as follows:

ℹ **This icon indicates a note that gives you additional information on the current subject.**

STOP **This icon indicates a caution that alerts you to the potential loss of information. Please read any text preceded by this icon carefully.**

💡 **This icon indicates helpful tips, suggestions, shortcuts, and alternative actions.**

Introducing Mapping

A map is a set of instructions that you define in the Gentran:Director Application Integration subsystem to indicate the relationship between your application file and the EDI standards. A map also defines how the system translates data.

Inbound

For inbound mapping, the system translates your trading partner’s EDI standard formatted business document to your application file format. To translate inbound data, you must create an export map in the Gentran:Director Application Integration subsystem because the system is *exporting to* your application file.

Outbound

For outbound mapping, the system translates your application file format to EDI standard formats, so you can send documents to your partners. To translate outbound data, must to create an import map and a system import map. You must create an import map because the system is *importing from* your application file. The import map defines how to move data from your application file (flat file definition) to the EDI standard-formatted documents that your partner expects to receive from you. When an application file contains multiple documents, you must create a system import map to specify which trading relationship (established in Partner Editor) corresponds to which document in the application file so the system knows the import map to use to process the document.

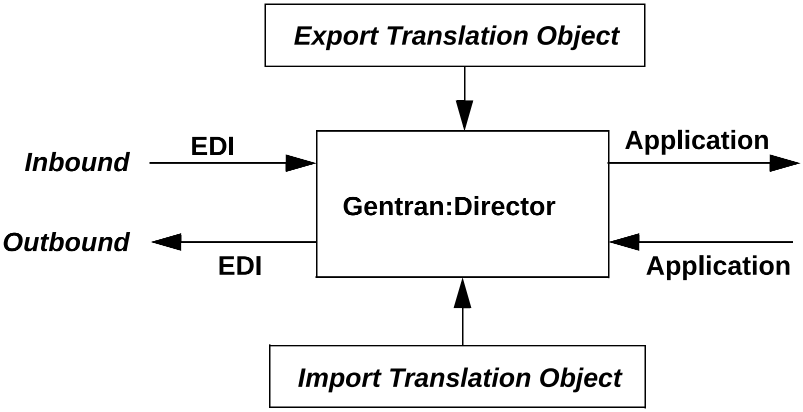
Translation Objects

The map you create is the *source* from which you compile a translation object. A translation object is a set of rules that instructs the translator how to convert a file from one format to another. Each function performed during translation requires a translation object. Translation objects control all Gentran:Director EDI translation functions. Translation objects can be either system translation objects or user translation objects. System translation objects control the creation and separation of interchanges, functional groups, and transaction sets. They are also used to generate and reconcile functional acknowledgements. User translation objects control data entry, importing, exporting, document turnaround, and creating printed reports. All required system translation objects are automatically installed with Gentran:Director.

Gentran:Director Application Integration allows you to generate import, export, and document turnaround translation objects. The Gentran:Director Forms Integration subsystem enables you to generate data entry and print translation objects. Once you generate a translation object using the Gentran:Director Application Integration subsystem (or the

Gentran:Director Forms Integration subsystem), you must register the translation object with Gentran:Director (see the Gentran:Director *User's Guide* for more information). In addition, you must establish correct trading relationships for all trading partners. To establish a trading relationship, you must specify the appropriate translation object name that the translator uses to translate the data. (See the Gentran:Director *User's Guide* for more information.)

The diagram below illustrates the inbound and outbound translation process:



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

Application File Format

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

The map components that you define your application file are:

- A **group** is a looping structure that contains related records and/or groups that repeat in sequence until either the group data ends or the maximum number of times that the loop is allowed to repeat is exhausted. If you create a group that is subordinate to another group (a sub-group), this corresponds to a nested looping structure (a loop within a loop). The application (positional) file is a group and is visually represented the same way as other groups and sub-groups in the Application Integration subsystem: ❒

- A **record** contains a group of related fields. A record occurs once or repeats multiple times. A record is visually represented in the Application Integration subsystems as: ●
- A **field** is the smallest piece of information defined in the application file. A field is the application map component that is mapped (linked) to a corresponding EDI element. It is visually represented in the Application Integration subsystems as: 🟩

  After a field has a mapping operation performed against it, a red checkmark is displayed over the field icon.

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

EDI File Format

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

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

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

Map Components

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

The map components that Gentran:Director uses to define the EDI file format are:

- A **group** is a looping structure that contains related segments and/or groups that repeat in sequence until either the group data ends or the maximum number of times that the loop is allowed to repeat is exhausted. Groups are defined by the EDI standards. A group that is subordinate to another group is a sub-group (this corresponds to a nested looping structure – a loop within a loop). The EDI file is a group and is visually represented the same way as other groups and sub-groups in the Application Integration subsystem: ❐

- A **segment** contains a group of related elements or composite data elements that combine to communicate useful data. Segments are defined by the EDI standards. A segment occurs once or repeats multiple times. A segment is visually represented in the Application Integration subsystems as: ◉

- A **composite** is 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). A composite is visually represented in the Application Integration subsystems as: ▦

- An **element** is the smallest piece of information defined by the EDI standards. An individual element has different meanings depending on the context. Elements are typically not considered useful until combined into segments. An element is the EDI map component that is mapped (linked) to a corresponding application field to move data to and from the EDI file. An element is visually represented in the Application Integration subsystem as the following: 🟩

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

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

Introducing the Application Integration Subsystem

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

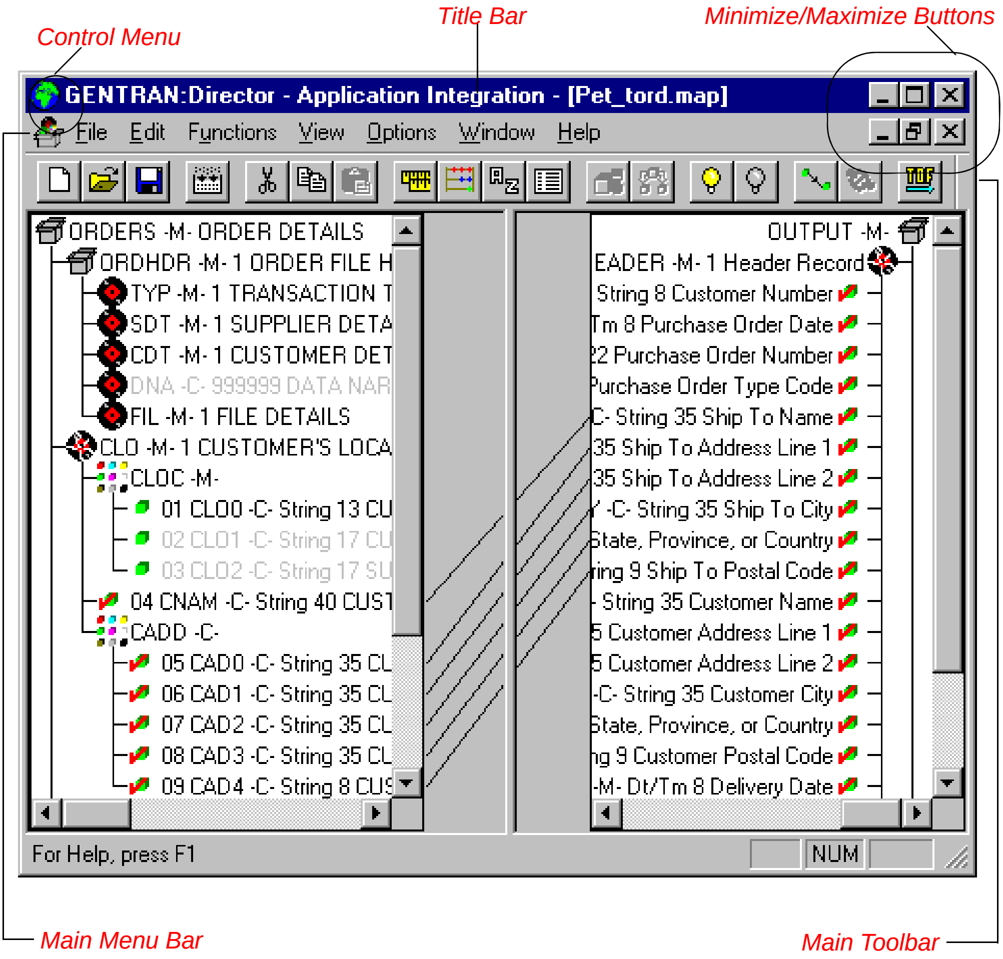
Navigation

You can access the Gentran:Director Application Integration Window in one of four different ways:

- Click the menu option from the Main Menu Bar.
- Click the appropriate button on the Main Toolbar.
- Click the appropriate part of the map.
- Click a map component with the right mouse button to access a short-cut menu, which contains all the functions that are currently available for that map component. The content of the short-cut menus varies, depending on the type and level of the selected map component.

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

The Gentran:Director - Application Integration Window appears as follows:


Control Menu
Title Bar
Minimize/Maximize Buttons
GENTRAN:Director - Application Integration - [Pet_tord.map]
File Edit Functions View Options Window Help
Main Menu Bar
Main Toolbar
For Help, press F1
NUM

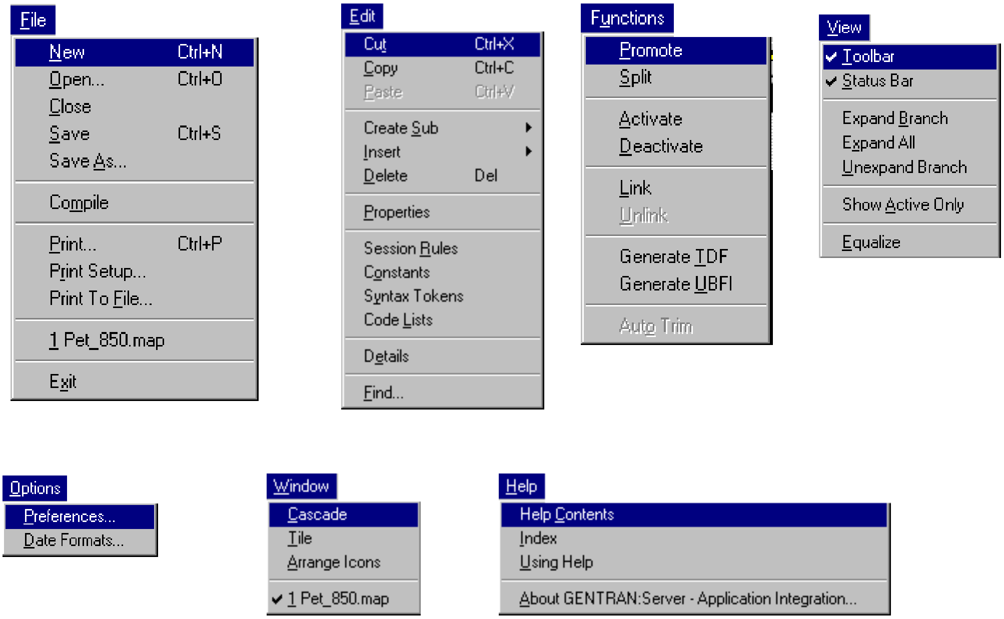
```
ORDERS -M- ORDER DETAILS
  ORDHDR -M- 1 ORDER FILE H
    TYP -M- 1 TRANSACTION T
    SDT -M- 1 SUPPLIER DETA
    CDT -M- 1 CUSTOMER DET
    DNA -C- 999999 DATA NAR
    FIL -M- 1 FILE DETAILS
  CLO -M- 1 CUSTOMER'S LOCA
    CLOC -M-
      01 CLO0 -C- String 13 CU
      02 CLO1 -C- String 17 CU
      03 CLO2 -C- String 17 SU
    04 CNAM -C- String 40 CUST
    CADD -C-
      05 CAD0 -C- String 35 CL
      06 CAD1 -C- String 35 CL
      07 CAD2 -C- String 35 CL
      08 CAD3 -C- String 35 CL
      09 CAD4 -C- String 8 CUS
```

```
                          OUTPUT -M-
          EADER -M- 1 Header Record
         String 8 Customer Number
         Tm 8 Purchase Order Date
        2 Purchase Order Number
        urchase Order Type Code
        C- String 35 Ship To Name
        35 Ship To Address Line 1
        35 Ship To Address Line 2
        / -C- String 35 Ship To City
        State, Province, or Country
        ing 9 Ship To Postal Code
        - String 35 Customer Name
        5 Customer Address Line 1
        5 Customer Address Line 2
        -C- String 35 Customer City
        State, Province, or Country
        ng 9 Customer Postal Code
        -M- Dt/Tm 8 Delivery Date
```

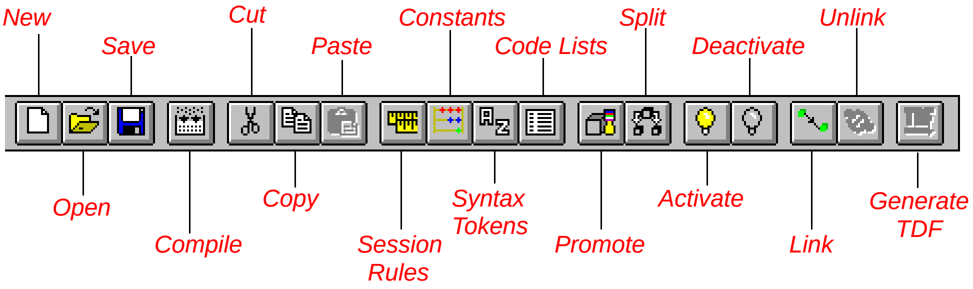
- The **Control Menu** contains the following commands: **Restore**, **Move**, **Size**, **Minimize**, **Maximize**, and **Close**.
- The **Title Bar** contains the name of the application. You can use this to move the position of the EDI Director window on your desktop.
- The **Minimize** button reduces the window to an icon.

  The **Maximize** button enlarges the window to its maximum size.

- The **Main Menu Bar** contains the following drop-down menus (detailed descriptions of each of the commands on these menus are found in the online Help). Unavailable items are dimmed.



- The **Main Toolbar** gives you access to some of the most common operations in the Application Integration subsystem. Unavailable items are dimmed. The Main Toolbar is a dockable toolbar, so you can affix it to any edge of the client window. The Main Toolbar is illustrated below:



Customizing Global Display Options

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

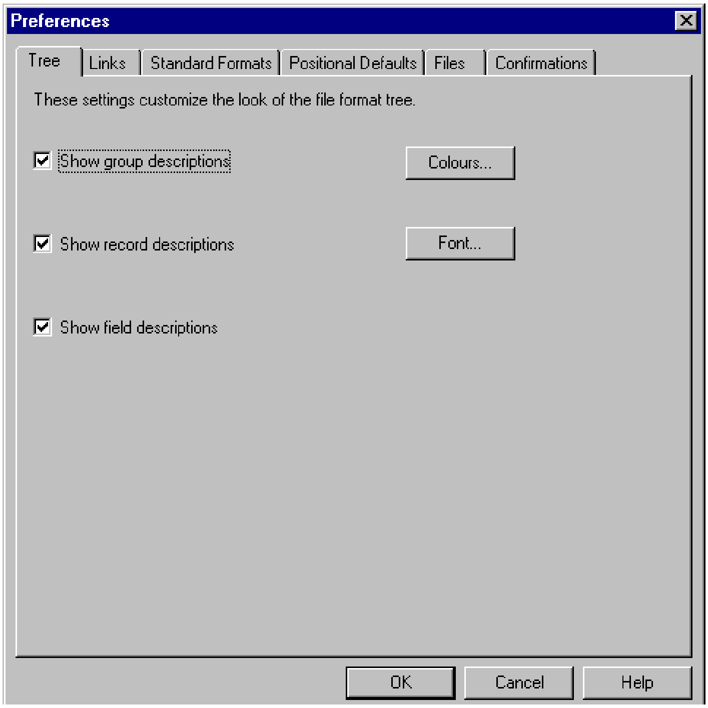
- Customizing global display options
- Displaying links (global option)

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

Customizing Global Display Options

Complete the following steps to customize global (for all maps) display options:

1. Select Preferences... from the Options menu to display the Preferences dialogue (the Tree tab is displayed by default). The Preferences dialogue is a property sheet that enables you to set global defaults for Gentran:Director.



These settings customize the look of the file format tree.

☑ Show group descriptions

☑ Show record descriptions

☑ Show field descriptions

2. To turn on the default display of group, record (segment), and field (element) descriptions, select the appropriate options.

   You typically want all descriptions displayed for reference. But depending on the size of your monitor, it may be easier to see the entire map if the descriptions are not displayed.

You may want to experiment with shrinking the size of the font for the map (see *Customizing Global Fonts* in the Gentran:Director *Application Integration User's Guide*) before you turn off the display of descriptions.

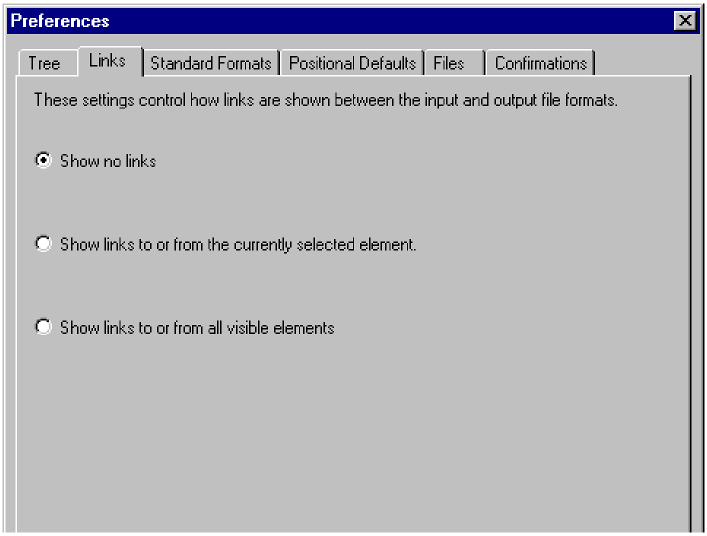
3. Click OK to save changes and exit the Preferences dialog.

Customizing Global Display of Links

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

Complete the following steps to customize the global display of mapping links:

1. Select Preferences... from the Options menu to display the Preferences dialog.
2. Click the Links tab to display the linking options. The Links tab enables you to set the default display of mapping links.



These settings control how links are shown between the input and output file formats.

● Show no links

○ Show links to or from the currently selected element.

○ Show links to or from all visible elements

The options in the Links section of the Preferences dialog enable you to select:

- **Show no links**
  Do not visually display mapping links.
- **Show links to or from the currently selected element**
  Display only the mapping links for the currently selected field (this option enables you to concentrate on the selected field and removes the confusion of viewing many links at once).
- **Show links to or from all visible elements**
  Display all the mapping links.

The option you select is the default for all maps.

3. Click `OK` to save changes and exit the Preferences dialog.

Building a Map with Application Integration

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

1. **Preparation and Analysis**
   Obtain a layout of your application file and determine how it corresponds with the EDI standard you will be using. Determine how you will move data to or from each application field.

2. **Set Global Defaults (first time only)**
   The first time you use the Gentran:Director Application Integration subsystem, you should establish the default date format that the system will use. Please see the Gentran:Director *User's Guide* for more information on setting the global date default.

3. **Create, Save, and Name a New Map**
   Please see *Creating the Export Map* on page 28 for more information on creating, saving, and naming a map.

4. **Activate the Appropriate EDI Groups, Segments, and Elements**
   Please see *Activating EDI Map Components* on page 35 for more information on activation.

5. **Define your Application File Format**
   Please see *Defining Your Application File Format* on page 40 for more information on defining your application file format.

6. **Map the Appropriate Data for Each Application Field**
   Please see *Mapping Your Data* on page 62 for more information on mapping your data.

7. **Compile the Translation Object**
   Please see *Compiling the Map* on page 93 for more information on compiling the translation object and translation object naming conventions.

8. **Print the Mapping Report**
   Validate and review the map, and make modifications as needed. Please see *Printing the Report* on page 95 for more information on printing the mapping report.

9. **Register the Translation Object with Gentran:Director**
   Please see the Gentran:Director *User's Guide* for more information on registering translation objects.

10. **Create the Appropriate Trading Relationship**
    Establish the appropriate trading relationship in Gentran:Director for your trading partners. Please see the Gentran:Director *User's Guide* for more information on creating trading relationships. The trading relationship that you will use in this tutorial (PETZONE3) is provided in the TUTORIAL folder. You need to import the partner profile (PETZONE3.PAR) into Gentran:Director before you begin this tutorial. Please see the Gentran:Director *User's Guide* for more information on importing trading relationships.

11. **Test the Translation Object**
    Obtain test data from your partner(s) and process the data. Verify acknowledgement processing (if applicable). Verify communications with your network.

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

Translation Object Type	Test Data Name
Inbound (export)	PETTSTT1.INT
Outbound (system import and import)	PET_TINV.TXT

CHAPTER

2

Inbound Purchase Order (ORDERS)

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

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

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

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

Getting Started

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

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

Inbound Mapping Scenario

In this tutorial exercise, your company is MWT Manufacturing Co., which manufactures a wide variety of supplies for pet stores. Your company sells pet supplies directly to large pet supply retail chains. You need to create a map that will enable your company to translate the purchase orders that you receive from one of your trading partners, Pet Zone (a large pet supply retailer). After the purchase orders are translated into your application file format, they can be processed through your order processing system, and your company can ship the goods to Pet Zone.

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

Mapping Analysis

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

Regardless of whether you are translating data inbound or outbound, mapping analysis involves the following steps:

1. Analysing your application file format
2. Analysing your partner(s) EDI file format
3. Correlating your application and the EDI file formats

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

Analysing Your Application File Format

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

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

The application file layout for this tutorial is as follows:

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (POHEADER) 1 per message					
CUSTNUMHDR	X	Customer Number	String	1	8
PODATEHDR	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMHDR	X	Purchase Order Number	String	1	22
POTYPCODE		Purchase Order Type Code	String	0	2
SHIPTONAME		Ship To Name	String	0	35
SHIPTOADD1		Ship To Address Line 1	String	0	35
SHIPTOADD2		Ship To Address Line 2	String	0	35
SHIPTOCITY		Ship To City	String	0	35
SHIPTOSTATE		Ship To State, Province, or Country	String	0	35
SHIPTOPOSTALCODE		Ship To Postal Code	String	0	9
CUSTNAME		Customer Name	String	0	35
CUSTADD1		Customer Address Line 1	String	0	35
CUSTADD2		Customer Address Line 2	String	0	35
CUSTCITY		Customer City	String	0	35
CUSTSTATE		Customer State, Province, or Country	String	0	35
CUSTPOSTAL CODE		Customer Postal Code	String	0	9

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
DELDATE	X	Delivery Date	Dt/Tm (DD/MM/YY)	8	8
NUMLINEITEMS	X	Number of Line Items	Number (N0)	1	6
PO Detail Group (PODetailGroup) up to 200,000 per message					
Detail Record (PODETAIL) 1 per message					
CUSTNUMDET	X	Customer Number	String	1	8
PODATEDET	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMDET	X	Purchase Order Number	String	1	22
CUSTPROCODE	X	Customer Product Code	String	1	30
UPCCODE	X	UPC Code	String	1	30
CONSUMER UNITS		Consumer Units in Trading Unit	Number (N0)	0	15
UOM		Unit of Measure	String	0	7
UNITPR		Unit Price	Number (R2)	0	14
QTYORD	X	Quantity Ordered	Number (N0)	1	9
Description Record (PODESC) up to 5 per message					
CUSTNUMDESC	X	Customer Number	String	1	8
PODATEDESC	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMDESC	X	Purchase Order Number	String	1	22
DESC		Description	String	0	81
Summary Record (POSUMMARY) 1 per message					
CUSTNUMSUM	X	Customer Number	String	1	8
PODATESUM	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMSUM	X	Purchase Order Number	String	1	22
NUMLINEITEMSUM	X	Number of Line Items	Number (N0)	1	6

The above table contains the minimum information necessary to define your application file format to the Application Integration subsystem (name, mandatory status, description, data type, and field length). Each record contains a group of logically-related application fields. These fields will contain the information from the EDI file that your system needs to process the purchase order. The group contains logically-related application records.

Each record contains a key (consisting of three fields) that ensures that the same identifying information is located at the beginning of each record. Your application contains a record key so that each record can be identified individually. The record key for this purchase order consists of your partner’s customer number, the purchase order date, and the purchase order number. The records and the group are defined as follows:

- The POHEADER (header) record contains the purchase order type code, shipping and customer information, delivery date information, and the number of line items in the purchase order. This header record is mandatory and occurs once for each purchase order translated.
- The PODetailGroup (detail information) contains all the line item information for the order, including the item, unit of measure, price, quantity ordered, and the description information. Each PODetailGroup can occur up to 200,000 times per order translated. This equals a maximum order of 200,000 items. The information within the PODetailGroup is further segregated by record.
- The PODETAIL (detail) record contains line item code number, consumer units in trading unit, unit of measure, price, and quantity ordered. The PODETAIL record occurs once per PODetailGroup.
- The PODESC (description) record contains descriptive information about the line item ordered. The PODESC record can occur up to five times per PODetailGroup, so multiple descriptive terms can be used for each item. For example, if the line item is a “dog collar,” the description fields may be “red,” “large,” and “leather.”
- The POSUMMARY (summary) record contains the total number of line items in the purchase order. The POSUMMARY record occurs once for each purchase order translated.

Now that you have thoroughly analysed your application file format, the next step is to determine the requirements of the purchase order that you will receive from your partner(s).

Analysing the EDI File

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

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

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

After discussing with your partner(s) what they will send, determining which segments and elements your company requires, and reviewing the EDI standard, you can list the map components that you will need to make available for use (activate). When Gentran:Director generates the EDI side(s) of the map, the system includes all the groups, segments, composites, and elements that are defined by the standard agency for the version of the message you selected. The system activates all the groups, segments, composites, and elements that are defined as “mandatory” (must be present) by the standard. The system does not enable you to deactivate the mandatory groups, segments, composites, and elements.

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

For this tutorial, the map components that are required but not currently active are the following (the composite column is informational only, so you will know where to locate the appropriate element):

Group	Segment	Comp.	Element
ORDHDR (already active)	SDT (already active)	SIDN	SID0
	CDT (already active)	CIDN	CID0
			CNAM
		CADD	CAD0 CAD1 CAD2 CAD3 CAD4

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

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

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

Also, we recommend that you set up a Document Name for each map to make the identification of a message created by this map easier in Gentran:Director. The Document Name allows you to differentiate between messages in the document browsers in Gentran:Director.

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

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

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

Reconciling Your Application File and the EDI File

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

Linking (simple mapping) enables you to map a field or element from the input side of the map to a field or element on the output side of the map. The link between two map components (hereafter referred to as fields) is visually represented with a line connecting the two fields.

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

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

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

The following mapping operations (by application field) will be required for this tutorial:

Field Name	Description	Mapping Operations
Header Record (POHEADER) 1 per message		
CUSTNUMHDR	Customer Number	Extract Application ID from Partner record
PODATEHDR	Purchase Order Date	Link to ORD ORNO ORN2
PONUMHDR	Purchase Order Number	Link to ORD ORNO ORN0
POTYPCODE	Purchase Order Type Code	Link to ORD CD07
SHIPTONAME	Ship To Name	Link to CLO CNAM

Field Name	Description	Mapping Operations
SHIPTOADD1	Ship To Address Line 1	Link to CLO CADD CAD0
SHIPTOADD2	Ship To Address Line 2	Link to CLO CADD CAD1
SHIPTOCITY	Ship To City	Link to CLO CADD CAD2
SHIPTOSTATE	Ship To State, Province, or Country	Link to CLO CADD CAD3
SHIPTOPOSTALCODE	Ship To Postal Code	Link to CLO CADD CAD4
CUSTNAME	Customer Name	Link to CDT CNAM
CUSTADD1	Customer Address Line 1	Link to CDT CADD CAD0
CUSTADD2	Customer Address Line 2	Link to CDT CADD CAD1
CUSTCITY	Customer City	Link to CDT CADD CAD2
CUSTSTATE	Customer State, Province, or Country	Link to CDT CADD CAD3
CUSTPOSTALCODE	Customer Postal Code	Link to CDT CADD CAD4
DELDATE	Delivery Date	Link to DIN LDAT
NUMLINEITEMS	Number of Line Items	Link to OTR LORD
PO Detail Group (PODetailGroup) up to 200,000 per message		
Detail Record (PODETAIL) 1 per message		
CUSTNUMDET	Customer Number	Extract Application ID from Partner record
PODATEDET	Purchase Order Date	Link to ORD ORNO ORN2 using extended rule
PONUMDET	Purchase Order Number	Link to ORD ORNO ORN0 using extended rule
CUSTPROCODE	Customer Product Code	Link to OLD:2 CPRO CPR1
UPCCODE	UPC Code	Extract Partner xref by partner item from subtable “PRODCODE”
CONSUMERUNITS	Consumer Units in Trading Unit	Link to OLD:2 UNOR UNO0
UOM	Unit of Measure	Link to OLD:2 UNOR CD04
UNITPR	Unit Price	Link to OLD:2 OUCT OUC0

Field Name	Description	Mapping Operations
QTYORD	Quantity Ordered	Link to OLD:2 OQTY OQT0
Description Record (PODESC) up to 5 per message		
CUSTNUMDESC	Customer Number	Extract Application ID from Partner record
PODATEDESC	Purchase Order Date	Link to ORD ORNO ORN2 using extended rule
PONUMDESC	Purchase Order Number	Link to ORD ORNO ORN0 using extended rule
DESC	Description	Link to OLD:2 TDES TDE0 and OLD:2 TDES TDE1 after concatenating them with an extended rule
Summary Record (POSUMMARY) 1 per message		
CUSTNUMSUM	Customer Number	Extract Application ID from Partner record
PODATESUM	Purchase Order Date	Link to ORD ORNO ORN2
PONUMSUM	Purchase Order Number	Link to ORD ORNO ORN0
NUMLINEITEMSUM	Number of Line Items	Link to OTR LORD

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

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

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

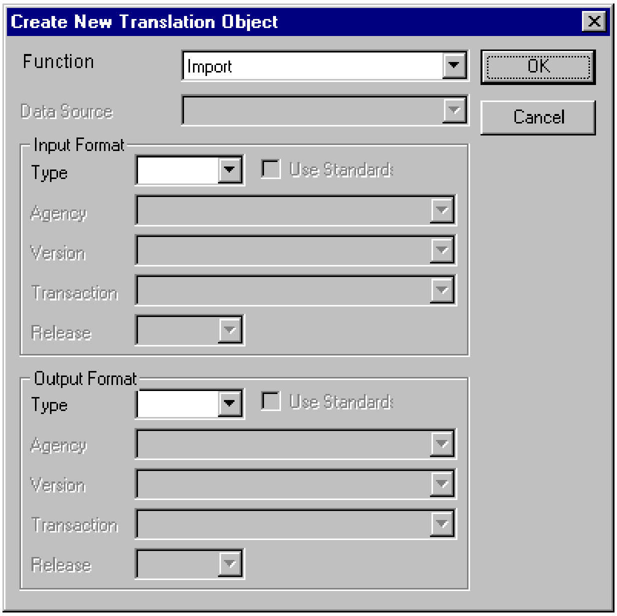
After you finish correlating your application file format and EDI file on paper, you can begin creating the map.

Creating the Export Map

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

Complete the following steps to create the export map for the Inbound Purchase Order:

1. Select New from the File menu to display the Create New Translation Object dialogue.



Create New Translation Object

Function: Import

Data Source

Input Format
Type
□ Use Standards
Agency
Version
Transaction
Release

Output Format
Type
□ Use Standards
Agency
Version
Transaction
Release

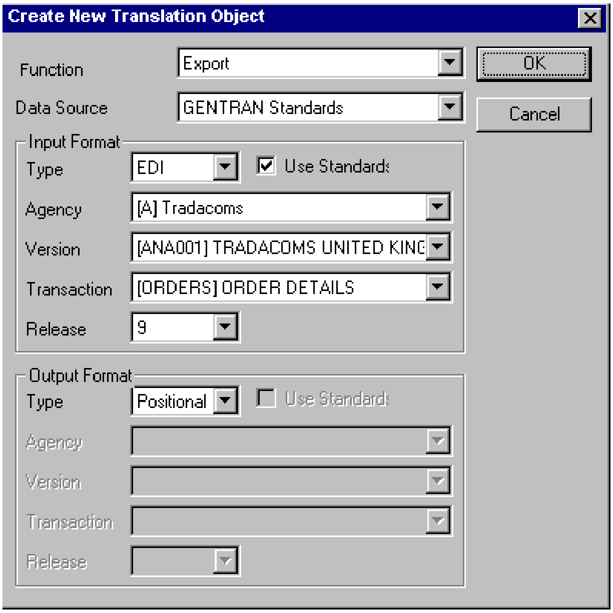
2. From the Translation Object Function drop-down list box, select “Export”.

   Complete the Input Format section of the dialogue. This section contains the format of the data that will be translated by the Gentran:Director system.

3. From the (Input Format) Type drop-down list box, select “EDI” as the type of format for the input data.
4. Click the Use Standards check box, because you selected “EDI” in the Type field and because you want the system to use the standards database on your system to build the purchase order according to the standard.
5. To select a data source name from which Application Integration will access EDI standards, select one from the Data Source drop-down list box. This DSN will be prompted the next time you create a map.
6. From the Agency drop-down list box, select “[A] Tradacoms” as the EDI standard you will be using for the input side of the map.
7. From the Version drop-down list box, select “[ANA001] TRADACOMS UNITED KINGDOM” as the version of the standard you will use. The versions that are available depend on which standard you selected.
8. From the Transaction drop-down list box, select “[ORDERS] ORDER DETAILS” as the transaction set (message) you will use. The transactions that are available depend on which standard and version you selected.
9. From the Release drop-down list box, select “9” as the message version you will use. The message versions that are available depend on which message (transaction) you selected.

   Now you need to complete the Output Format section of the dialogue.
10. From the Type drop-down list box, select “Positional” as the type of format for the output data. Gentran:Director uses the term “positional” to describe any flat file that you create.

The Create New Translation Object dialogue should now look like this:



Create New Translation Object

Function: Export

Data Source: GENTRAN Standards

Input Format

Type: EDI ☑ Use Standards

Agency: [A] Tradacoms

Version: [ANA001] TRADACOMS UNITED KING

Transaction: [ORDERS] ORDER DETAILS

Release: 9

Output Format

Type: Positional □ Use Standards

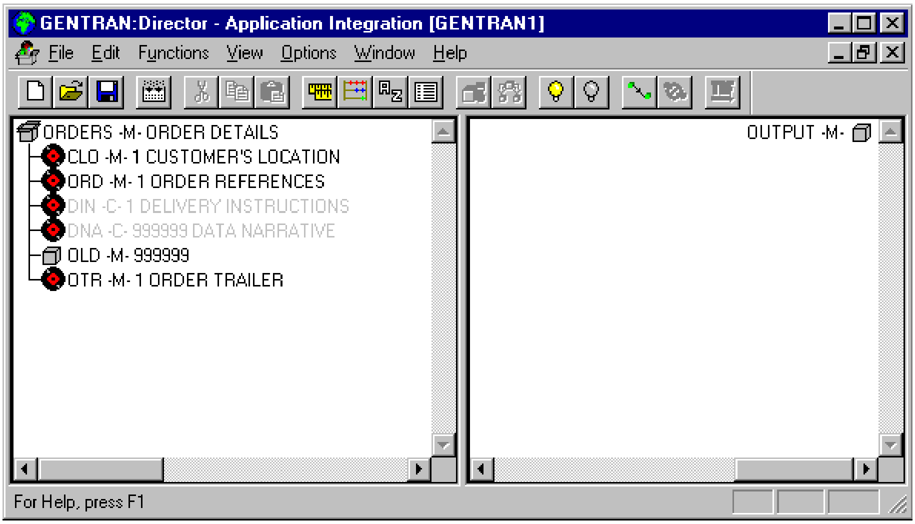
Agency:

Version:

Transaction:

Release:

11. Click OK to generate the new map (the generation may take a few seconds). The new map is displayed in the Gentran:Director - Application Integration Window.



GENTRAN:Director - Application Integration [GENTRAN1]

ORDERS -M- ORDER DETAILS
CLO -M- 1 CUSTOMER'S LOCATION
ORD -M- 1 ORDER REFERENCES
DIN -C- 1 DELIVERY INSTRUCTIONS
DNA -C- 999999 DATA NARRATIVE
OLD -M- 999999
OTR -M- 1 ORDER TRAILER

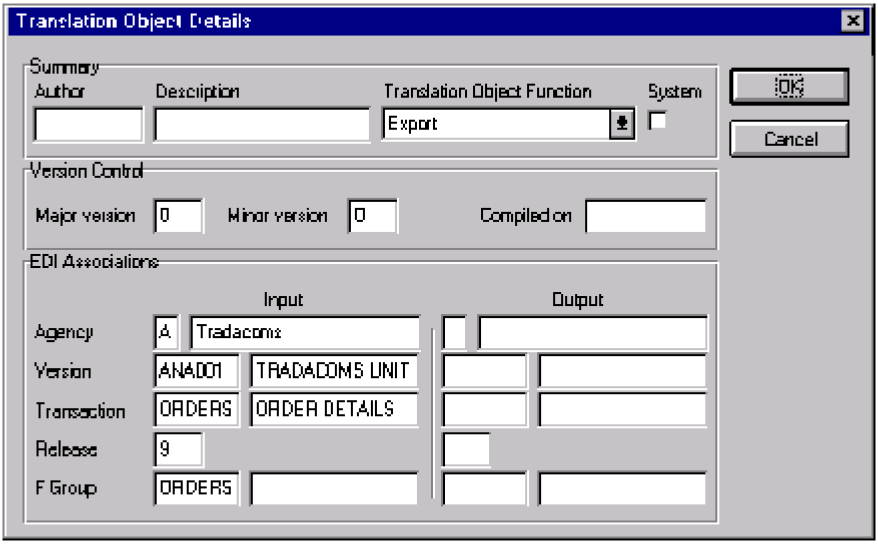
OUTPUT -M-

For Help, press F1

After you create a new map, it is important to define the details of the map and save it. The Save function writes (saves) the map with current focus to disk.

We recommend that you save your maps as often as possible, to mitigate the possible loss of additions or modifications in the event of a hardware or system error.

12. Select Save from the File menu to display the Translation Object Details dialogue.



Author:

Description:

Translation Object Function: Export

System: □

Major version: 0

Minor version: 0

Compiled on:

	Input		Output	
Agency	A	Tradacoms		
Version	ANA001	TRADACOMS UNIT		
Transaction	ORDERS	ORDER DETAILS		
Release	9			
F Group	ORDERS			

13. In the Author field, type your name. This field typically contains the name of the person who created the map.

14. In the Description field, type "PET T ANA001 9 ORDERS Export". This field must be unique because it is used by the system to identify the map. We recommend that you use the following identifying characteristics: the partner that this map is used for, the standard, the version, the release, the type of transaction this map uses, and the direction of the map (e.g., Import, Export, etc.). For example, "PET T ANA001 9 ORDERS Export" is the description of a map used with partner Pet Zone, for a TRADACOMS version ANA001 release 9 Purchase Order (ORDERS), that is received Inbound (Export).

15. Verify that the type of map shown in the Translation Object Function drop-down list box is "Export".

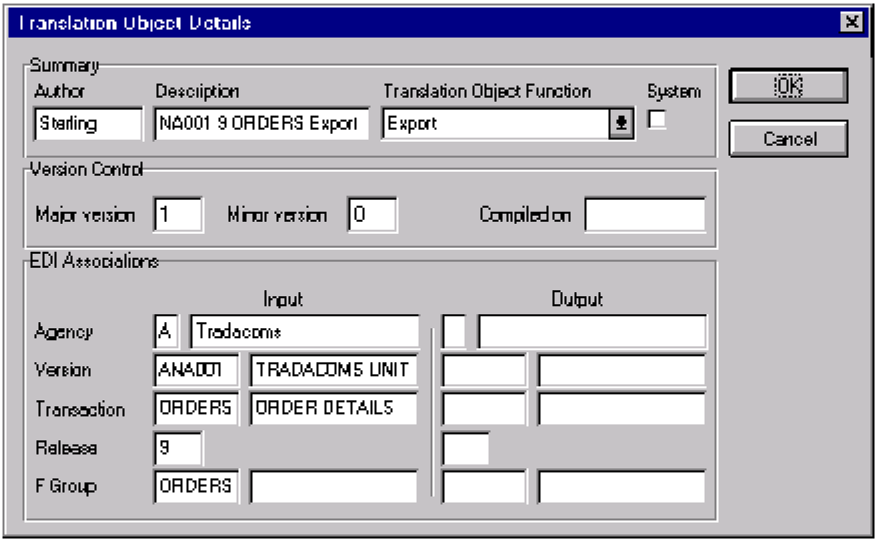
16. Complete the Major and Minor fields to designate a version number for this translation object. Type “1” as the version number in the Major field and accept “0” as the release number in the Minor field.

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

17. The Input and Output fields (Agency, Version, Transaction, Release, and F Group) contain information only if the Input and/or Output sides of the map are of EDI format. For the tutorial, DO NOT change the information in these fields.

    The system allows you to change the information in the Input and Output fields, but modifying this information will not alter the content of the map. An example of when you may wish to alter these fields is if you want the map to reflect a standard version that is not loaded on your system. You can change the version on this dialogue, and then physically alter the map to be compliant with that version.

The Translation Object Details dialogue should now look like this:



Summary

Author	Description	Translation Object Function	System
Sterling	NA001 9 ORDERS Export	Export	☐

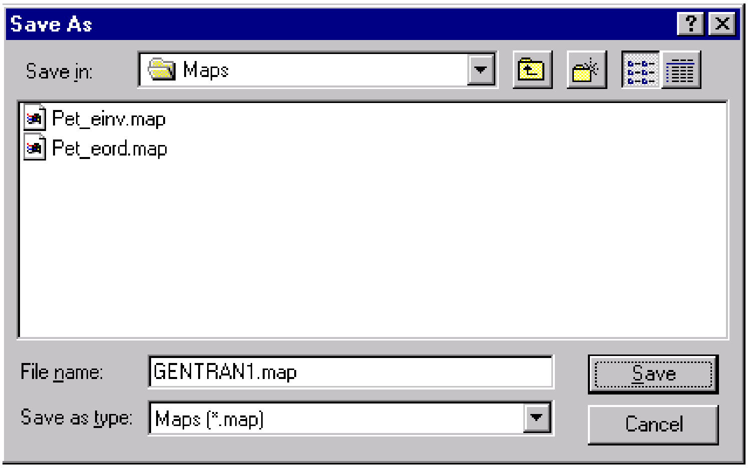
Version Control

Major version 1 Minor version 0 Compiled on

EDI Associations

	Input		Output	
Agency	A	Tradacoms		
Version	ANA001	TRADACOMS UNIT		
Transaction	ORDERS	ORDER DETAILS		
Release	9			
F Group	ORDERS			

18. Click OK to save the translation object details and access the Save As dialogue.



19. The "Save in" drop-down list box should reflect the drive\folder where Gentran:Director is installed. The default is GENTRAN Director\Maps. If the current subfolder is not the correct one, select the appropriate subfolder.

20. Type "PET_TORD.MAP" as the name of the map in the "File name" field. You allocate a filename to a map when you save the map for the first time or use the Save As function to save a copy of the map under a different name. The name of the map must be one to eight unique characters that define the map in a meaningful way. For example, "PET_TORD.MAP" is the name of a map for a TRADACOMS Purchase Order (ORDERS) for partner Pet Zone.

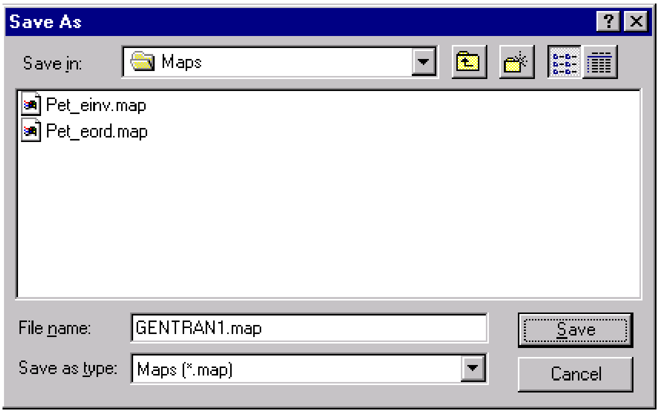
    The default file extension for maps is .MAP. Using this extension distinguishes map files from other types of files, such as translation objects. The naming conventions in Gentran:Director are as follows:

    - The source map (map created in Gentran:Director) should be named using the file extension **.MAP** (e.g., PET_TORD.MAP).
    - The compiled translation object (translation object that is the result of compiling the source map) should be named using the same filename that was allocated to the source map, with the file extension **.TPL** (e.g., PET_TORD.TPL).

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

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

The Save As dialogue should now look like this:



File name: GENTRAN1.map

Save as type: Maps (*.map)

21. Click SAVE to save the map.

After you complete each section of this tutorial or when any changes are applied to the map, you should save the map by selecting Save from the File menu.

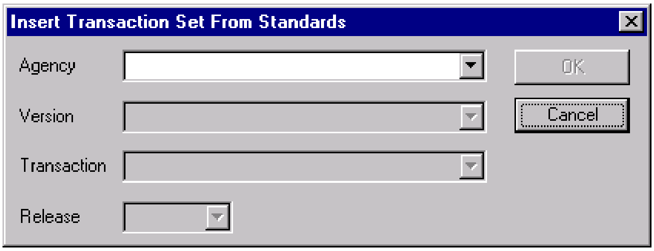
Now that you have finished creating and saving the new Export map, you need to define the Input and Output sides of the map.

Loading the ORDHDR Message

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

Complete the following steps to load the ORDHDR message into the ORDERS EDI file:

1. Highlight the ORDERS ORDER DETAILS EDI file icon. Select Create Sub from the Edit menu.
2. Select Transaction Set... from the Create Sub cascading menu to display the Insert Transaction Set From Standards dialogue. You will be prompted to select a data source. Do so and then click OK.



3. From the Agency drop-down list box, select “[A] Tradacoms” as the EDI standard you will use for the inserted message.
4. From the Version drop-down list box, select “[ANA001] TRADACOMS UNITED KINGDOM” as the version of the standard you will use. The versions that are available depend on which standard you selected.
5. From the Transaction drop-down list box, select “[ORDHDR] ORDER FILE HEADER” as the transaction set (message) you will use. The transactions that are available depend on which standard and version you selected.
6. From the Release drop-down list box, select “9” as the message version you will use. The message versions that are available depend on which message (transaction) you selected.
7. Click OK to load the ORDHDR message into the ORDERS EDI file as a subordinate group.

Activating EDI Map Components

The map components for an EDI File are the following:

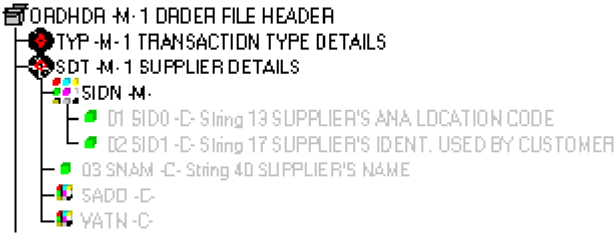
EDI File
Segment
Loop (Group)
Composite Element
Element

The map components for a Positional File are the following:

▢ Positional File
● Record
▢ Loop (Group)
🟩 Field

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

When Gentran:Director generates the EDI side(s) of the map, the system includes all the groups, segments, composites, and elements that are defined by the standard agency for the version of the message you selected. The system activates all the groups, segments, composites, and elements that are defined as "mandatory" (must be present) by the standard. The system does not enable you to deactivate the mandatory groups, segments, composites, and elements. By default, Gentran:Director displays active map components with a black fount, and inactive map components with a grey fount (dimmed). In the example below, the ORDHDR group, TYP segment, SDT segment, and the SIDN composite are mandatory, so they are already activated by the system. You also need to use the SDT SIDN SID0 element (a conditional element), so you need to activate it.



```
ORDHDR -M- 1 ORDER FILE HEADER
├─TYP -M- 1 TRANSACTION TYPE DETAILS
├─SDT -M- 1 SUPPLIER DETAILS
│  ├─SIDN -M-
│  │  ├─01 SID0 -C- String 13 SUPPLIER'S ANA LOCATION CODE
│  │  └─02 SID1 -C- String 17 SUPPLIER'S IDENT. USED BY CUSTOMER
│  ├─03 SNAM -C- String 40 SUPPLIER'S NAME
│  ├─SADD -C-
│  └─VATN -C-
```

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

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

1. Double-click the groups, segments, and composites that are listed in the table below to open them. The table below contains the groups, segments, and elements that you will need to activate. The composite (Comp.) column is provided for your information, so you will know

where to locate the appropriate element(s). Opening all the necessary groups, segments, and composites before you start to activate allows you to scroll down the map and activate the segments and elements sequentially.

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

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

2. Click the Activate button on the Main Toolbar.
3. Click each inactive segment that you need to use. This will activate the segments.

If you accidentally click a segment that you did not mean to activate, click the segment with the right mouse button to access the shortcut menu. Select Deactivate from the shortcut menu.

4. Click each inactive element that you need to use. This will activate the elements.

If you accidentally click an element that you did not mean to activate, click the element with the right mouse button to access the shortcut menu. Select Deactivate from the shortcut menu.

5. Once you have activated all the necessary groups, segments, and elements, click the Activate button on the Main Toolbar to turn activation mode off.

Verifying EDI Delimiters

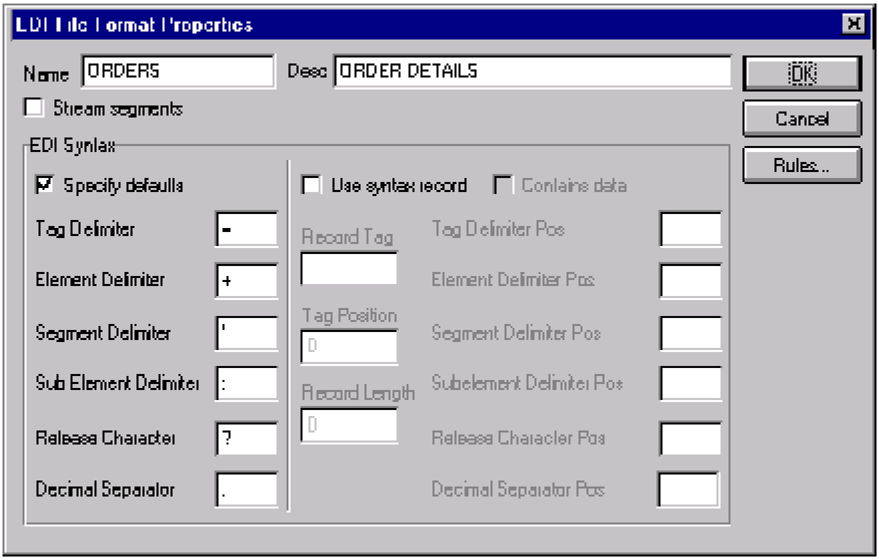
If you are using an EDI standard that contains composite elements or sub-elements, you must verify that Gentran:Director is specifying the correct delimiters. Delimiters are flags that you define to the system as separating specific EDI components. Delimiters are necessary for all variable field-length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

STOP

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

Complete the following steps to verify EDI delimiters:

1. Click the ORDERS ORDER DETAILS EDI file icon with the right mouse button to access the shortcut menu. Select Properties... from the shortcut menu to display the EDI File Format Properties dialogue.



EDI File Format Properties

Name: ORDERS

Desc: ORDER DETAILS

□ Stream segments

EDI Syntax

☑ Specify defaults

Tag Delimiter	=
Element Delimiter	+
Segment Delimiter	'
Sub Element Delimiter	:
Release Character	?
Decimal Separator	.

□ Use syntax record □ Contains data

Record Tag	
Tag Position	0
Record Length	0
Tag Delimiter Pos	
Element Delimiter Pos	
Segment Delimiter Pos	
Subelement Delimiter Pos	
Release Character Pos	
Decimal Separator Pos	

2. In the EDI Syntax section, verify that the Specify defaults check box is selected.

3. Verify the required delimiters for the TRADACOMS standard. If the delimiters differ from the defaults in the table below, type either the character or the hexadecimal value in the correct field. The default delimiters for TRADACOMS are the following:

Field	TRADACOMS Default Character	TRADACOMS Default Hex Value
Tag Delimiter	= (equal sign)	0x3D
Element Delimiter	+ (plus sign)	0x2B
Segment Delimiter	' (single quote)	0x27
Sub Element Delimiter	: (colon)	0x3A
Release Character	? (question mark)	0x3F
Decimal Separator	. (full stop)	0x2E

4. Click OK to exit the EDI File Format Properties dialogue.

Defining Your Application File Format

If either side of your map is positional (fixed-format), you must define your application file format to Gentran:Director. In this tutorial, the output side of the map is positional. Your application file must contain all the information that you need to extract from your partner’s message, so that your system can process the data accurately.

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

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

Creating the POHEADER Record

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

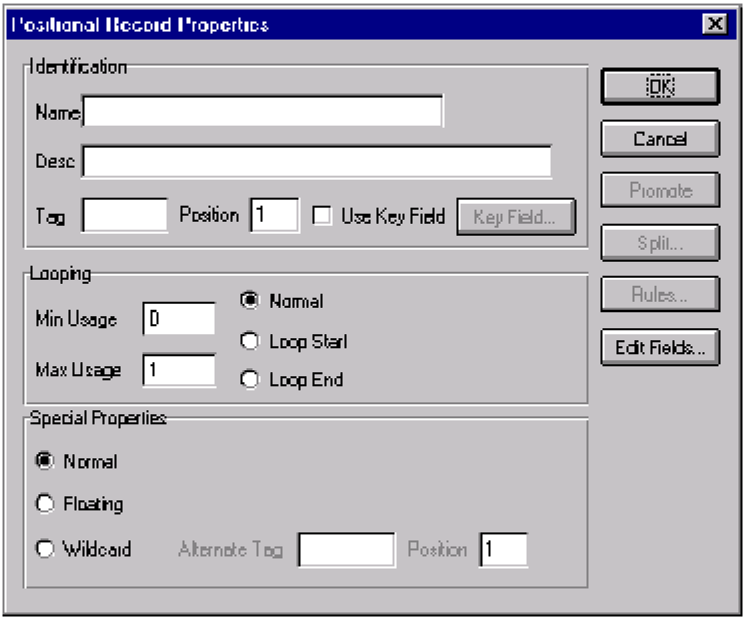
The POHEADER (header) record contains the purchase order type code, shipping and customer information, delivery date information, and the number of line items in the purchase order. This header record is mandatory and occurs once for each purchase order translated.

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

Complete the following steps to create the first application record (POHEADER):

1. Click the Positional File icon to highlight it (Output for an inbound map).
2. Select Create Sub from the Edit menu. Select Record from the cascading menu.

The Positional Record Properties dialogue is displayed.



Name
Desc
Tag
Position 1
□ Use Key Field
Min Usage 0
Max Usage 1
● Normal
○ Loop Start
○ Loop End
● Normal
○ Floating
○ Wildcard
Alternate Tag
Position 1

3. In the Name field, type “POHEADER” as the record identification code.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

4. In the Desc field, type “Header Record” as the description of the record. This field is used to provide a brief explanation of the record, which will allow you to differentiate it from similar records.

5. In the Tag field, type “HDR” as the record identification code <TAG>. For example, a record is recognised by the system as the following:

   <TAG>[Field_1][Field_2]

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

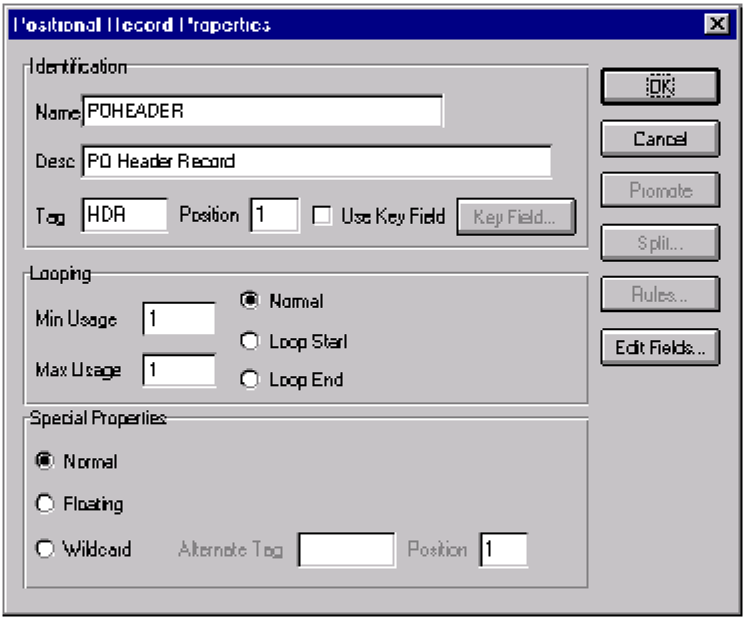
We recommend that you define the record tag on the Positional Record Properties dialogue, instead of defining fields with the purpose of explicitly containing the record tag.

6. In the Position field, accept the default of “1” as the starting column position of the tag in the data record.
7. In the Min Usage field, type “1” to make the header record mandatory. This is the minimum number of times the record must repeat.

If the Min Usage field contains “0” (zero), the record is conditional. If the Min Usage field contains a “1” or greater, the record is mandatory.

If you want the record to repeat (loop), type the maximum number of times it can repeat in the Max Usage field. However, for the header record, we recommend that you always accept the default of “1” so it remains a record that only occurs once (not a looping structure).

The Positional Record Properties dialogue should now look like this:



Positional Record Properties

Identification

Name: POHEADER

Desc: PO Header Record

Tag: HDR Position: 1 ☐ Use Key Field

Looping

Min Usage: 1

Max Usage: 1

● Normal ○ Loop Start ○ Loop End

Special Properties

● Normal

○ Floating

○ Wildcard Alternate Tag: Position: 1

8. Click OK to create the POHEADER record.

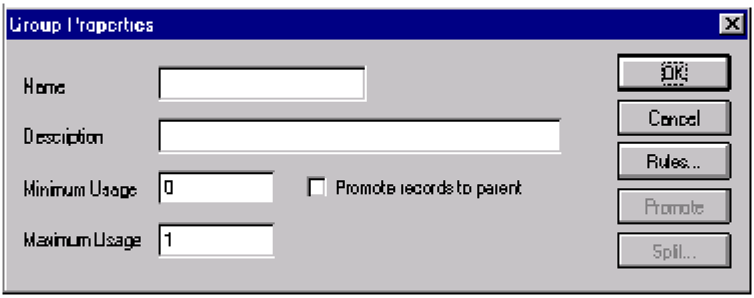
Creating the PODetailGroup

The PODetailGroup (detail information) contains all the line item information for the order, including the item, unit of measure, price, quantity ordered, description, and the allowance and charge information. Each PODetailGroup can occur up to 200,000 times per order translated. This equals a maximum order of 200,000 items. The information within the PODetailGroup is further segregated by related records that repeat in sequence until either the data ends or the maximum number of times that the loop is allowed to repeat is exhausted.

Complete the following steps to create the PODetailGroup at the same level as the POHEADER record:

1. Click the POHEADER record to highlight it and select Insert from the Edit menu. Select Group from the cascading menu.

   The Group Properties dialogue is displayed.



Group Properties

Name

Description

Minimum Usage 0 □ Promote records to parent

Maximum Usage 1

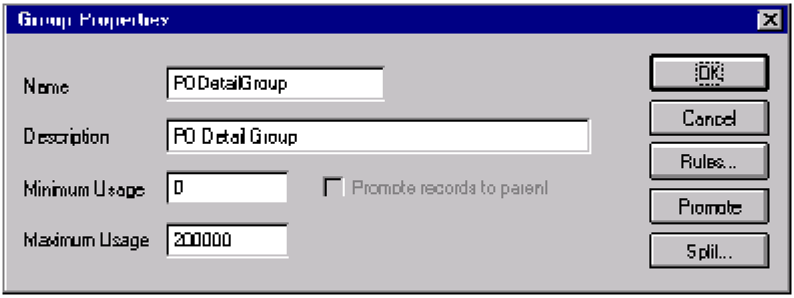
2. In the Name field, type “PODetailGroup” to indicate that this group will contain all of the detail records. This is the segment loop identification code.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

3. In the Description field, type “PO Detail Group” as the loop description. This field is used to provide a brief explanation of the loop.

   Accept the default of “0” in the Minimum Usage field. This is the minimum number of times the loop must be repeated. For a conditional loop, the minimum usage should always be “0” (zero). For a mandatory loop, the minimum usage should be “1” or greater.

4. In the Maximum Usage field, type “200000”. This indicates that this group of records can repeat up to 200,000 times. The Group Properties dialogue should now look like this:



Name: PODetailGroup

Description: PO Detail Group

Minimum Usage: 0

☐ Promote records to parent

Maximum Usage: 200000

5. Click OK to create the PODetailGroup.

Creating the PODETAIL and PODESC Records

In this tutorial, the PODetailGroup contains two subordinate records: PODETAIL and PODESC. After you create the first subordinate record (PODETAIL), you can define the subsequent record at the same level (PODESC).

Complete the following steps to create the PODETAIL record:

1. Click the PODetailGroup to highlight it and select Create Sub from the Edit menu. Select Record from the cascading menu.

   The Positional Record Properties dialogue is displayed.

2. In the Name field, type “PODETAIL” as the record identification code.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

3. In the Desc field, type “Detail Record” as the description of the record. This field is used to provide a brief explanation of the record, which will allow you to differentiate it from similar records.

4. In the Tag field, type “DET” as the record identification code <TAG>. For example, a record is recognised by the system as the following:

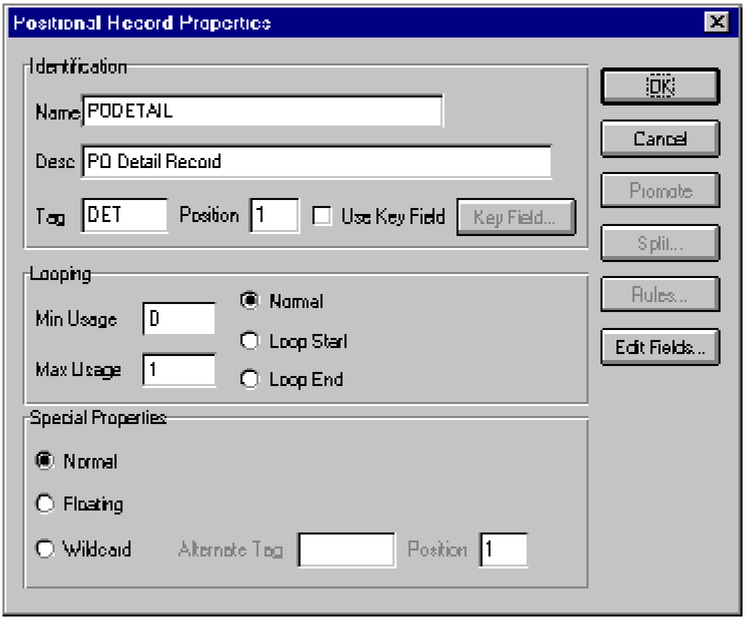
   <TAG>[Field_1][Field_2]

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

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

5. In the Position field, accept the default of “1” as the starting column position of the tag in the data record.

   The Positional Record Properties dialogue should now look like this:



Identification

Name: PODETAIL

Desc: PO Detail Record

Tag: DET Position: 1 ☐ Use Key Field

Looping

Min Usage: 0

Max Usage: 1

● Normal
○ Loop Start
○ Loop End

Special Properties

● Normal
○ Floating
○ Wildcard Alternate Tag: Position: 1

6. Click OK to create the PODETAIL record.

Complete the following steps to create the PODESC record:

1. Click the PODETAIL record to highlight it and select Insert from the Edit menu. Select Record from the cascading menu to display the Positional Record Properties dialogue.

2. In the Name field, type “PODESC” the record identification code.

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

3. In the Desc field, type “Description Record” as a description of the record. This field is used to provide a brief explanation of the record, which will allow you to differentiate it from similar records.

4. In the Tag field, type “DES” as the record identification code <TAG>. For example, a record is recognised by the system as the following:

   <TAG>[Field_1][Field_2]

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

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

5. In the Position field, accept the default of “1” as the starting column position of the tag in the data record.

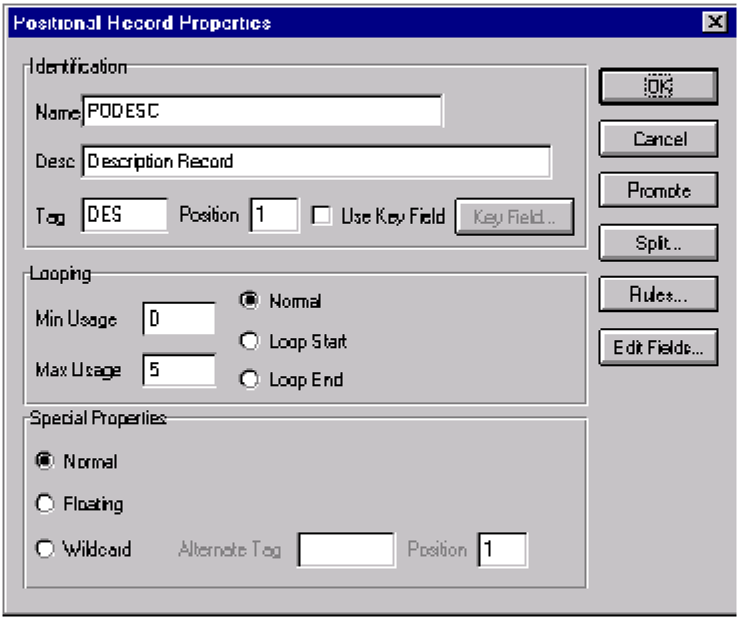
   The Looping section of the Positional Record Properties dialogue enables you to specify the minimum and maximum number of times this record can repeat, and whether a record is mandatory or conditional. Alter this section *only* if the record is a looping structure. Otherwise, accept the defaults.

   Accept the default of “0” in the Min Usage field. This is the minimum number of times the record must repeat.

   If the Min Usage field contains “0” (zero), the record is conditional. If the Min Usage field contains “1” or greater, the record is mandatory.

6. In the Max Usage field, type “5” as the maximum number of times the record can repeat.

The Positional Record Properties dialogue should now look like this:



Identification

Name: PODESC

Desc: Description Record

Tag: DES Position: 1 ☐ Use Key Field

Looping

Min Usage: 0

Max Usage: 5

● Normal

○ Loop Start

○ Loop End

Special Properties

● Normal

○ Floating

○ Wildcard Alternate Tag: Position: 1

7. Click OK to create the PODESC record.

Creating the POSUMMARY Record

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

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

Complete the following steps to create the POSUMMARY record:

1. Click the PODetailGroup to highlight it and select Insert from the Edit menu. Select Record from the cascading menu to display the Positional Record Properties dialogue.

Remember to select "Insert" from the Edit menu when you want to create equal records/groups and select "Create Sub" from the Edit menu to create subordinate groups/records/fields.

2. In the Name field, type "POSUMMARY" as the record identification code.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

3. In the Desc field, type “Summary Record” as the description of the record. This field is used to provide a brief explanation of the record, which will allow you to differentiate it from similar records.
4. In the Tag field, type “SUM” as the record identification code <TAG>. For example, a record is recognised by the system as the following:

   <TAG>[Field_1][Field_2]

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

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

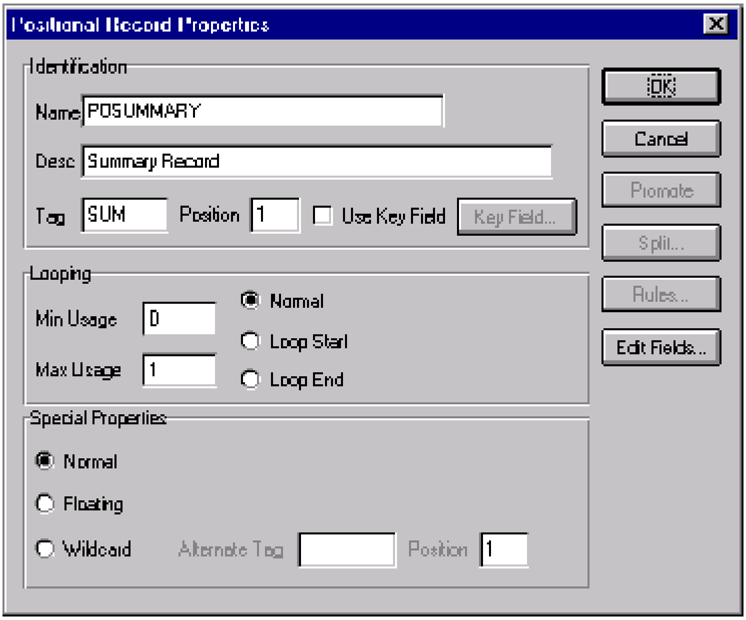
5. In the Position field, accept the default of “1” as the starting column position of the tag in the data record.

   Accept the default of “0” in the Min Usage field. This is the minimum number of times the record must repeat.

   If the Min Usage field contains “0” (zero), the record is conditional. If the Min Usage field contains “1” or greater, the record is mandatory.

   Also, accept the default of “1” in the Max Usage field so the summary record only occurs once (not a looping structure).

The Positional Record Properties dialogue should now look like this:



Identification

Name POSUMMARY

Desc Summary Record

Tag SUM Position 1 ☐ Use Key Field

Looping

Min Usage 0 ● Normal

Max Usage 1 ○ Loop Start

○ Loop End

Special Properties

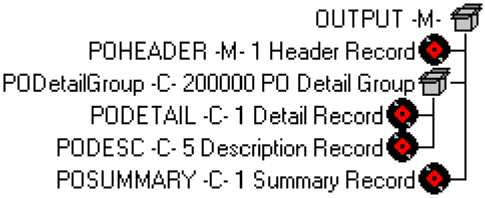
● Normal

○ Floating

○ Wildcard Alternate Tag Position 1

6. Click OK to create the POSUMMARY record.

The application side of the map (output) should now look like this:



OUTPUT -M-
POHEADER -M- 1 Header Record
PODetailGroup -C- 200000 PO Detail Group
PODETAIL -C- 1 Detail Record
PODESC -C- 5 Description Record
POSUMMARY -C- 1 Summary Record

Creating the Application Fields

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

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

The application fields that you need to create are listed under their respective records in the following table. The name, mandatory status, description, data type, and field length for each field are given in the table.

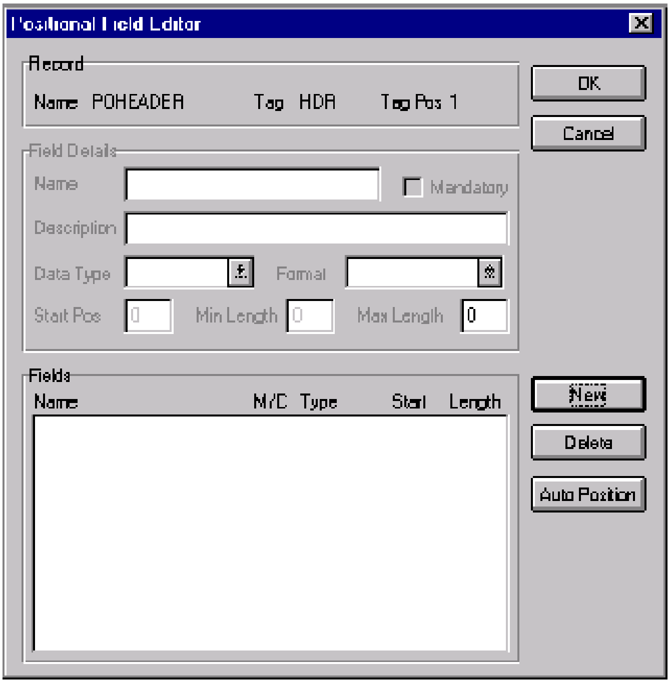
Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (POHEADER) 1 per message					
CUSTNUMHDR	X	Customer Number	String	1	8
PODATEHDR	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMHDR	X	Purchase Order Number	String	1	22
POTYPCODE		Purchase Order Type Code	String	0	2
SHIPTONAME		Ship To Name	String	0	35
SHIPTOADD1		Ship To Address Line 1	String	0	35
SHIPTOADD2		Ship To Address Line 2	String	0	35
SHIPTOCITY		Ship To City	String	0	35
SHIPTOSTATE		Ship To State, Province, or Country	String	0	35
SHIPTOPOSTALCODE		Ship To Postal Code	String	0	9
CUSTNAME		Customer Name	String	0	35
CUSTADD1		Customer Address Line 1	String	0	35
CUSTADD2		Customer Address Line 2	String	0	35
CUSTCITY		Customer City	String	0	35
CUSTSTATE		Customer State, Province, or Country	String	0	35
CUSTPOSTAL CODE		Customer Postal Code	String	0	9
DELDATE	X	Delivery Date	Dt/Tm (DD/MM/YY)	8	8
NUMLINEITEMS	X	Number of Line Items	Number (N0)	1	6
PO Detail Group (PODetailGroup) up to 200,000 per message					
Detail Record (PODETAIL) 1 per message					
CUSTNUMDET	X	Customer Number	String	1	8
PODATEDET	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMDET	X	Purchase Order Number	String	1	22
CUSTPROCODE	X	Customer Product Code	String	1	30

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
UPCCODE	X	UPC Code	String	1	30
CONSUMERUNITS		Consumer Units in Trading Unit	Number (N0)	0	15
UOM		Unit of Measure	String	0	7
UNITPR		Unit Price	Number (R2)	0	14
QTYORD	X	Quantity Ordered	Number (N0)	1	9
Description Record (PODESC) up to 5 per message					
CUSTNUMDESC	X	Customer Number	String	1	8
PODATEDESC	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMDESC	X	Purchase Order Number	String	1	22
DESC		Description	String	0	81
Summary Record (POSUMMARY) 1 per message					
CUSTNUMSUM	X	Customer Number	String	1	8
PODATESUM	X	Purchase Order Date	Dt/Tm (DD/MM/YY)	8	8
PONUMSUM	X	Purchase Order Number	String	1	22
NUMLINEITEM SUM	X	Number of Line Items	Number (N0)	1	6

Complete the following steps to create the application fields for the POHEADER record:

1. Click the POHEADER application record with the right mouse button to access the shortcut menu.

2. Select Edit Fields... from the shortcut menu to access the Positional Field Editor dialogue.



Name POHEADER Tag HDR Tag Pos 1

Start Pos 0 Min Length 0 Max Length 0

3. If the field you are creating is the first field in the record, just click NEW. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.

4. In the Name field, type “CUSTNUMHDR”, which is the name of the first field in the POHEADER record. Each application field must have a unique name. It is useful to tag the end of each field that occurs in multiple records with a suffix that identifies the record that contains it. For example, use the tag “HDR” for the Header record.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

5. Click the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system will not be able to identify who sent the purchase order.

6. In the Description field, type “Customer Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.

7. From the Data Type drop-down list box, select “String” as the type of the field, to specify that this field must contain alphanumeric data.

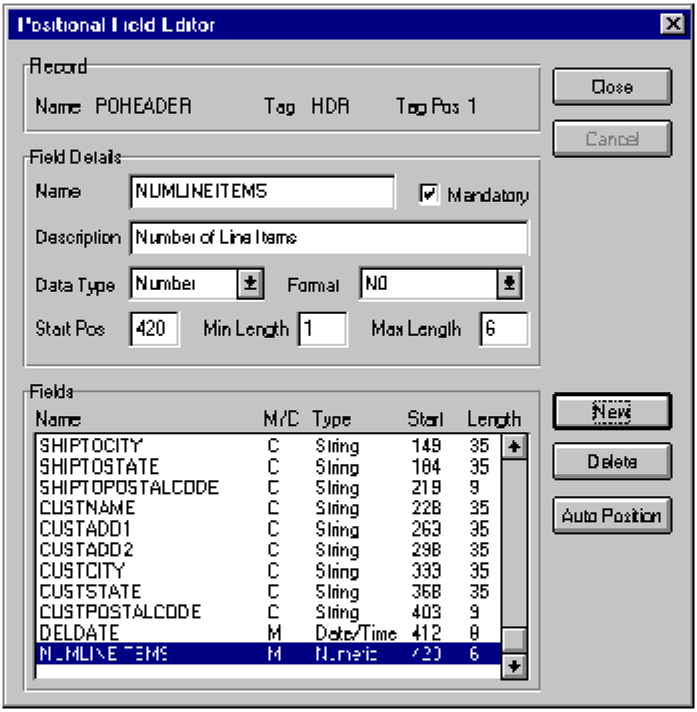
If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos field. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.

In this tutorial, you will use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see step 13. below for more information on the Auto Position function.

8. If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format drop-down list box.
9. In the Min Length field, type “1” as the minimum field length. If the minimum length of the field is “0”, then accept that default in the Min Length field.
10. In the Max Length field, type “8” as the maximum length of the field.
11. To add the field and to create the next field positioned after it, press **ENTER**. A new field with blank values ready for you to identify will be positioned below the field.
12. Create the rest of the fields according to the record layout table starting on page 50.
13. .After you add the last field, click AUTO POSITION to automatically position the fields in the record. The Auto Position function will automatically calculate the start position in the record of each field based on the criteria that each field is positioned directly after the previous field, and is of the length specified in the Max Length field. Click YES to acknowledge the warning message that fields will be sequenced in order.

STOP **Please note that it is only valid to use the Auto Position function if you define a record tag on the Positional Record Properties dialogue, and if you define every field in the record in the sequence that each field occurs (as you are doing in this tutorial).**

The Positional Field Editor dialogue should now look like the following (only the last eleven fields in the record are currently visible in the Fields list box):



Record

Name POHEADER Tag HDR Tag Pos 1

Field Details

Name NUMLINEITEMS ☑ Mandatory

Description Number of Line Items

Data Type Number Format N0

Start Pos 420 Min Length 1 Max Length 6

Fields

Name	M/C	Type	Start	Length
SHIPTOCITY	C	String	149	35
SHIPTOSTATE	C	String	184	35
SHIPTOPOSTALCODE	C	String	219	9
CUSTNAME	C	String	228	35
CUSTADD1	C	String	263	35
CUSTADD2	C	String	298	35
CUSTCITY	C	String	333	35
CUSTSTATE	C	String	368	35
CUSTPOSTALCODE	C	String	403	9
DELDATE	M	Date/Time	412	8
NUMLINEITEMS	M	Numeric	420	6

14. When you finish adding fields to the POHEADER record, click CLOSE.

Complete the following steps to create the application fields for the PODETAIL record:

1. Click the PODETAIL application record with the right mouse button to access the shortcut menu.
2. Select Edit Fields... from the shortcut menu to access the Positional Field Editor dialogue.
3. If the field you are creating is the first field in the record, just click NEW. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4. In the Name field, type “CUSTNUMDET”, which is the name of the first field in the PODETAIL record. Each application field must have a unique name. It is useful to tag the end of the each field that occurs in multiple records with a suffix that identifies the record that contains it. For example, use the tag “DET” for the detail record.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

5. Click the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system will not be able to identify who sent the purchase order.
6. In the Description field, type “Customer Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7. From the Data Type drop-down list box, select “String” as the type of the field, to specify that this field must contain alphanumeric data.

If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos field. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.

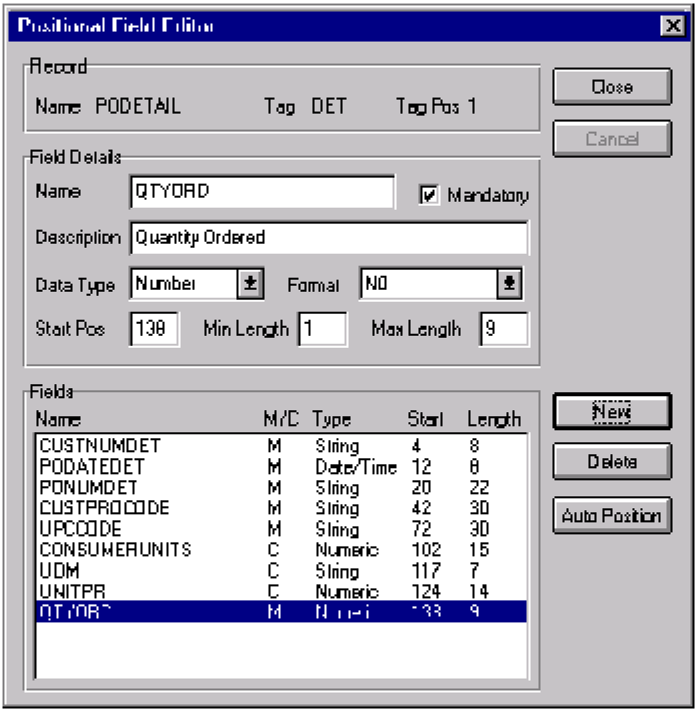
In this tutorial, you will use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see step 13. below for more information on the Auto Position function.

8. If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format drop-down list box.
9. In the Min Length field, type “1” as the minimum field length. If the minimum length of the field is “0”, then just accept that default in the Min Length field.
10. In the Max Length field, type “8” as the maximum length of the field.
11. To add the field and to create the next field positioned after it, press **ENTER**. A new field with blank values ready for you to identify will be positioned below the field.
12. Create the rest of the fields according to the record layout table starting on page 50.
13. After you add the last field, click AUTO POSITION to automatically position the fields in the record. The Auto Position function will automatically calculate the start position in the record of each field based on the criteria that each field is positioned directly after the previous field, and is of the length specified in the Max Length field.

Click YES to acknowledge the warning message that fields will be sequenced in order.

STOP **Please note that it is only valid to use the Auto Position function if you define a record tag on the Positional Record Properties dialogue, and if you define every field in the record in the sequence that each field occurs (as you are doing in this tutorial).**

The Positional Field Editor dialogue should now look like the following:



Record

Name PODETAIL Tag DET Tag Pos 1

Field Details

Name QTYORD ☑ Mandatory

Description Quantity Ordered

Data Type Number Format N0

Start Pos 138 Min Length 1 Max Length 9

Fields

Name	M/C	Type	Start	Length
CUSTNUMDET	M	String	4	8
PODATEDET	M	Date/Time	12	8
PONUMDET	M	String	20	22
CUSTPRODCODE	M	String	42	30
UPCCODE	M	String	72	30
CONSUMERUNITS	C	Numeric	102	15
UOM	C	String	117	7
UNITPR	C	Numeric	124	14
QTYORD	M	Numeric	138	9

14. When you finish adding fields to the PODETAIL record, click CLOSE.

Complete the following steps to create the application fields for the PODESC record:

1. Click the PODESC application record with the right mouse button to access the shortcut menu.

2. Select Edit Fields... from the shortcut menu to access the Positional Field Editor dialogue.

3. If the field you are creating is the first field in the record, just click NEW. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4. In the Name field, type “CUSTNUMDESC”, which is the name of the first field in the PODESC record. Each application field must have a unique name. It is useful to tag the end of each field that occurs in multiple records with a suffix that identifies the record that contains it. For example, use the tag “DESC” for the description record.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

5. Click the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system will not be able to identify who sent the purchase order.
6. In the Description field, type “Customer Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7. From the Data Type drop-down list box, select “String” as the type of the field, to specify that this field must contain alphanumeric data.

If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos field. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.

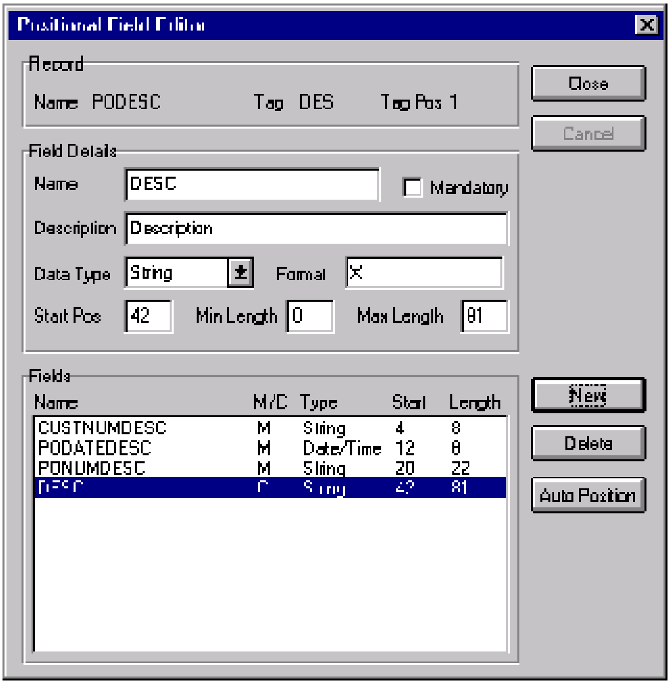
In this tutorial, you will use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see step 13. below for more information on the Auto Position function.

8. If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format drop-down list box.
9. In the Min Length field, type “1” as the minimum field length. If the minimum length of the field is “0”, then accept that default in the Min Length field.
10. In the Max Length field, type “8” as the maximum length of the field.

11. To add the field and to create the next field positioned after it, press **ENTER**. A new field with blank values ready for you to identify will be positioned below the field.
12. Create the rest of the fields according to the record layout table starting on page 50.
13. After you add the last field, click AUTO POSITION to automatically position the fields in the record. The Auto Position function will automatically calculate the start position in the record of each field based on the criteria that each field is positioned directly after the previous field, and is of the length specified in the Max Length field. Click YES to acknowledge the warning message that fields will be sequenced in order.

STOP **Please note that it is only valid to use the Auto Position function if you define a record tag on the Positional Record Properties dialogue, and if you define every field in the record in the sequence that each field occurs (as you are doing in this tutorial).**

The Positional Field Editor dialogue should now look like this:



Record
Name PODESC Tag DES Tag Pos 1

Field Details
Name DESC ☐ Mandatory
Description Description
Data Type String
Format X
Start Pos 42 Min Length 0 Max Length 81

Fields

Name	M/C	Type	Start	Length
CUSTNUMDESC	M	String	4	8
PODATEDESC	M	Date/Time	12	8
PONUMDESC	M	String	20	22
DESC	C	String	42	81

14. When you finish adding fields to the PODESC record, click CLOSE.

Complete the following steps to create the application fields for the POSUMMARY record:

1. Click the POSUMMARY application record with the right mouse button to access the shortcut menu.
2. Select Edit Fields... from the shortcut menu to access the Positional Field Editor dialogue.
3. If the field you are creating is the first field in the record, just click NEW. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4. In the Name field, type “CUSTNUMSUM”, which is the name of the first field in the POSUMMARY record. Each application field must have a unique name. It is useful to tag the end of each field that occurs in multiple records with a suffix that identifies the record that contains it. For example, “SUM” for the summary record.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

5. Click the Mandatory check box to identify it as a required field. For this example, the customer number must be present or else your system will not be able to identify who sent the purchase order.
6. In the Description field, type “Customer Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7. From the Data Type drop-down list box, select “String” as the type of the field, to specify that this field must contain alphanumeric data.

If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos field. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.

In this tutorial, you will use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see step 13. below for more information on the Auto Position function.

8. If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format drop-down list box.

9. In the Min Length field, type “1” as the minimum field length. If the minimum length of the field is “0”, then just accept that default in the Min Length field.

10. In the Max Length field, type “8” as the maximum length of the field.

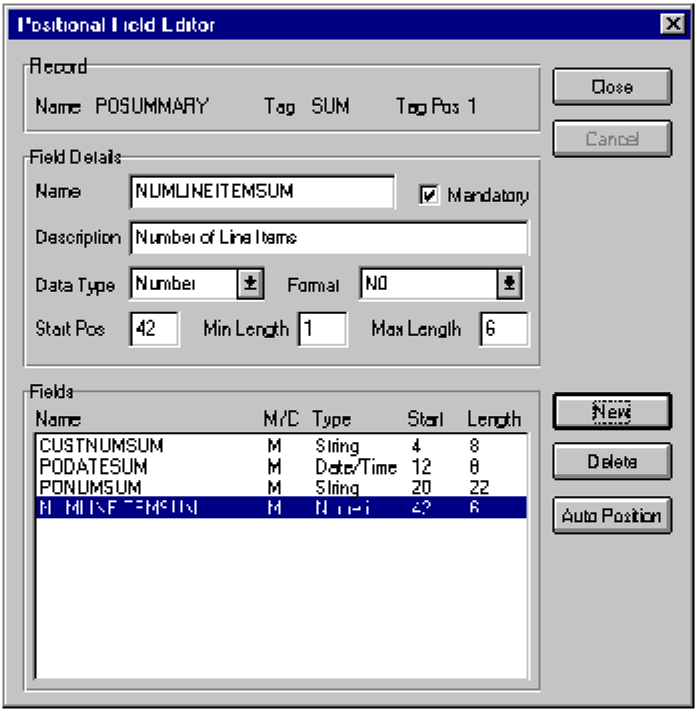
11. To add the field and to create the next field positioned after it, press **ENTER**. A new field with blank values ready for you to identify will be positioned below the field.

12. Create the rest of the fields according to the record layout table starting on page 50.

13. .After adding the last field, click AUTO POSITION to automatically position the fields in the record. The Auto Position function will automatically calculate the start position in the record of each field based on the criteria that each field is positioned directly after the previous field, and is of the length specified in the Max Length field. Click YES to acknowledge the warning message that fields will be sequenced in order.

STOP **Please note that it is only valid to use the Auto Position function if you define a record tag on the Positional Record Properties dialogue, and if you define every field in the record in the sequence that each field occurs (as you are doing in this tutorial).**

The Positional Field Editor dialogue should now look like the following:



Record

Name POSUMMARY Tag SUM Tag Pos 1

Field Details

Name NUMLINEITEMSUM ☑ Mandatory

Description Number of Line Items

Data Type Number Format N0

Start Pos 42 Min Length 1 Max Length 6

Fields

Name	M/C	Type	Start	Length
CUSTNUMSUM	M	String	4	8
PODATESUM	M	Date/Time	12	8
PONUMSUM	M	String	20	22
NUMLINEITEMSUM	M	Number	42	6

14. When you are completely finished adding fields to the POSUMMARY record, click CLOSE.

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

Mapping Your Data

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

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

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

POHEADER Record

This section describes the actions necessary for you to map to each field in the POHEADER (header) record. The POHEADER (header) record contains the purchase order type code, shipping and customer information, delivery date information, and the number of line items in the purchase order. This header record is mandatory and occurs once for each purchase order translated.

CUSTNUMHDR

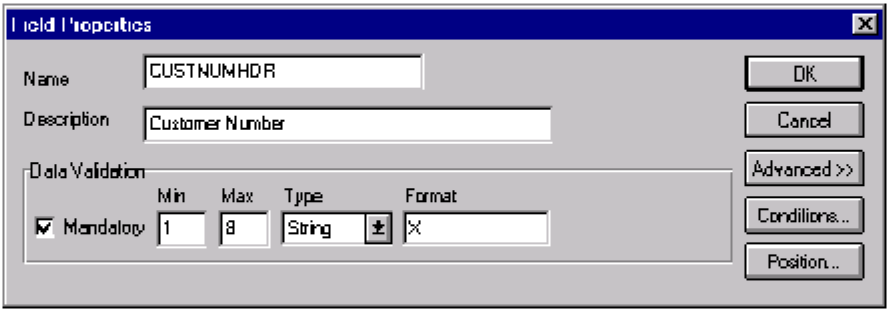
The Application Integration subsystem allows you to use selected information from the Gentran:Director Partner database in your maps via a standard rule. You can map information from your trading partner's profile in the Partner Editor to a selected element or field in the map. The information that you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

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

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

Complete the following steps to map the customer number from the document record:

1. Double-click the CUSTNUMHDR field (in the POHEADER record) to display the Field Properties dialogue.


Field Properties

Name: CUSTNUMHDR

Description: Customer Number

Data Validation

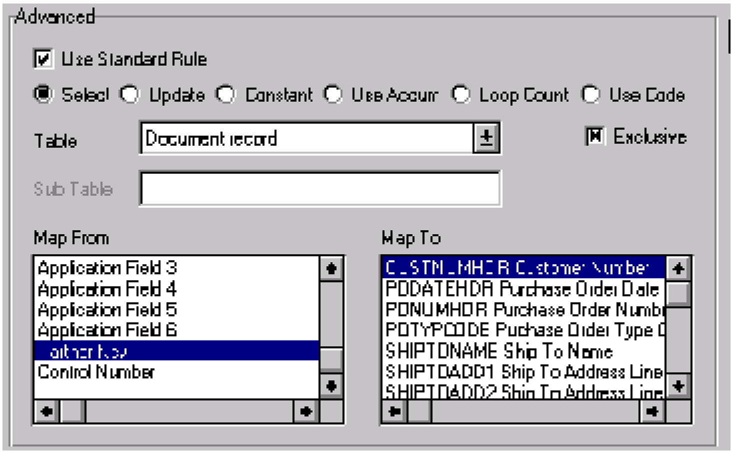
☑ Mandatory — Min: 1 — Max: 8 — Type: String — Format: X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.

The Advanced section will be displayed only if you previously clicked ADVANCED>> on a Field Properties dialogue (during this session of Gentran:Director), and did not click <<ADVANCED to hide this section before closing the Field Properties dialogue.

3. In the Advanced section, click the Use Standard Rule check box to display the rule options.
4. Verify that the Select radio button is selected.
5. From the Table drop-down list box, select “Document record”. This indicates that the system will update this field with selected information from the document record.
6. Click the Exclusive check box to indicate that a compliance error should be generated if the Select function does not find a valid entry.
7. From the Map From list box, choose “Partner Key”. This is the field that contains the customer number for this partner.
8. From the Map To list box, choose “CUSTNUMHDR”. This is the field to which the information from the Partner Editor will be mapped.

The Advanced section of the Field Properties dialogue should now look like this:



Advanced

☑ Use Standard Rule

● Select ○ Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table: Document record

☒ Exclusive

Sub Table:

Map From:
- Application Field 3
- Application Field 4
- Application Field 5
- Application Field 6
- [illegible]
- Control Number

Map To:
- CUSTNUMHDR Customer Number
- PODATEHDR Purchase Order Date
- PONUMHDR Purchase Order Numb
- POTYPCODE Purchase Order Type C
- SHIPTONAME Ship To Name
- SHIPTOADD1 Ship To Address Line
- SHIPTOADD2 Ship To Address Line

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

PODATEHDR

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

Complete the following steps to map the date order placed by customer:

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

PONUMHDR

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

We recommend that you set up a Document Name for each map, to make the identification of a message created by this map easier in Gentran:Director. The Document Name allows you to differentiate between messages in the document browsers in Gentran:Director.

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

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

Mapping the customer’s order number

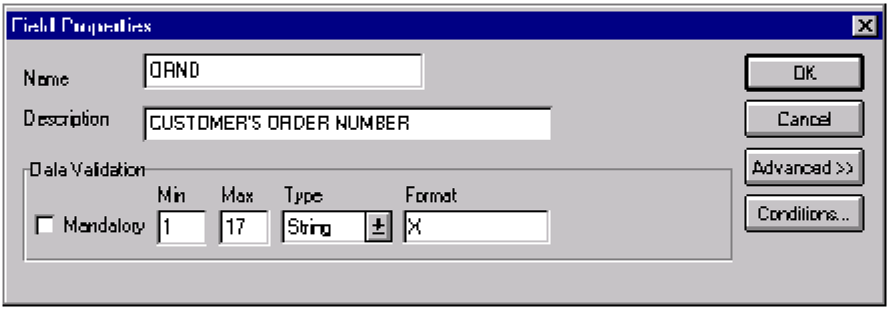
Complete the following steps to map the customer’s order number:

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

Setting up the document name

Complete the following steps to set up the document name for this purchase order:

1. Double-click the ORD ORNO ORN0 element (Customer’s Order Number) to display the Field Properties dialogue.



Field Properties

Name: ORNO

Description: CUSTOMER'S ORDER NUMBER

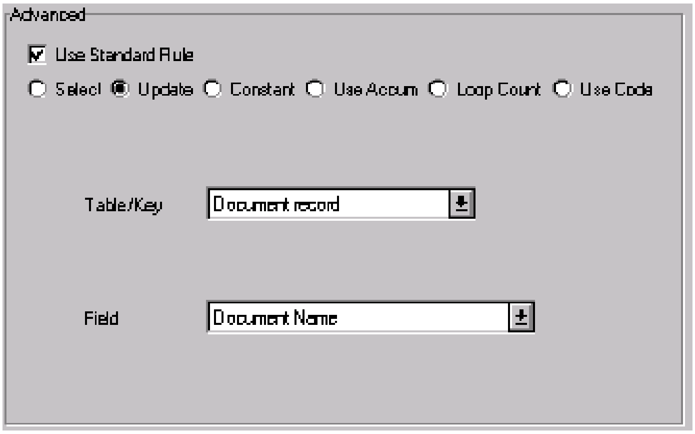
Data Validation

□ Mandatory

Min	Max	Type	Format
1	17	String	X

2. If the Advanced section of the Field Properties dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. In the Advanced section, click the Use Standard Rule check box, to display the rule options.
4. Click the Update radio button.
5. From the Table/Key drop-down list box, choose “Document record”. This indicates that you are updating the document record in Gentran:Director.
6. From the Field drop-down list box, choose “Document Name.” This indicates that you are updating the Name field in the document browsers with the contents of this element.

The Advanced section of the dialogue should now look like this:



Advanced

☑ Use Standard Rule

○ Select ● Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table/Key: Document record

Field: Document Name

7. Click OK to set up the document name.

POTYPCODE

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

Complete the following steps to map the coded message function:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the ORD CD07 element (Order Code). When you click the element, the cursor changes to a "Link Arrow": LINK ⇨
3. Click the application field POTYPCODE in the POHEADER record.
4. A line is displayed, visually illustrating the link between the ORD CD07 element and the POTYPCODE field.

SHIPTONAME

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

Complete the following steps to map the ship to name:

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

SHIPTOADD1

Complete the following steps to map the ship to address line 1:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the CLO CADD CAD0 element (Customer’s Address Line 1). When you click the element, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the application field SHIPTOADD1 in the POHEADER record.
4. A line is displayed, visually illustrating the link between the CLO CADD CAD0 element and the SHIPTOADD1 application field.

SHIPTOADD2

Complete the following steps to map the ship to address line 2:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the CLO CADD CAD1 element (Customer’s Address Line 2). When you click the element, the cursor changes to a “Link Arrow”.
3. Click the application field SHIPTOADD2 in the POHEADER record.
4. A line is displayed, visually illustrating the link between the CLO CADD CAD1 element and the SHIPTOADD2 application field.

SHIPTOCITY

Complete the following steps to map the ship to city:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the CLO CADD CAD2 element (Customer’s Address Line 3). When you click the element, the cursor changes to a “Link Arrow”.
3. Click the application field SHIPTOCITY in the POHEADER record.
4. A line is displayed, visually illustrating the link between the CLO CADD CAD2 element and the SHIPTOCITY application field.

SHIPTOSTATE

Complete the following steps to map the ship to state:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the CLO CADD CAD3 element (Customer’s Address Line 4). When you click the element, the cursor changes to a “Link Arrow”.
3. Click the application field SHIPTOSTATE in the POHEADER record.
4. A line is displayed, visually illustrating the link between the CLO CADD CAD3 element and the SHIPTOSTATE application field.

SHIPTOPOSTAL CODE

Complete the following steps to map the ship to postal code:

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

CUSTNAME

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

Complete the following steps to map the customer name:

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

CUSTADD1

Complete the following steps to map the customer address line 1:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the CDT CADD CAD0 element (Customer's Address Line 1). When you click the element, the cursor changes to a "Link Arrow".
3. Click the application field CUSTADD1 in the POHEADER record.
4. A line is displayed, visually illustrating the link between the CDT CADD CAD0 element and the CUSTADD1 application field.

CUSTADD2

Complete the following steps to map the customer address line 2:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.

2. Click the CDT CADD CAD1 element (Customer’s Address Line 2). When you click the element, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the application field CUSTADD2 in the POHEADER record.
4. A line is displayed, visually illustrating the link between the CDT CADD CAD1 element and the CUSTADD2 application field.

CUSTCITY

Complete the following steps to map the customer city:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the CDT CADD CAD2 element (Customer’s Address Line 3). When you click the element, the cursor changes to a “Link Arrow”.
3. Click the application field CUSTCITY in the POHEADER record.
4. A line is displayed, visually illustrating the link between the CDT CADD CAD2 element and the CUSTCITY application field.

CUSTSTATE

Complete the following steps to map the customer state:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the CDT CADD CAD3 element (Customer’s Address Line 4). When you click the element, the cursor changes to a “Link Arrow”.
3. Click the application field CUSTSTATE in the POHEADER record.
4. A line is displayed, visually illustrating the link between the CDT CADD CAD3 element and the CUSTSTATE application field.

CUSTPOSTAL CODE

Complete the following steps to map the customer postal code:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the CDT CADD CAD4 element (Customer’s Post Code). When you click the element, the cursor changes to a “Link Arrow”.
3. Click the application field CUSTPOSTALCODE in the POHEADER record.
4. A line is displayed, visually illustrating the link between the CDT CADD CAD4 element and the CUSTPOSTALCODE application field.

DELDATE

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

Complete the following steps to map the delivery date:

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

NUMLINEITEMS

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

Complete the following steps to map the number of line items:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the OTR LORD element (Lines Ordered). When you click the element, the cursor changes to a "Link Arrow".
3. Click the application field NUMLINEITEMS in the POHEADER record.
4. A line is displayed, visually illustrating the link between the OTR LORD element and the NUMLINEITEMS field.

PODetailGroup

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

PODETAIL Record

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

CUSTNUMDET

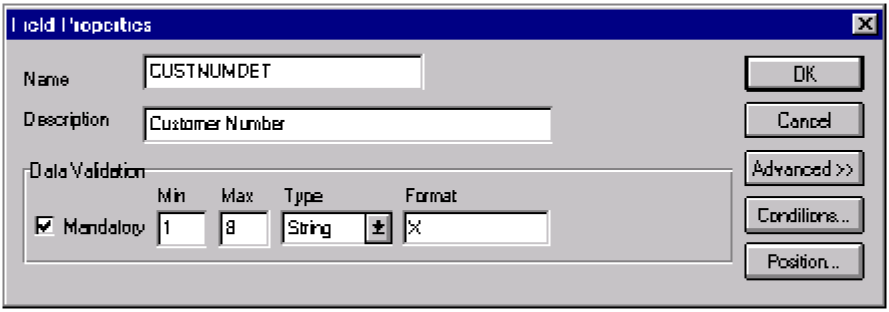
The Application Integration subsystem allows you to use selected information from the Gentran:Director Partner Editor in your maps via a standard rule. You can map information from your trading partner's profile in the Partner Editor to a selected element or field in the map. The information that you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

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

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

Complete the following steps to map the customer number from the partner definition:

1. Double-click the CUSTNUMDET field (in the PODETAIL record) to display the Field Properties dialogue.

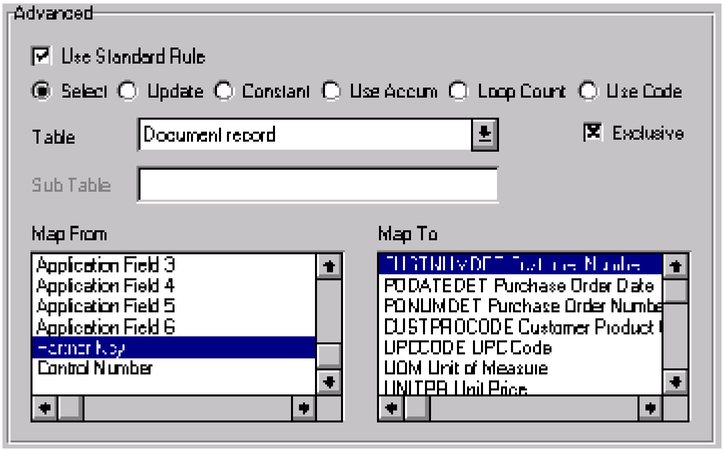


Name: CUSTNUMDET
Description: Customer Number
Data Validation
☑ Mandatory
Min: 1
Max: 8
Type: String
Format: X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. In the Advanced section, click the Use Standard Rule check box to display the rule options.
4. Verify that the Select radio button is selected.
5. From the Table drop-down list box, select “Document record”. This indicates that the system will update this field with selected information from the document record.
6. Click the Exclusive check box to indicate that a compliance error should be generated if the Select function does not find a valid entry.

7. From the Map From list box, choose “Partner Key”. This is the field that contains the customer number for this partner.
8. From the Map To list box, choose “CUSTNUMDET”. This is the field to which the information from the Partner Editor will be mapped.

   The Advanced section of the Field Properties dialogue should now look like this:



Advanced

☑ Use Standard Rule

● Select ○ Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table: Document record ☒ Exclusive

Sub Table:

Map From:
Application Field 3
Application Field 4
Application Field 5
Application Field 6
Partner Key
Control Number

Map To:
CUSTNUMDET Customer Number
PODATEDET Purchase Order Date
PONUMDET Purchase Order Numbe
CUSTPROCODE Customer Product
UPCCODE UPC Code
UOM Unit of Measure
UNITPR Unit Price

9. Click OK to add the standard rule to the CUSTNUMDET field.

PODATEDET

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

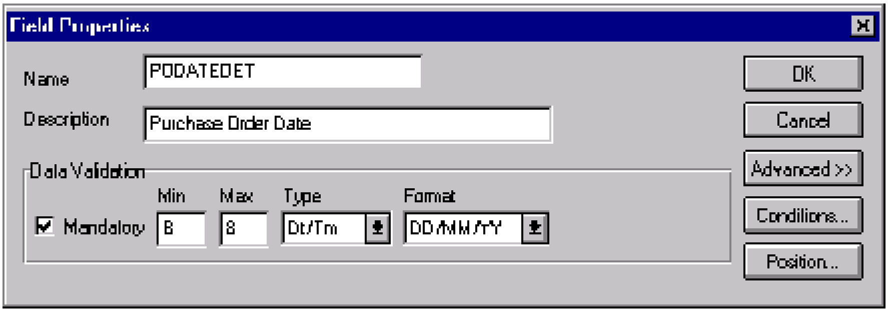
You can only map fields and elements that are at the same level. Therefore, you cannot map to a group from a segment that does not repeat. For example, you cannot map the date order placed by customer in the ORD segment (which repeats once) directly to your application PODATEDET field (which is included in the PODetailGroup that repeats 200,000 times).

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

Using an extended rule

Complete the following steps to create the extended rule:

1. Double-click the PODATEDET field to access the Field Properties dialogue.



Name: PODATEDET

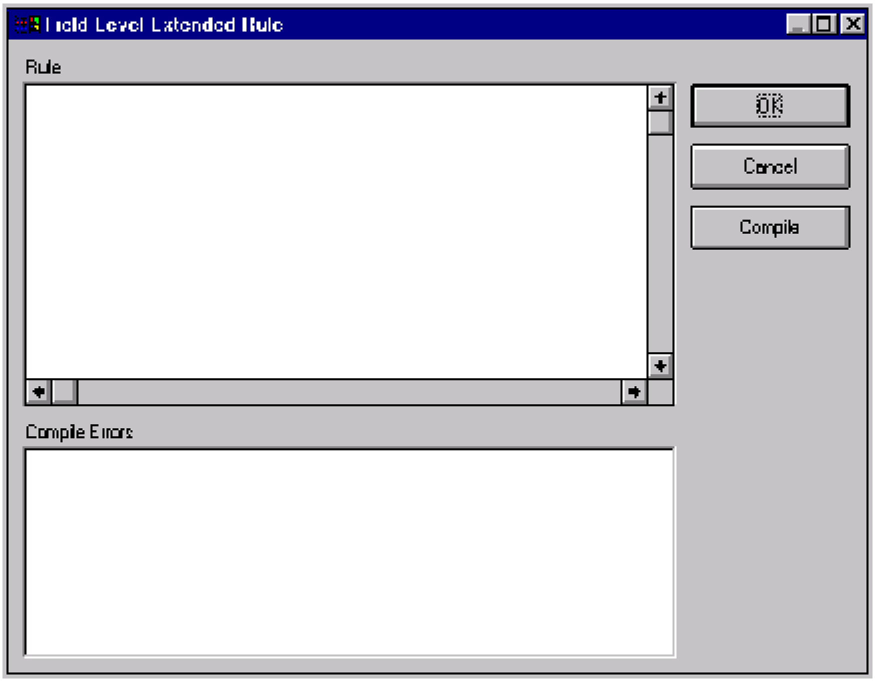
Description: Purchase Order Date

Data Validation

	Min	Max	Type	Format
☑ Mandatory	8	8	Dt/Tm	DD/MM/YY

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.

3. Click RULE... to display the Field Level Extended Rule dialogue.



4. In the Rule list box, type the following:

#PODATEDET = $ORDERS.#ORN2;

The “$ORDERS.#ORN2” is the addressing for the ORD ORNO ORD2 element. The system will execute this rule when it processes the PODATEDET field.

5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

   The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process are displayed in the Errors list box.

6. Click OK to add the extended rule to the PODATEDET field.
7. Click CLOSE to exit the Field Properties dialogue.

PONUMDET

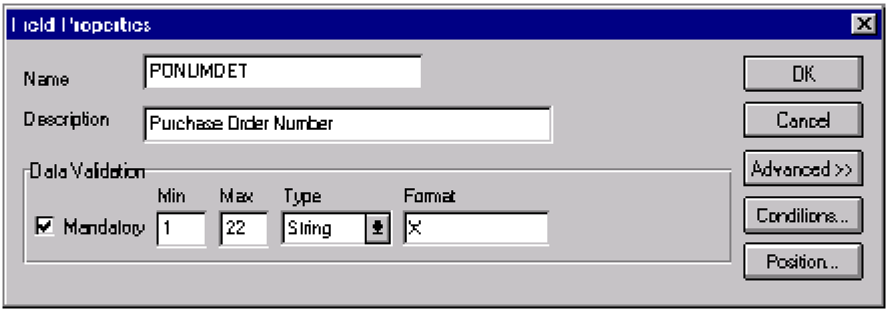
The ORD ORNO ORN0 element contains the customer’s order number. The ORD segment does not repeat. You need to map the information from the ORD ORNO ORN0 element to the PONUMDET field. However, the PONUMDET field is included in the PODetailGroup, which repeats 200,000 times.

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

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

Using an extended rule Complete the following steps to create the extended rule:

1. Double-click the PONUMDET field to access the Field Properties dialogue.



Name: PONUMDET

Description: Purchase Order Number

Data Validation

	Min	Max	Type	Format
☑ Mandatory	1	22	String	X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. Click RULE... to display the Field Level Extended Rule dialogue.
4. In the Rule list box, type the following:

   #PONUMDET = $ORDERS.#ORN0;

   The “$ORDERS.#ORN0” is the addressing for the ORD ORNO ORN0 element. The system will execute this rule when it processes the PONUMDET field.
5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

   The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process are displayed in the Errors list box.
6. Click OK to add the extended rule to the PONUMDET field.
7. Click CLOSE to exit the Field Properties dialogue.

CUSTPROCODE UPCCODE

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

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

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

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

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

My Value	Partner Value	Description	Reference Data
12345678901234	555666411	Large Dog Collar	Red
23456789012345	555666817	Spring Breeze Cat Litter	5 Kg. Bag
34567890123456	555666908	Small Beef Basted Rawhide Bones	Bag of 12

My Value	Partner Value	Description	Reference Data
45678901234567	555666321	X-Large Food Bowl	Black
56789012345678	555666287	FatKat Kitty Bed	Blue with White Trim
67890123456789	555666413	Catnip Mouse Toy	

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

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

Mapping the customer product code

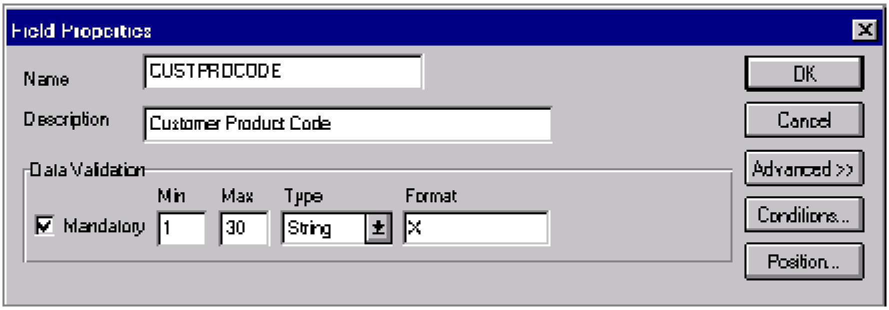
Complete the following steps to map the customer's item code:

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

Translating CUSTPROCODE to UPCCODE

Complete the following steps to translate your partner's customer product code to your UPC Code for each item your partner ordered:

1. Double-click the CUSTPROCODE field (in the PODETAIL record). The Field Properties dialogue is displayed.



Field Properties

Name: CUSTPROCODE

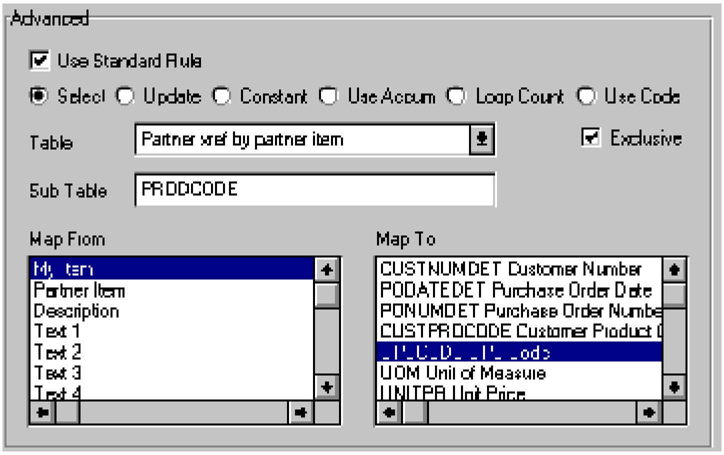
Description: Customer Product Code

Data Validation

☑ Mandatory Min: 1 Max: 30 Type: String Format: X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. In the Advanced section, click the Use Standard Rule check box to display the rule options.
4. Verify that the Select radio button is selected.
5. From the Table drop-down list box, select “Partner xref by partner item”. This indicates that you will be using a cross-reference table and translating by your partner’s item.
6. In the Sub Table field, type “PRODCODE”. This is the name of the cross-reference table.
7. Click the Exclusive check box to signal the system that if this translation fails, you want the system to generate an error.
8. From the Map From list box, choose “My Item”. This is the field from which the contents will be mapped.
9. From the Map To list box, choose “UPCCODE”. This is the field to which the information from the Partner Editor will be mapped.

   The Advanced section of the Field Properties dialogue should now look like this:



Advanced

☑ Use Standard Rule

● Select ○ Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table: Partner xref by partner item

☑ Exclusive

Sub Table: PRODCODE

Map From:
- My Item
- Partner Item
- Description
- Text 1
- Text 2
- Text 3
- Text 4

Map To:
- CUSTNUMDET Customer Number
- PODATEDET Purchase Order Date
- PONUMDET Purchase Order Numbe
- CUSTPRODCODE Customer Product C
- UPCCODE UPC Code
- UOM Unit of Measure
- UNITPR Unit Price

10. Click OK to map from the cross-reference table.

CONSUMERUNITS

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

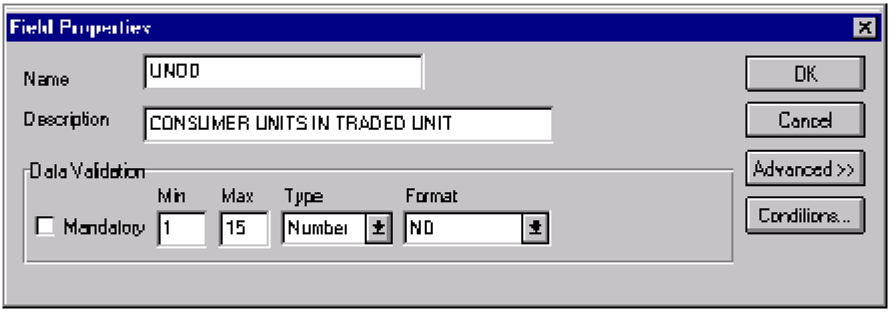
Therefore, you will need to change the properties of the OLD:2 UNOR UNO0 element to reflect the type and the format that you have determined that your partner will send you. Then, you can map the consumer units information directly from the OLD:2 UNOR UNO0 element to the CONSUMERUNITS application field.

Changing the properties of the OLD:2 UNOR UNO0 element

Complete the following steps to change the type and format of the OLD:2 UNOR UNO0 element:

1. Double-click the OLD:2 UNOR UNO0 element (Consumer Units in Traded Units) to display the Field Properties dialogue.
2. From the Type drop-down list box, select “Number” to indicate that this is a number element. The system will prompt you with a message that changing the field’s data type will invalidate its format. Click YES to acknowledge this message and proceed.
3. From the Format drop-down list box, select “N0” to indicate that this element will be formatted with an implicit decimal point and no decimal place.

   The Field Properties dialogue should now look like this:



Field Properties

Name: UNO0

Description: CONSUMER UNITS IN TRADED UNIT

Data Validation

□ Mandatory

Min: 1

Max: 15

Type: Number

Format: N0

4. Click OK to exit the Field Properties dialogue.

Mapping the consumer units

Complete the following steps to map the consumer units:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.

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

UOM

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

Complete the following steps to map the measure indicator:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the OLD:2 UNOR CD04 element (Measure Indicator). When you click the element, the cursor changes to a “Link Arrow”.
3. Click the application field UOM in the PODETAIL record.
4. A line is displayed, visually illustrating the link between the OLD:2 UNOR CD04 element and the UOM field.

UNITPR

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

Complete the following steps to map the unit cost price:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the OLD:2 OUCT OUC0 element (Cost Price). When you click the element, the cursor changes to a “Link Arrow”.
3. Click the application field UNITPR in the PODETAIL record.
4. A line is displayed, visually illustrating the link between the OLD:2 OUCT OUC0 element and the UNITPR field.

QTYORD

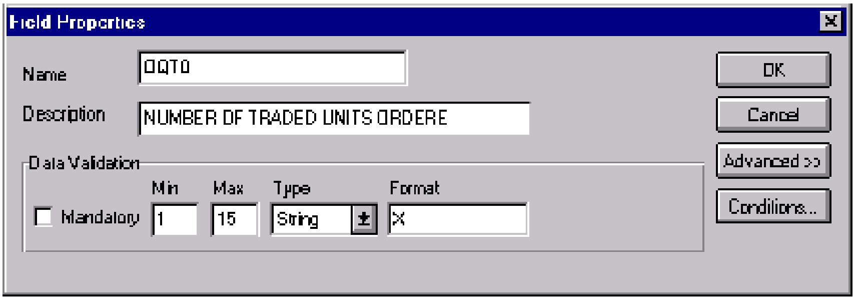
On the EDI side of the map, the quantity ordered information is contained in the OLD:2 OQTY OQT0 element (Number of Traded Units Ordered). The OLD:2 OQTY OQT0 element is defined as a string-type element in the standard. However, you have determined that your partner will send you a number-type element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

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

Changing the properties of the OLD:2 OQTY OQT0 element

Complete the following steps to change the type and format of the OLD:2 OQTY OQT0 element:

1. Double-click the OLD:2 OQTY OQT0 element (Number of Traded Units Ordered) to display the Field Properties dialogue:



Field Properties

Name: OQT0

Description: NUMBER OF TRADED UNITS ORDERE

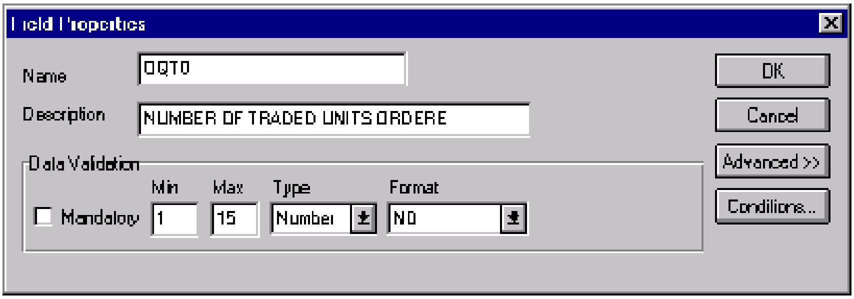
Data Validation

□ Mandatory Min: 1 Max: 15 Type: String Format: X

2. From the Type drop-down list box, select “Number” to indicate that this is a number element. The system will prompt you with a message that changing the field’s data type will invalidate its format. Click YES to acknowledge this message and proceed.

3. From the Format drop-down list box, select “N0” to indicate that this element will be formatted with an implicit decimal point and no decimal place.

   The Field Properties dialogue should now look like this:



Field Properties

Name: OQT0

Description: NUMBER OF TRADED UNITS ORDERE

Data Validation

□ Mandatory Min: 1 Max: 15 Type: Number Format: N0

4. Click OK to exit the Field Properties dialogue.

Mapping the quantity ordered

Complete the following steps to map the quantity ordered:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the OLD:2 OQTY OQT0 element (Number of Traded Units Ordered). When you click the element, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the application field QTYORD in the PODETAIL record.
4. A line is displayed, visually illustrating the link between the OLD:2 OQTY OQT0 element and the QTYORD field.

PODESC Record

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

CUSTNUMDESC

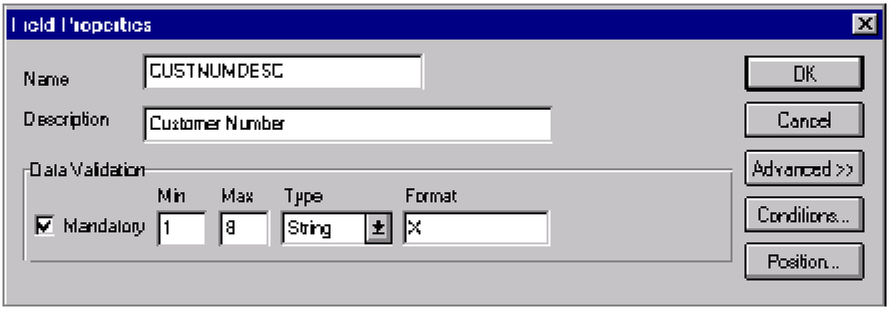
The Application Integration subsystem allows you to use selected information from the Gentran:Director Partner Editor in your maps via a standard rule. You can map information from your trading partner's profile in the Partner Editor to a selected element or field in the map. The information that you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

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

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

Complete the following steps to map the customer number from the partner definition:

1. Double-click the CUSTNUMDESC field (in the PODESC record) to display the Field Properties dialogue.



Name: CUSTNUMDESC

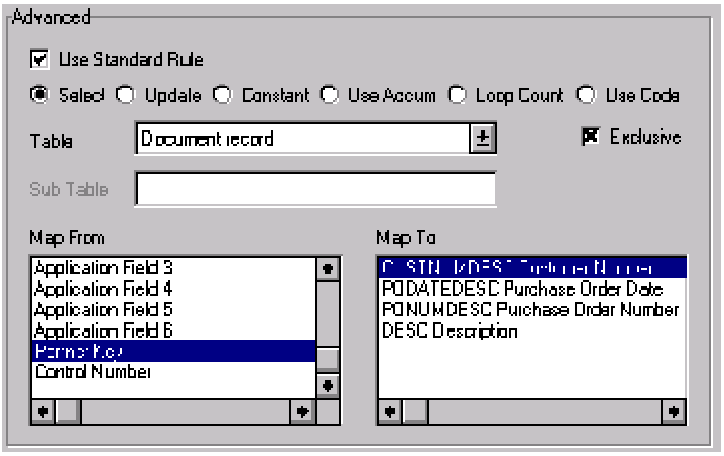
Description: Customer Number

Data Validation

☑ Mandatory Min: 1 Max: 8 Type: String Format: X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. In the Advanced section, click the Use Standard Rule check box to display the rule options.
4. Verify that the Select radio button is selected.
5. From the Table drop-down list box, select “Document record”. This indicates that the system will update this field with selected information from the document record.
6. Click the Exclusive check box to indicate that a compliance error should be generated if the Select function does not find a valid entry.
7. From the Map From list box, choose “Partner Key”. This is the field that contains the customer number for this partner.
8. From the Map To list box, choose “CUSTNUMDESC”. This is the field to which the information from the Partner Editor will be mapped.

The Advanced section of the Field Properties dialogue should now look like this:



Advanced

☑ Use Standard Rule

● Select ○ Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table: Document record ☒ Exclusive

Sub Table

Map From:
Application Field 3
Application Field 4
Application Field 5
Application Field 6
Partner Key
Control Number

Map To:
CUSTNUMDESC Customer Number
PODATEDESC Purchase Order Date
PONUMDESC Purchase Order Number
DESC Description

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

PODATEDESC

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

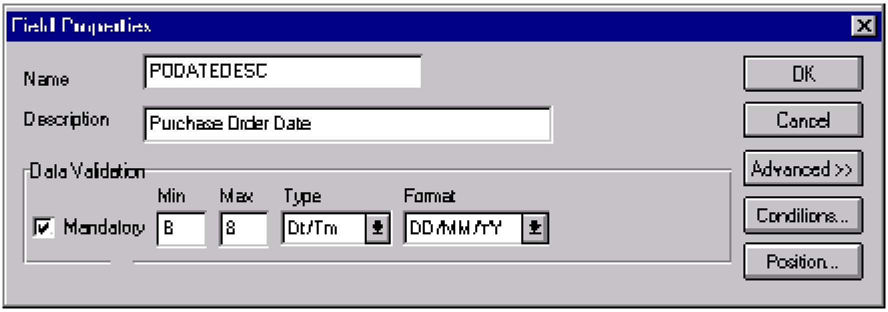
You can only map fields and elements that are at the same level. Therefore, you cannot map to a group from a segment that does not repeat. For example, you cannot map the date order placed by customer in the ORD segment (which repeats once) directly to your application PODATEDESC field (which is included in the PODetailGroup that repeats 200,000 times).

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

Using an extended rule

Complete the following steps to create the extended rule:

1. Double-click the PODATEDESC field to access the Field Properties dialogue.



Name: PODATEDESC

Description: Purchase Order Date

Data Validation

	Min	Max	Type	Format
☑ Mandatory	8	8	Dt/Tm	DD/MM/YY

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. Click RULE... to display the Field Level Extended Rule dialogue.
4. In the Rule list box, type the following:

   #PODATEDESC = $ORDERS.#ORN2;

   The “$ORDERS.#ORN2” is the addressing for the ORD ORNO ORD2 element. The system will execute this rule when it processes the PODATEDESC field.
5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

   The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Errors list box.
6. Click OK to add the extended rule to the PODATEDESC field.
7. Click CLOSE to exit the Field Properties dialogue.

## PONUMDESC

The ORD ORNO ORN0 element contains the customer's order number. The ORD segment does not repeat. You need to map the information from the ORD ORNO ORN0 element to the PONUMDESC field. However, the PONUMDESC field is included in the PODetailGroup, which repeats 200,000 times.

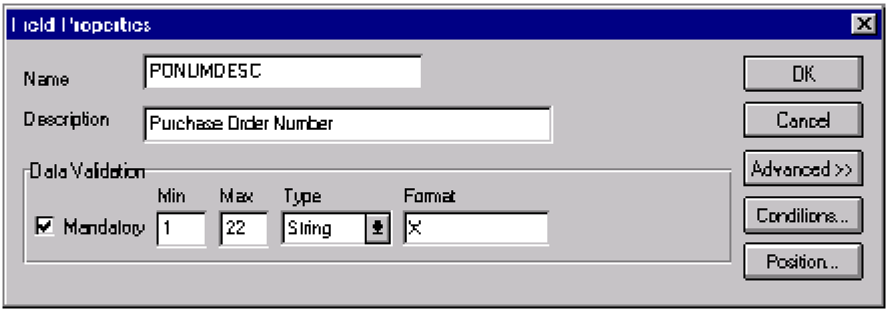
You can only map fields and elements that are at the same level. Therefore, you cannot map to a group from a segment that does not repeat. For example, you cannot map the customer's order number in the ORD segment directly to your application PONUMDESC field.

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

Using an extended rule

Complete the following steps to create the extended rule:

1. Double-click the PONUMDESC field to access the Field Properties dialogue.



Name: PONUMDESC
Description: Purchase Order Number
Data Validation
☑ Mandatory Min: 1 Max: 22 Type: String Format: X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. Click RULE... to display the Field Level Extended Rule dialogue.
4. In the Rule list box, type the following:

   #PONUMDESC = $ORDERS.#ORN0;

   The "$ORDERS.#ORN0" is the addressing for the ORD ORNO ORN0 element. The system will execute this rule when it processes the PONUMDESC field.
5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Errors list box.

6. Click OK to add the extended rule to the PONUMDESC field.
7. Click CLOSE to exit the Field Properties dialogue.

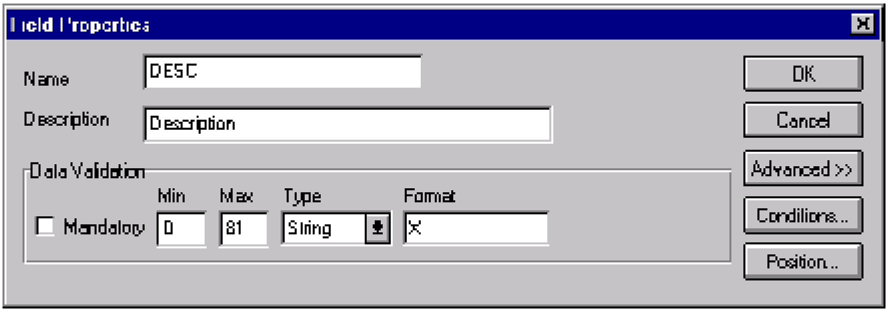
## DESC

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

Using an extended rule

Complete the following steps to create the extended rule:

1. Double-click the DESC field to access the Field Properties dialogue.



Field Properties

Name: DESC

Description: Description

Data Validation

☐ Mandatory

Min: 0

Max: 81

Type: String

Format: X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. Click RULE... to display the Field Level Extended Rule dialogue.
4. In the Rule list box, type the following:

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

   The “$OLD.#TDE0” is the addressing for the OLD:2 TDES TDE0 element. The “+” concatenates the character strings. The “ ” is a blank space, to distinguish the two concatenated lines of text. The “$OLD.#TDE1” is the addressing for the OLD:2 TDES TDE1 element. The system will execute this rule when it processes the PONUMDET field.

5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

   The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Errors list box.

6. Click OK to add the extended rule to the PONUMDESC field.
7. Click CLOSE to exit the Field Properties dialogue.

POSUMMARY Record

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

CUSTNUMSUM

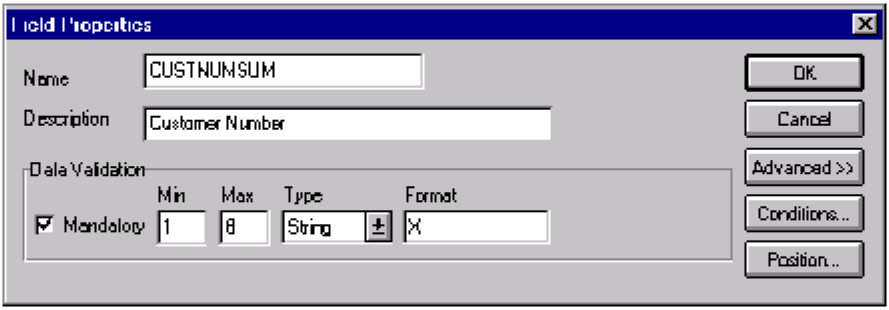
The Application Integration subsystem allows you to use selected information from the Gentran:Director Partner Editor in your maps via a standard rule. You can map information from your trading partner's profile in the Partner Editor to a selected element or field in the map. The information that you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

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

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

Complete the following steps to map the customer number from the partner definition:

1. Double-click the CUSTNUMSUM field (in the POSUMMARY record) to display the Field Properties dialogue.



Field Properties

Name: CUSTNUMSUM

Description: Customer Number

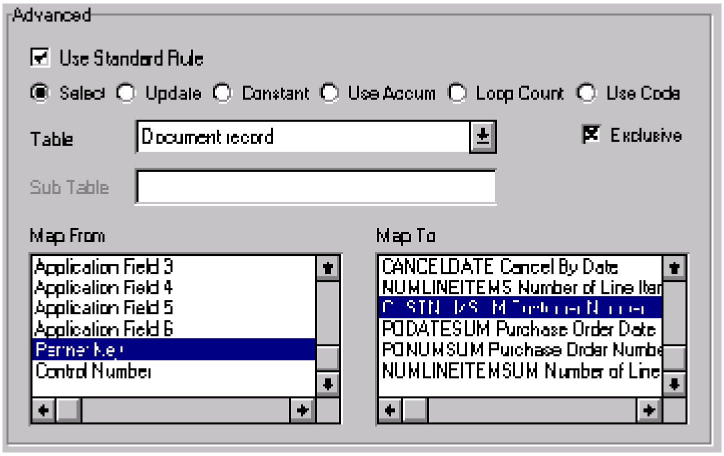
Data Validation

☑ Mandatory

Min	Max	Type	Format
1	8	String	X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. In the Advanced section, click the Use Standard Rule check box to display the rule options.
4. Verify that the Select radio button is selected.
5. From the Table list box, select "Document record" to indicate that the system will update this field with selected information from the document record.
6. Click the Exclusive check box to indicate that a compliance error should be generated if the Select function does not find a valid entry.
7. From the Map From list box, choose "Partner Key". This is the field that contains the customer number for this partner.
8. From the Map To list box, choose "CUSTNUMSUM". This is the field to which the information from the Partner Editor will be mapped.

The Advanced section of the Field Properties dialogue should now look like this:



Advanced

☑ Use Standard Rule

● Select ○ Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table: Document record ☒ Exclusive

Sub Table:

Map From:
Application Field 3
Application Field 4
Application Field 5
Application Field 6
Partner Key
Control Number

Map To:
CANCELDATE Cancel By Date
NUMLINEITEMS Number of Line Iten
CUSTNUMSUM Customer Number
PODATESUM Purchase Order Date
PONUMSUM Purchase Order Numbe
NUMLINEITEMSUM Number of Line

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

PODATESUM

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

Complete the following steps to map the date order placed by customer:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the ORD ORNO ORN2 element (Date Order Placed By Customer). When you click the element, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the application field PODATESUM in the POSUMMARY record.
4. A line is displayed, visually illustrating the link between the ORD ORNO ORN2 element and the PODATESUM field.

PONUMSUM

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

Complete the following steps to map the customer’s order number:

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

NUMLINEITEMSUM

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

Complete the following steps to map the number of line items:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the OTR LORD element (Lines Ordered),. When you click the element, the cursor changes to a “Link Arrow”.
3. Click the application field NUMLINEITEMSUM in the POSUMMARY record.
4. A line is displayed, visually illustrating the link between the OTR LORD element and the NUMLINEITEMSUM field.

After you complete the mapping operation for this final application field, the icons representing each of the application fields should be bisected by a red tick (except the UPCCODE field in the PODETAIL application record). The red tick indicates that a mapping operation has been performed against the field.

Finalising the Map

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

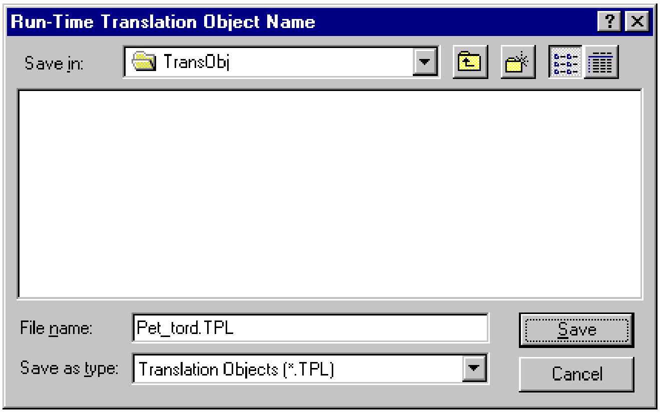
Compiling the Map

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

This translation object will need to be registered with the Gentran:Director system before you can use it. You will use the Compile function after the map is completed and saved.

Complete the following steps to compile a map and generate a translation object:

1. Select Save from the File menu to save the source map, prior to using the Compile function.
2. Select Compile from the File menu to display the Run-Time Translation Object Name dialogue.



Save in: TransObj

File name: Pet_tord.TPL

Save as type: Translation Objects (*.TPL)

3. The “File name” field will automatically contain “pet_tord.tpl”. This is the name of the translation object, using the default .TPL file extension and our recommendation that you name the translation object (.TPL file) the same filename (one to eight characters long) as you named the map (.MAP file). Preserving the same filename (with different file

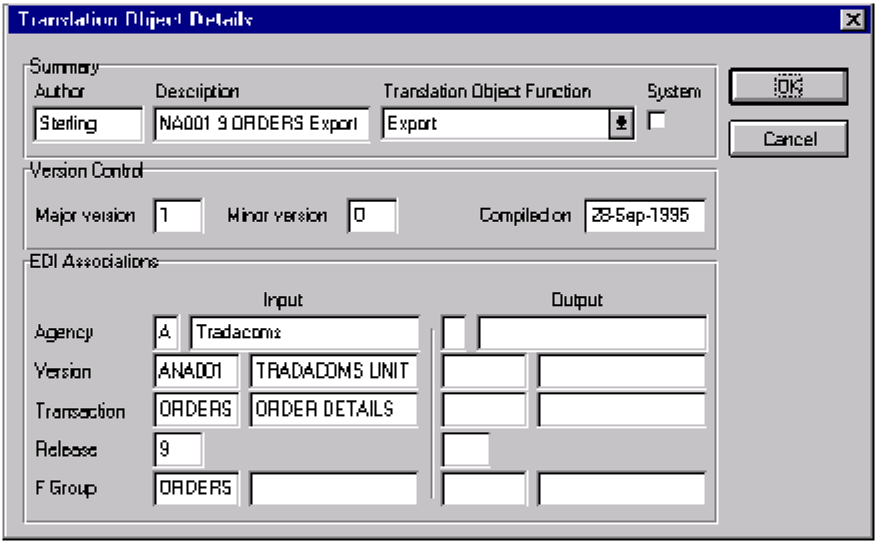
extensions) means that the relationship between the source map and the compiled translation object will remain evident.

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

4. Change the Drive and Folder where the compiled translation object will be stored, if necessary.

STOP **Do *NOT* store the compiled translation object in the GENTRAN Director\RegTransObj subfolder. This subfolder is reserved for storing a copy of each translation object you register with Gentran:Director.**

5. Click SAVE and the system will compile the map and generate a translation object. The Compile Error dialogue is displayed. Verify that no errors occurred and click OK to exit the dialogue. The date on which the translation object was compiled will be automatically loaded into the Compiled on field on the Translation Object Details dialogue.



Translation Object Details

Summary

Author: Sterling
Description: NA001 9 ORDERS Export
Translation Object Function: Export
System: □

Version Control

Major version: 1
Minor version: 0
Compiled on: 28-Sep-1995

EDI Associations

	Input		Output	
Agency	A	Tradacoms		
Version	ANA001	TRADACOMS UNIT		
Transaction	ORDERS	ORDER DETAILS		
Release	9			
F Group	ORDERS			

6. Select Save from the File menu to save the source map with the Compiled on date.

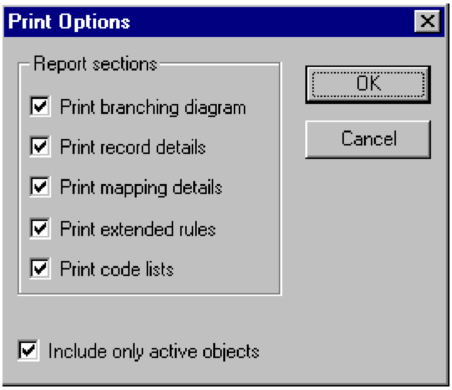
7. You must register this translation object with the Gentran:Director system before you can use it. See the Gentran:Director *User's Guide* for information on registering a translation object.

Printing the Report

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

Complete the following steps to print the Gentran:Director report:

1. Select Print from the File menu to access the Print Options dialogue.



Report sections

☑ Print branching diagram

☑ Print record details

☑ Print mapping details

☑ Print extended rules

☑ Print code lists

☑ Include only active objects

2. Accept the defaults on this dialogue (all options checked), so the report will include all sections and only the activated map components. Click OK and the Print dialogue is displayed.

3. Set the appropriate options. Click OK if you do *not* need to change Setup options, and the mapping report will be printed.

   Click SETUP... if you need to access printer setup information (to select a specific printer, paper orientation, or size and source of paper). The Print Setup dialogue is displayed.

4. Set the appropriate options. Click OK if you do *not* need to change Print Options (dithering, intensity control, or print true type as graphics), and you will return to the Print dialogue. Click OK on the Print dialogue and the mapping report will be printed.

   Click OPTIONS... if you need to access printer setup options information. The Options dialogue is displayed.

5. Set the appropriate options. Click OK to return to the Print Setup dialogue. Click OK on the Print Setup dialogue to return to the Print dialogue. Click OK on the Print dialogue and the mapping report will be printed.

Testing the Translation Object

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

Complete the following steps to test the inbound (export) translation object you just created:

1. Register the PET_TORD.TPL translation object with Gentran:Director. Please see the Gentran:Director *User's Guide* for more information on registering a translation object.

2. Import the PETZONE3.PAR partner relationship into Gentran:Director. Please see the Gentran:Director *User's Guide* for more information on importing a partner.

   The default full path for the partner file is: C:\Program Files\Sterling Commerce\GENTRAN Director\TUTORIAL\PETZONE3.PAR

3. Verify (in Partner Editor) that the PET T ANA001 9 ORDERS Export translation object is selected for the Inbound Relationship. Please see the Gentran:Director *User's Guide* for more information on selecting inbound translation objects.

4. Use the Process File option in Gentran:Director to process the data file (PETTSTT1.INT) through the translation object. The data file will be located in the TUTORIAL subfolder under the folder where Gentran:Director is installed.

   The default full path for the data file is: C:\Program Files\Sterling Commerce\GENTRAN Director\TUTORIAL\PETTSTT1.INT

5. After the message is translated, it will be located in the In Documents in Gentran:Director. Export the message to ensure that it was translated correctly. Please see the Gentran:Director *User's Guide* for more information on exporting files.

CHAPTER

3

Outbound Invoice (INVOIC)

This tutorial exercise defines the step-by-step tasks that are necessary for you to create a map that will be used to translate your application file to TRADACOMS invoices (INVOIC) that you send to your trading partner.

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

- Mapping analysis
- Creating the new map
- Activating EDI map components
- Verifying EDI delimiters
- Defining the application side of the map
- Mapping your data
- Finalising the map

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

Getting Started

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

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

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

Outbound Mapping Scenario

In this tutorial exercise, your company is MWT Manufacturing Co., which manufactures a wide variety of supplies for pet stores. Your company sells pet supplies directly to large pet supply retail chains. You need to create a map that will enable your company to translate your application file to the invoices that you will send to one of your trading partners, Pet Zone (a large pet supply retailer). After you ship an order to Pet Zone and generate the corresponding invoice, and once your application file is translated into an EDI invoice, your company can send the invoice to Pet Zone. You will invoice only *one* order per message. You will not be using settlement discounts.

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

Mapping Analysis

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

Regardless of whether you are translating data inbound or outbound, mapping analysis involves the following steps:

1. Analysing your application file format
2. Analysing your partner(s) EDI file
3. Correlating your application file format and the EDI file

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

Analysing Your Application File Format

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

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

The application file layout for the system import map in this tutorial is as follows:

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

The application file layout for the import map in this tutorial is as follows:

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

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Beginning Record (INVBEGIN) 1 per message					
BILLTOACCTBEG	X	Bill to Acct Number	String	1	9
DOCTYPBEG	X	Document Type	String	1	6
INVNUMBEG	X	Invoice Number	String	1	10
INVDATE	X	Invoice Date	Dt/Tm (YYMMDD)	6	6
PONUM		Purchase Order Number	String	0	20
PODATE		Purchase Order Date	Dt/Tm (YYMMDD)	6	6
SHIPTONAME		Ship To Name	String	0	35
SHIPTOADD1		Ship To Address Line 1	String	0	35
SHIPTOADD2		Ship To Address Line2	String	0	35
SHIPTOCITY		Ship To City	String	0	35
SHIPTOSTATE		Ship To State	String	0	35
SHIPTOPOSTALCODE		Ship To Postal Code	String	0	8
SHIPTOLOC		Ship To Store Location	String	0	8
BOLNUM		Bill of Lading Number	String	0	12
SHIPDATE		Ship Date	Dt/Tm (YYMMDD)	6	6
TERMSNETDUE		Terms New Due Days	Number (R0)	0	3
TERMSTEXT		Payment Terms Text	String	0	40
Comment Group (INVCommentGroup) 10 per message					
Invoice Comment Record (INVCOMMENT) 1 per message					
BILLTOACCTCMT	X	Bill To Acct Number	String	1	9
DOCTYPCMT	X	Document Type	String	1	6
INVNUMCMT	X	Invoice Number	String	1	10
MESSAGETEXT1		Message Text Line 1	String	0	40
MESSAGETEXT2		Message Text Line 2	String	0	40
MESSAGETEXT3		Message Text Line 3	String	0	40
MESSAGETEXT4		Message Text Line 4	String	0	40
Detail Record (INVDETAIL) up to 200,000 per message					
BILLTOACCTDET	X	Bill To Acct Number	String	1	9
DOCTYPDET	X	Document Type	String	1	6

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
INVNUMDET	X	Invoice Number	String	1	10
ITEMNUM	X	Item Number (Our Item)	Number (R0)	1	6
QTYSHIP	X	Quantity Shipped	Number (R0)	1	5
UNITPRICE	X	Unit Price	Number (R2)	1	7
UOM	X	Unit of Measure	String	1	4
CUSTPROCODE	X	Customer Product Code	String	1	10
UPCCODE	X	UPC Code	String	1	12
ITEMDESC		Item Description	String	0	30
VATCODE	X	VAT Rate Code	String	1	2
TEMPVATRATE		Temporary Storage VAT Percentage Rate	String	0	10
TEMPLEXC		Temporary Line Extended Cost	Number (R4)	0	14
Temporary Storage VAT Record (TEMPVATSTL) up to 10 per message					
BILLTOACCTTS	X	Bill To Acct Number	String	1	9
DOCTYPTS	X	Document Type	String	1	6
INVNUMTS	X	Invoice Number	String	1	10
SEQA		Sequence Number of STL	Number (R0)	0	10
VATRATE		VAT Rate	Number (R3)	0	6
VATRATECODE		VAT Code	String	0	1
NUMLINES		Number of Lines for VAT Code	Number (R0)	0	10
LVLA		Line Subtotal Amount (Before VAT)	Number (R2)	0	12
VATA		VAT Amount Payable	Number (R2)	0	12
Summary Record (INVSUMMARY) 1 per message					
BILLTOACCTSUM	X	Bill To Acct Number	String	1	9
DOCTYPSUM	X	Document Type	String	1	6
INVNUMSUM	X	Invoice Number	String	1	10
TOTALAMOUNT	X	Total Monetary Value Summary	Number (R2)	1	10

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

Each record (except the HEADER record, which is used for selecting the appropriate trading relationship) contains a three-field record key that ensures that the same identifying information is located at the beginning of each record. Your application file format contains a record key so that each record can be identified individually. The record key for this invoice consists of your partner's bill to account number, the document type, and the invoice number. The records are defined as follows:

- The HEADER (header) record contains the three-field key (partner identifier, application identifier, and application alias value), which is used for selecting the appropriate trading relationship. This header record is mandatory and occurs once for each invoice translated.
- The INVBEGIN (beginning) record contains the purchase order number and date, the shipping address information, bill of lading number, ship date, and terms information. The beginning record is mandatory and occurs once for each invoice.
- The INVCommentGroup (comment information) groups all the comment information for the invoice, including the message text. The INVCommentGroup is mandatory and occurs up to ten times per invoice translated.
- The INVCOMMENT (comment record) record contains the message text. The first comment record is mandatory and occurs once per invoice translated (once for each comment line).
- The INVDETAIL (detail) record contains line item number and description, quantity shipped, unit of measure, price, product code, and VAT calculation information. The detail record is mandatory and occurs up to 200,000 times per invoice translated (once for each line item invoiced).
- The TEMPVATSTL (temporary storage for VAT subtotals) record contains temporary storage fields to enable VAT subtotal calculations. The temporary storage record is conditional and occurs up to 20 times per invoice translated (once for each possible VAT code subtotal).
- The INVSUMMARY (summary) record contains the total invoice amount. The summary record is mandatory and occurs once per invoice translated.

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

Analysing the EDI File

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

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

After discussing with your partner what you will send, determining which segments and elements your partner requires, and reviewing the EDI standard, you can list the map components that you will need to make available for use (activate). When Gentran:Director generates the EDI side(s) of the map, the system includes all the groups, segments, composites, and elements that are defined by the standards agency for the version of the message you selected. The system activates all the groups, segments, composites, and elements that are defined as “mandatory” (must be present) by the standard. The system does not enable you to deactivate the mandatory groups, segments, composites, and elements.

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

For this tutorial, the map components that are required, but not currently active, are the following (the composite column is informational only, so you will know where to locate the appropriate element):

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

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

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

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

Also, we recommend that you set up a Document Name for each map, to make the identification of a message created by this map easier in Gentran:Director. The Document Name allows you to differentiate between messages in the document browsers in Gentran:Director.

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

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

Reconciling Your Application File and the EDI File

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

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

The six-field key method requires six EDI-specific fields in the header record: partner key, standard, version, message, release (for TRADACOMS only), and test/production status. The combination of these six fields defines a unique key that identifies the appropriate partner relationship. We recommend using this method because it is very flexible. You will typically use this method when you are defining your application from scratch and can easily add the EDI-specific fields that are not already present in the header record.

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

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

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

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

The following mapping operations (by application field) will be required for the system import map in this tutorial:

Field Name	Description	Mapping Operations
Header Record (HEADER) 1 per message		
PARTNER_ID	Partner Identifier	Select standard rule to match the EDI code in the Partner record (Partner by EDI Code)
APPL_ID	Application ID	Update standard rule to update the Application Field 1 on the document record
APPL_ALIAS_VALUE	Application Alias Value	Update standard rule to update the Application Field 2 on the document record

The following mapping operations (by application field) will be required for the import map in this tutorial:

Field Name	Description	Mapping Operations
Header Record (HEADER) 1 per message		
PARTNER_ID	Partner Identifier	No mapping required
APPL_ID	Application ID	No mapping required
APPL_ALIAS_VALUE	Application Alias Value	No mapping required
Beginning Record (INVBEGIN) 1 per message		
BILLTOACCTBEG	Bill to Acct Number	No mapping required
DOCTYPBEG	Document Type	No mapping required
INVNUMBEG	Invoice Number	Link to IRF INVN
INVDATE	Invoice Date	Link to IRF IVDT and IRF TXDT
PONUM	Purchase Order Number	Link to ODD:2 ORNO ORN0
PODATE	Purchase Order Date	Link to ODD:2 ORNO ORN2
SHIPTONAME	Ship To Name	Link to CLO CNAM
SHIPTOADD1	Ship To Address Line 1	Link to CLO CADD CAD0
SHIPTOADD2	Ship To Address Line2	Link to CLO CADD CAD1

Field Name	Description	Mapping Operations
SHIPTOCITY	Ship To City	Link to CLO CADD CAD2
SHIPTOSTATE	Ship To State	Link to CLO CADD CAD3
SHIPTOPOSTALCODE	Ship To Postal Code	Link to CLO CADD CAD4
SHIPTOLOC	Ship To Store Location	Link to CLO CLOC CLO1
BOLNUM	Bill of Lading Number	Link to ODD:2 DELN DEL0
SHIPDATE	Ship Date	Link to ODD:2 DELN DEL1
TERMSNETDUE	Terms New Due Days	Link to PYT PAYY PAY2
TERMSTEXT	Payment Terms Text	Link to PYT PAYT
Comment Group (INVCommentGroup) 10 per message		
Invoice Comment Record (INVCOMMENT) 1 per message		
BILLTOACCTCMT	Bill To Acct Number	No mapping required
DOCTYPCMT	Document Type	No mapping required
INVNUMCMT	Invoice Number	No mapping required
MESSAGETEXT1	Message Text Line 1	Link to DNA GNAR GNA0
MESSAGETEXT2	Message Text Line 2	Link to DNA GNAR GNA1
MESSAGETEXT3	Message Text Line 3	Link to DNA GNAR GNA2
MESSAGETEXT4	Message Text Line 4	Link to DNA GNAR GNA3
Detail Record (INVDETAIL) up to 200,000 per message		
BILLTOACCTDET	Bill To Acct Number	No mapping required
DOCTYPDET	Document Type	No mapping required
INVNUMDET	Invoice Number	No mapping required
ITEMNUM	Item Number (Our Item)	Link to ILD:2 SEQB
QTYSHIP	Quantity Shipped	Link to ILD:2 QTYI QTY0 and use accum standard rule to calculate extended price
UNITPRICE	Unit Price	Link to ILD:2 AUCT AUC0 and use accum standard rule to calculate extended price
UOM	Unit of Measure	Link to ILD:2 QTYI CD04:2 and ILD:2 AUCT CD04:3
CUSTPROCODE	Customer Product Code	Link to ILD:2 CPRO CPR1
UPCCODE	UPC Code	Link to ILD:2 SPRO SPR1
ITEMDESC	Item Description	Link to ILD:2 TDES TDE0

Field Name	Description	Mapping Operations
VATCODE	VAT Rate Code	Extract Division lookup from subtable “VATCODES” and Link to ILD:2 CD12
TEMPVATRATE	Temporary Storage VAT Percentage Rate	Standard rule to map Division lookup from subtable “VATCODES”
TEMPLEXC	Temporary Line Extended Cost	Use accum to extract extended price and Link to ILD:2 LEXC
Temporary Storage VAT Record (TEMPVATSTL) up to 10 per message		
BILLTOACCTTS	Bill To Acct Number	No mapping required
DOCTYPTS	Document Type	No mapping required
INVNUMTS	Invoice Number	No mapping required
SEQA	Sequence Number of STL	Link to STL SEQA:7
VATRATE	VAT Rate	Link to STL VATP:2
VATRATECODE	VAT Code	Link to STL CD12:2
NUMLINES	Number of Lines for VAT Code	Link to STL NRIL
LVLA	Line Subtotal Amount (Before VAT)	Link to STL LVLA, STL EVLA, and STL ASDA
VATA	VAT Amount Payable	Link to STL VATA
Summary Record (INVSUMMARY) 1 per message		
BILLTOACCTSUM	Bill To Acct Number	No mapping required
DOCTYPSUM	Document Type	No mapping required
INVNUMSUM	Invoice Number	No mapping required
TOTALAMOUNT	Total Monetary Value Summary	Link to TLR TPSI

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

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

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

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

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

Creating the System Import Map

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

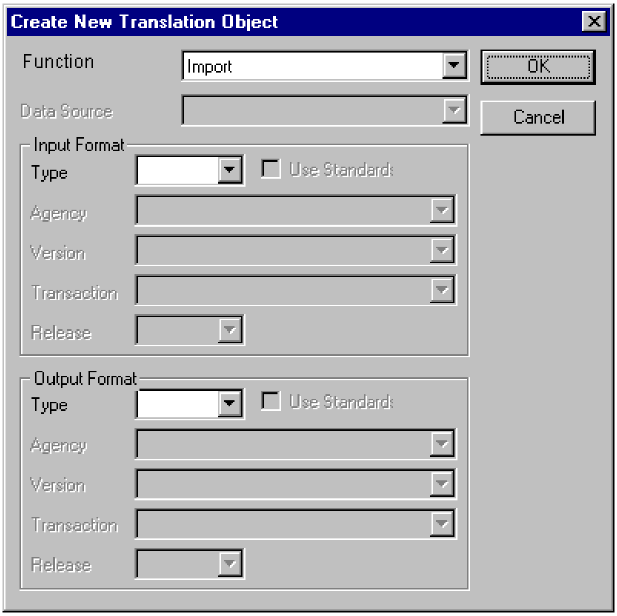
A system import map is used by the system to find the partner relationship for a message (flat file definition), to determine which import map will be used to translate the data. The system import map builds the key that the translator uses to find the partner relationship. The sole function of the system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

There are two ways to build the key in a system import map. The six-field key method requires six EDI-specific fields in the header record: partner key, standard, version, message, release (for TRADACOMS only), and test/production status. The combination of these six fields defines a unique key that identifies the appropriate partner relationship. We recommend using this method because it is very flexible. You will typically use this method when you are defining your application from scratch and can easily add the EDI-specific fields that are not already present in the header record. You will not be using the six-field key method for this tutorial, but you can refer to the *User's Guide* for more information about this method.

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

Complete the following steps to create the system import map:

1. Select New from the File menu to create a new map. The Create New Translation Object dialogue is displayed.



Create New Translation Object

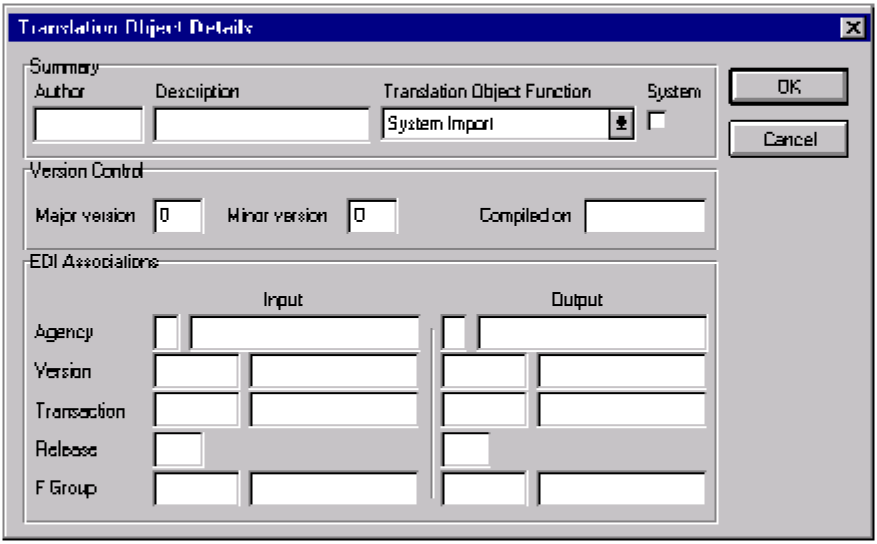
Function: Import

2. From the Translation Object Function drop-down list box, select “System Import”.
3. From the Input Format Type drop-down list box, select “Positional”.
4. From the Output Format Type drop-down list box, select “Positional”. The system will actually ignore the Output side of the map and only process the Input side.
5. Click OK to create the new system import map (this may take a few seconds). The new map will be displayed in the Gentran:Director - Application Integration Window.

   After you create the new map, it is important to define the details of the map and save it.

We recommend that you save your maps as often as possible, to mitigate the possible loss of additions or modifications in the event of a hardware or system error.

6. Select Save from the File menu to display the Translation Object Details dialogue.



Translation Object Function: System Import

System: □

Major version: 0

Minor version: 0

7. In the Author field, type your name. This field typically contains the name of the person who created the map.

8. In the Description field, type “System Import HDR”. For system import maps, we recommend that you type “System Import *”, where “*” is the identifier for this system import map. The Imports tab of the System Configuration program describes which system import translation object will be used to begin translation during the import process. Please see the *Administration Guide* for more information about how your system administrator will need to modify the system configuration program.

9. Verify that the Translation Object Function drop-down list box contains “System Import”.

10. Complete the Major and Minor fields to designate a version number for this translation object. Type “1” as the version number in the Major field and accept the “0” as the release number in the Minor field.

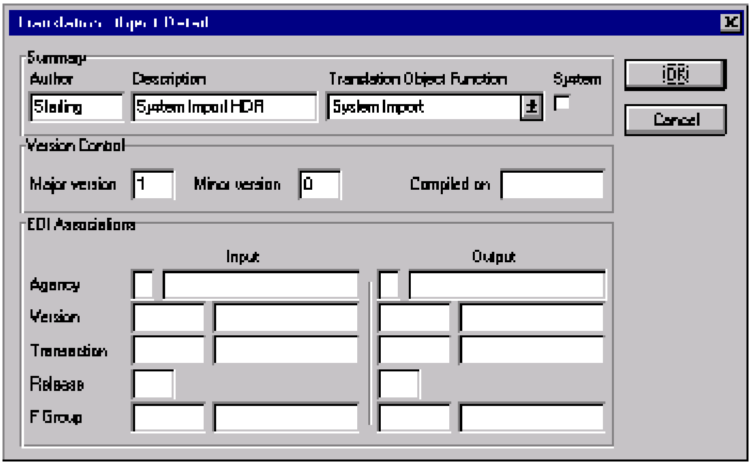
Whenever any major changes are made to your map, increase the version to the next real number (e.g., 2.0, 3.0). If you make a minor change to the map, increase the version by a percentage point (e.g., 2.1, 2.2).

11. The Input and Output fields (Agency, Version, Transaction, Release, and F Group) contain information only if the Input and/or Output sides

of the map are of EDI format. For the tutorial, *do not* change the information in these fields.

The system allows you to change the information in the Input and Output fields, but modifying this information will not alter the content of the map. An example of when you may wish to alter these fields is if you want the map to reflect a standard version that is not loaded on your system. You can change the version on this dialogue, and then physically alter the map to be compliant with that version.

The Translation Object Details dialogue should now look like this:



Translation Object Details

Summary

Author: Sterling

Description: System Import HDR

Translation Object Function: System Import

System: □

Version Control

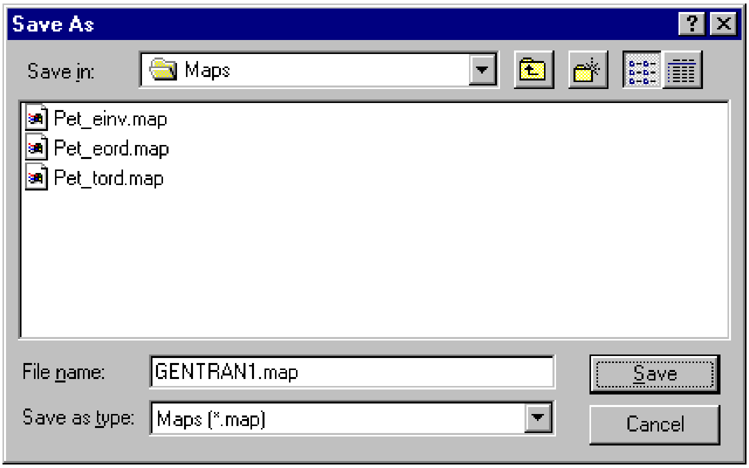
Major version: 1

Minor version: 0

Compiled on:

EDI Associations

12. Click OK to save the translation object details and access the Save As dialogue.



File name: GENTRAN1.map

Save as type: Maps (*.map)

13. The “Save in” drop-down list box should reflect the drive\folder where Gentran:Director is installed. The default is GENTRAN Director\Maps. If the current subfolder is not the correct one, select the appropriate subfolder. We recommend that you store maps in the map folder where Gentran:Director is installed.

14. In the “File name” field, type “HEADER_T.MAP” as the name of the map. You allocate a filename to a map when you save the map for the first time or use the Save As function to save a copy of the map under a different name. The name of the map must be one to eight unique characters that define the map in a meaningful way. The default file extension is “.MAP”. Using this extension distinguishes map files from other types of files, such as translation objects. The naming conventions in Gentran:Director are as follows:

    - The source map (map created in Gentran:Director) should be named using the file extension **.MAP** (e.g., HEADER_T.MAP).
    - The compiled translation object (translation object that is the result of compiling the source map) should be named using the same filename that was allocated to the source map, *but* with the file extension **.TPL** (e.g., HEADER_T.TPL).

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

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

15. Click SAVE to save the map.

After you complete each section of this tutorial or when you make any changes to the map, you should save the map by selecting Save from the File menu.

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

Creating the HEADER Record

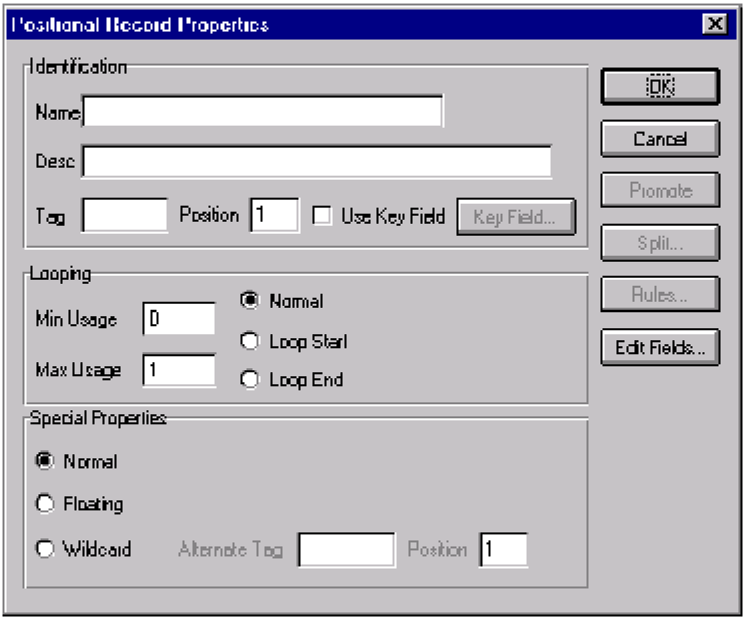
The only record that you will define for the system import map is the header record. The header record will be subordinate to the icon on the input side of the map.

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

Complete the following steps to create the header record (HEADER):

1. Click the Positional File icon on the input side of the map, to highlight it. Select Create Sub from the Edit menu. Select Record from the cascading menu.

The Positional Record Properties dialogue is displayed.



Name
Desc
Tag Position 1 □ Use Key Field
Min Usage 0
Max Usage 1
● Normal
○ Loop Start
○ Loop End
● Normal
○ Floating
○ Wildcard Alternate Tag Position 1

2. In the Name field, type “HEADER” as the record identification code.

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

3. In the Desc field, type “Header Record” as the description of the record. This field is used to provide a brief explanation of the record that will enable you to differentiate it from similar records.

4. In the Tag field, type “HDR” as the record identification code <TAG>. For example, a record is recognised by the system as the following:

   <TAG>[Field_1][Field_2]

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

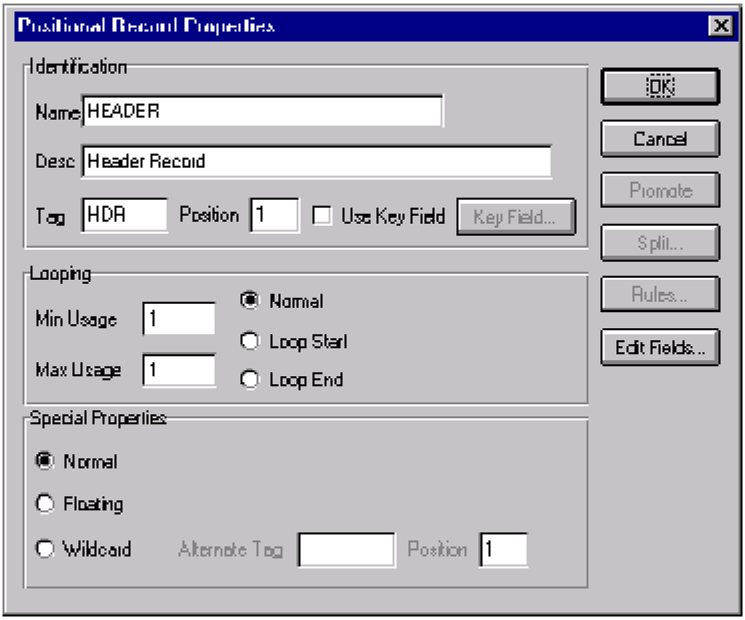
You must define the record tag on the Positional Record Properties dialogue for outbound maps, instead of defining fields with the purpose of explicitly containing the record tag.

5. In the Position field, accept the default of “1” for the starting column position of the tag in the data record.

6. In the Min Usage field, type “1” to make the header record mandatory. This is the minimum number of times the record must repeat.

If the Min Usage field contains “0” (zero), the record is conditional. If the Min Usage field contains “1” or greater, the record is mandatory.

The Positional Record Properties dialogue should now look like this:



Name HEADER

Desc Header Record

Tag HDR Position 1 □ Use Key Field

Min Usage 1

Max Usage 1

● Normal

○ Loop Start

○ Loop End

● Normal

○ Floating

○ Wildcard Alternate Tag Position 1

7. Click OK to create the HEADER record.

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

Creating the HEADER Fields

To use the preferred method of building the system import key, you need to define three fields for the header record (partner key, application identifier, and application alias value). These fields do not have to be in any order or sequence, but they must be part of the header record.

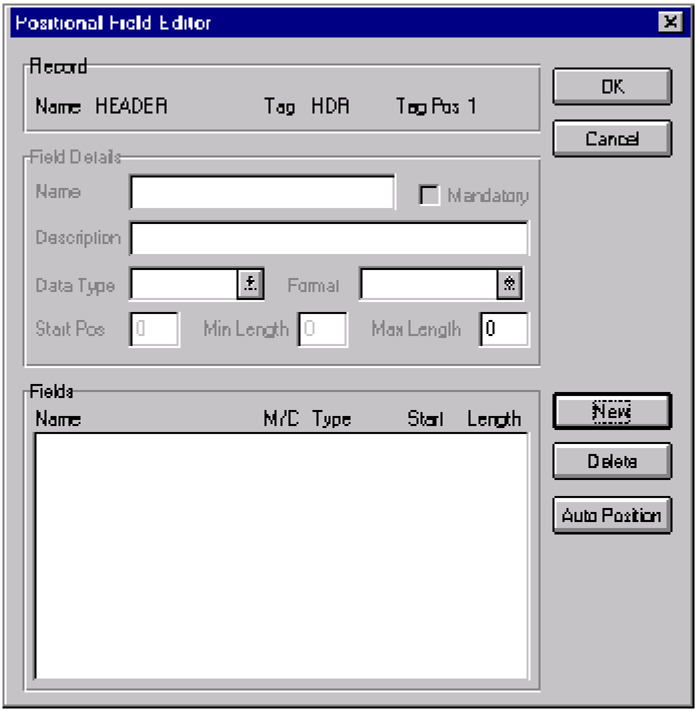
After you define the header record and fields, you can set up the mapping operations that will define the key that the translator uses to find the partner relationship.

The easiest way to add fields to a record is to use the Positional Field Editor. The fields that you need to create are listed in the following table. The name, mandatory status, description, data type, and field length for each field are listed in the table.

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

Complete the following steps to create the fields for the HEADER record:

1. Click the HEADER record with the right mouse button to access the shortcut menu.
2. Select Edit Fields... from the shortcut menu to access the Positional Field Editor dialogue.



Name HEADER Tag HDR Tag Pos 1

Start Pos 0 Min Length 0 Max Length 0

3. The field you are creating is the first field in the record, so click NEW. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.

4. In the Name field, type “PARTNER_ID”, which is the name of the first field in the HEADER record. Each field must have a unique name.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

5. Click the Mandatory check box to identify it as a required field. For the system import map, each field in the key is mandatory.
6. In the Description field, type “Partner Identifier” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7. From the Data Type drop-down list box, select “String” as the type of the field, to specify that this field must contain alphanumeric data.

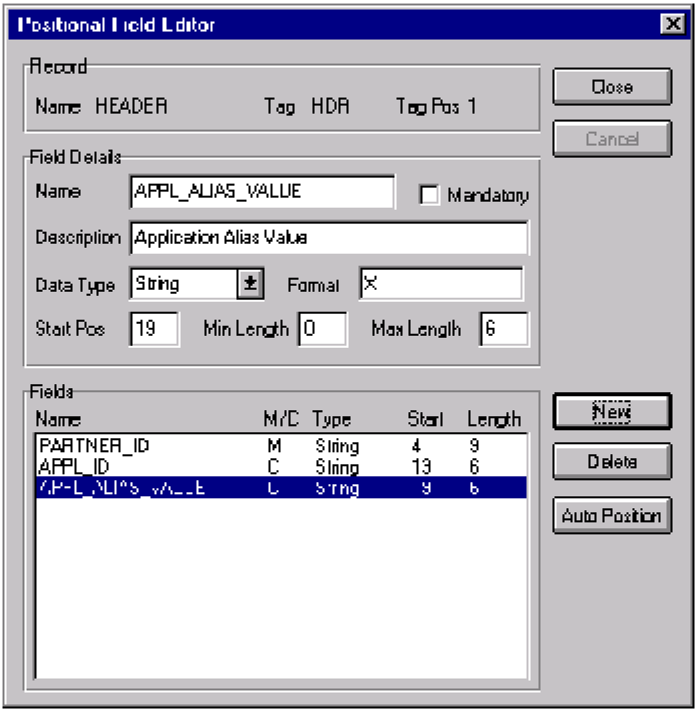
If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos field. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.

In this tutorial, you will use the alternative to specifying the start position of each field; you will add the fields sequentially in the record and then use the Auto Position function. Please see step 12 below for more information on the Auto Position function.

8. In the Min Length field, type “1” as the minimum field length.
9. In the Max Length field, type “9” as the maximum length of the field.
10. To add the field and to create the next field positioned after it, press **ENTER**. A new field with blank values ready for you to identify will be positioned below the field.
11. Create the rest of the fields according to the record layout table on page 117.
12. After you add the last field, click AUTO POSITION to automatically position the fields in the record. The Auto Position function will automatically calculate the start position in the record of each field based on the criteria that each field is positioned directly after the previous field and is of the length specified in the Max Length field. Click YES to acknowledge the warning message that fields will be sequenced in order.

STOP **Please note that it is only valid to use the Auto Position function if you define a record tag on the Positional Record Properties dialogue, and if you define every field in the record in the sequence that each field occurs (as you are doing in this tutorial).**

The Positional Field Editor dialogue should now look like the following:



Record

Name HEADER Tag HDR Tag Pos 1

Field Details

Name APPL_ALIAS_VALUE □ Mandatory

Description Application Alias Value

Data Type String Format X

Start Pos 19 Min Length 0 Max Length 6

Fields

Name	M/C	Type	Start	Length
PARTNER_ID	M	String	4	9
APPL_ID	C	String	13	6
APPL_ALIAS_VALUE	C	String	19	6

13. When you are finished adding fields to the HEADER record, click CLOSE.

Defining the Key

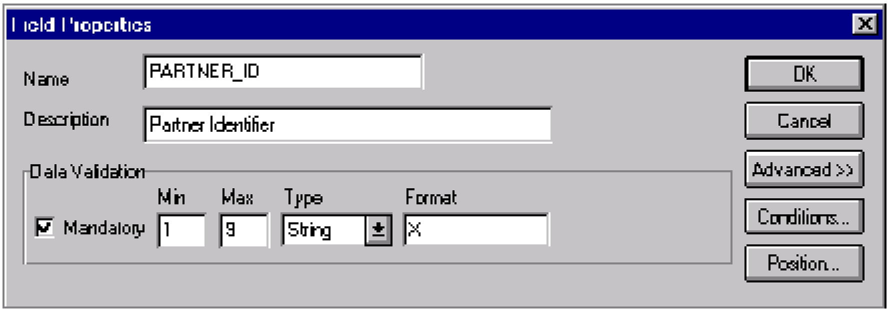
After you define the header record and fields, you can set up the mapping operations that will define the key that the translator uses to find the partner relationship.

Defining the Partner Key

Complete the following steps to define the partner key:

1. Double-click the HEADER record to display the fields you just created.

2. Double-click the PARTNER_ID field to display the Field Properties dialogue.



Field Properties

Name: PARTNER_ID

Description: Partner Identifier

Data Validation

☑ Mandatory

Min: 1

Max: 9

Type: String

Format: X

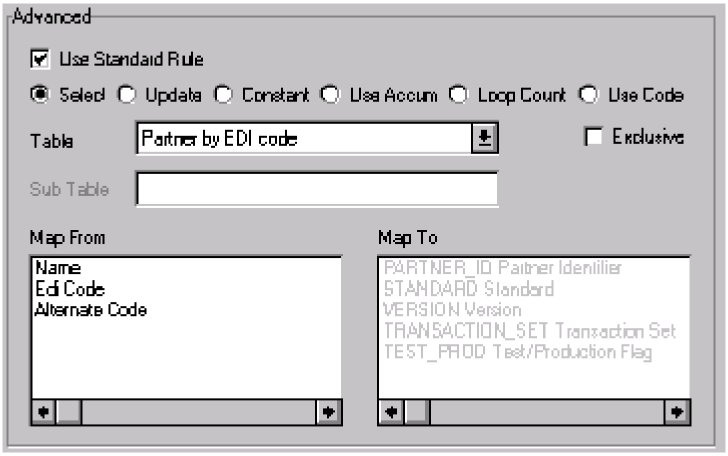
3. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.

4. In the Advanced section, click the Use Standard Rule check box to display the rule options.

5. Verify that the Select radio button is selected.

6. From the Table drop-down list box, select “Partner by EDI Code” (you could also select “Partner by Alternate code”, or “Partner by Partner Key”). This indicates that the system will look up the indicated partner key in the partner database and update this field.

   The Advanced section of the Field Properties dialogue should now look like this:



Advanced

☑ Use Standard Rule

● Select ○ Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table: Partner by EDI code

□ Exclusive

Sub Table:

Map From:
Name
Edi Code
Alternate Code

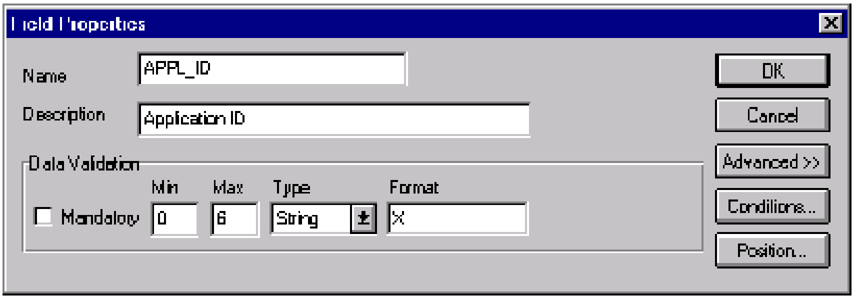
Map To:
PARTNER_ID Partner Identifier
STANDARD Standard
VERSION Version
TRANSACTION_SET Transaction Set
TEST_PROD Test/Production Flag

7. Click OK to add the standard rule to the partner key field.

Defining the Application ID Field

Complete the following steps to define the application ID field:

1. Double-click the application ID field to display the Field Properties dialogue.



Name: APPL_ID

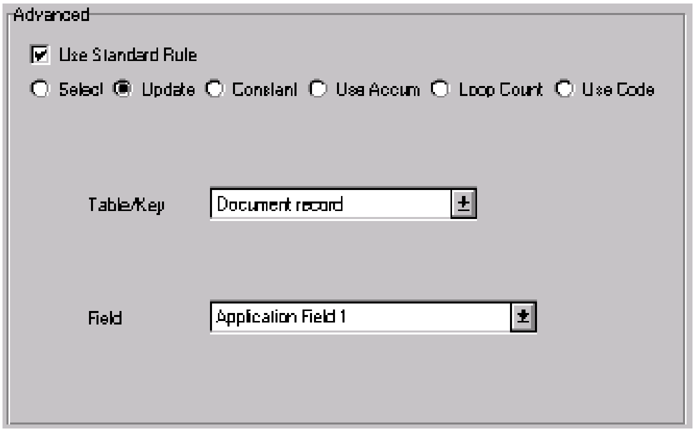
Description: Application ID

Data Validation

□ Mandatory Min: 0 Max: 6 Type: String Format: X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. In the Advanced section, click the Use Standard Rule check box to display the rule options.
4. Click the Update radio button.
5. From the Table/Key drop-down list box, choose "Document record". This indicates that you wish to store an item of data in the Gentran:Director tracking database for the document.
6. From the Field drop-down list box, choose "Application Field 1". This indicates that you are updating the first application field in the system buffer.

   The Advanced section should now look like this:



Advanced

☑ Use Standard Rule

○ Select ● Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table/Key: Document record

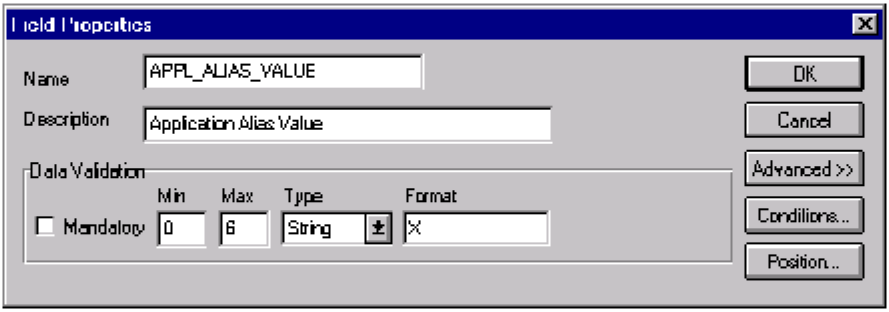
Field: Application Field 1

7. Click OK to add the standard rule to the application ID field.

Defining the Application Alias Value Field

Complete the following steps to define the application alias value field:

1. Double-click the application alias value field to display the Field Properties dialogue.



Name: APPL_ALIAS_VALUE

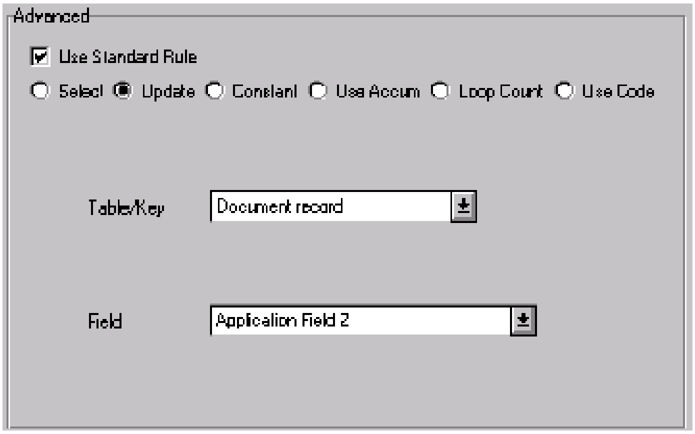
Description: Application Alias Value

Data Validation

□ Mandatory; Min: 0; Max: 6; Type: String; Format: X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. In the Advanced section, click the Use Standard Rule check box to display the rule options.
4. Click the Update radio button.
5. From the Table/Key drop-down list box, choose “Document record”. This indicates that you wish to store an item of data in the Gentran:Director tracking database for the document.
6. From the Field drop-down list box, choose “Application Field 2”. This indicates that you are updating the second application field in the system buffer.

   The Advanced section should now look like this:



Advanced

☑ Use Standard Rule

○ Select ● Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table/Key: Document record

Field: Application Field 2

7. Click OK to add the standard rule to the application alias value field.

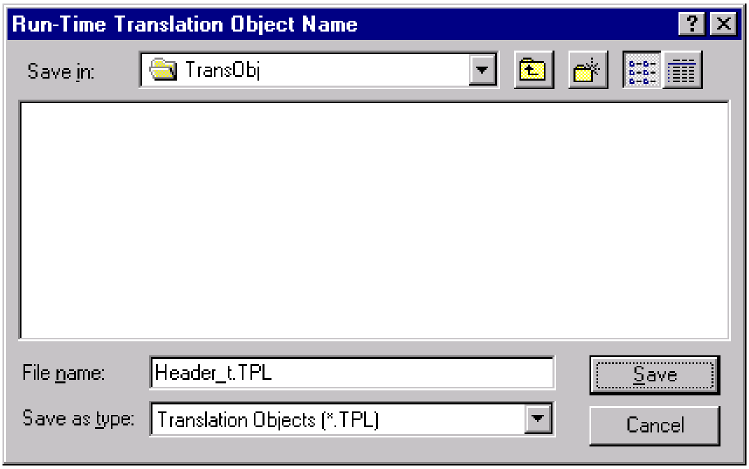
Compiling the Map

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

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

Complete the following steps to compile a map and generate a translation object:

1. Select Save from the File menu to save the source map, prior to using the Compile function.
2. Select Compile from the File menu to display the Run-Time Translation Object Name dialogue.



File name: Header_t.TPL

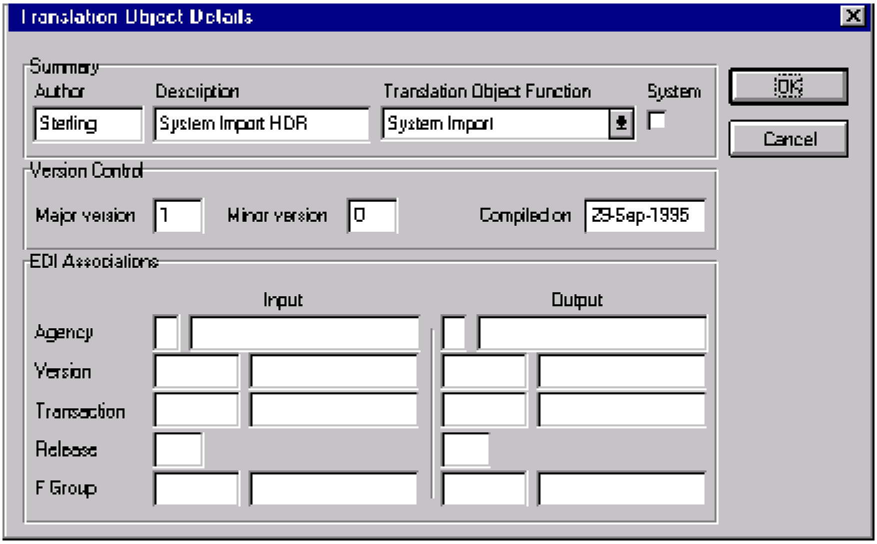
Save as type: Translation Objects (*.TPL)

3. The “File name” field will automatically contain “header_t.tpl”. This is the name of the translation object, using the default .TPL file extension and our recommendation that you name the translation object (.TPL file) the same filename (one to eight characters long) as you named the map (.MAP file). Preserving the same filename (with different file extensions) means that the relationship between the source map and the compiled translation object will remain evident.

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

4. Change the Drive and Folder where the compiled translation object will be stored, if necessary.
5. Click SAVE and the system will compile the map and generate a translation object. The Compile Error dialogue is displayed. Verify that no errors occurred and click OK to exit the dialogue.

   The date on which the translation object was compiled will be automatically loaded into the Compiled on field on the Translation Object Details dialogue.



Translation Object Details

Summary

Author: Sterling

Description: System Import HDR

Translation Object Function: System Import

System: □

Version Control

Major version: 1

Minor version: 0

Compiled on: 29-Sep-1995

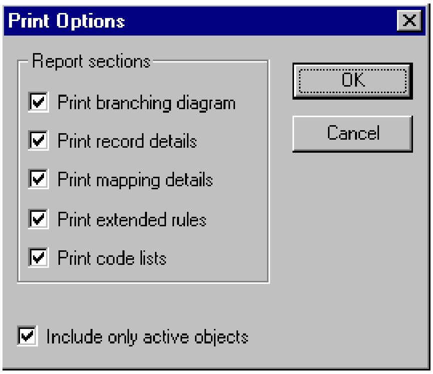
6. Select Save from the File menu to save the source map with the Compiled on date.
7. You must register this translation object with the Gentran:Director system before you can use it. See the Gentran:Director documentation for information on registering a translation object.

Printing the Report

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

Complete the following steps to print the Gentran:Director report:

1. Select Print from the File menu to access the Print Options dialogue.



Report sections

☑ Print branching diagram

☑ Print record details

☑ Print mapping details

☑ Print extended rules

☑ Print code lists

☑ Include only active objects

2. Accept the defaults on this dialogue (all options checked), so the report will include all sections and only the activated map components. Click OK and the Print dialogue is displayed.

3. Set the appropriate options. Click OK if you do *not* need to change Setup options, and the mapping report will be printed.

   Click SETUP... if you need to access printer setup information (to select a specific printer, paper orientation, or size and source of paper). The Print Setup dialogue is displayed.

4. Set the appropriate options. Click OK if you do *not* need to change Print Options (dithering, intensity control, or print true type as graphics), and you will return to the Print dialogue. Click OK on the Print dialogue and the mapping report will be printed.

   Click OPTIONS... if you need to access printer setup options information. The Options dialogue is displayed.

5. Set the appropriate options. Click OK to return to the Print Setup dialogue. Click OK on the Print Setup dialogue to return to the Print dialogue. Click OK on the Print dialogue and the mapping report will be printed.

Testing the Translation Object

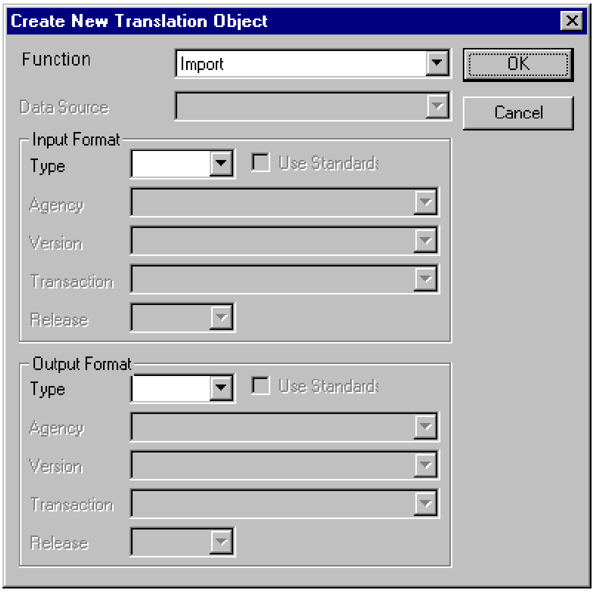
After you compile the map, print and verify the report, and register the translation object with Gentran:Director, you should test the translation object to verify that the data is translated correctly. You will test the system import translation object (HEADER_T.TPL) when you test the import map (please see page 211).

Creating the Import Map

The first task in creating a new map is to define the format that will be used for the input and output sides of the map. In this example, you are creating an import map so Gentran:Director can translate your application file to an invoice that you are sending to your partner. Therefore, the input side of the map needs to reflect your application file. The output side of the map needs to reflect the TRADACOMS ANA001 invoice (INVOIC) that you will send to your partner. Instead of defining the INVOIC message to the system yourself, you want Gentran:Director to use the standards database on your system to build the invoice according to the standard.

Complete the following steps to create the import map for the outbound invoice:

1. Select New from the File menu to display the Create New Translation Object dialogue.



Create New Translation Object

Function: Import

Data Source

Input Format

Type ☐ Use Standards

Agency

Version

Transaction

Release

Output Format

Type ☐ Use Standards

Agency

Version

Transaction

Release

2. In the Translation Object Function drop-down list box, accept the default of “Import”.

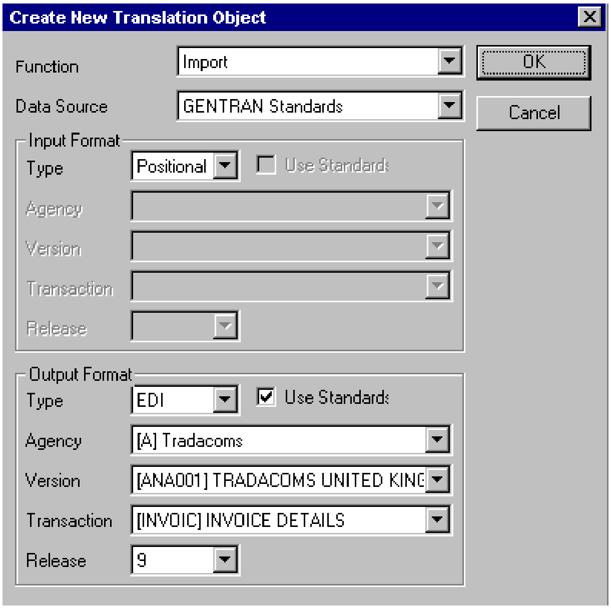
   Complete the Input Format section of the dialogue. This is the format of the data that will be translated by the Gentran:Director system.

3. From the (Input Format) Type drop-down list box, select “Positional” as the type of format for the output data. Gentran:Director uses the term “positional” to describe any flat file that you create.

Now you need to complete the Output Format section of the dialogue.

4. From the (Output Format) Type drop-down list box, select “EDI” as the type of format for the output data.
5. Click the Use Standards check box, because you selected “EDI” in the Type field and because you want the system to use the standards database on your system to build the invoice according to the standard.
6. To select a data source name from which Application Integration will access EDI standards, select one from the Data Source drop-down list box. This DSN will be prompted the next time you create a map.
7. From the Agency drop-down list box, select “[A] Tradacoms” as the EDI standard you will be using for the input side of the map.
8. From the Version drop-down list box, select “[ANA001] TRADACOMS UNITED KINGDOM” as the version of the standard you will use. The versions that are available depend on which standard you selected.
9. From the Transaction drop-down list box, select “[INVOIC] INVOICE DETAILS” as the transaction set (message) you will use. The transactions that are available depend on which standard and version you selected.
10. From the Release drop-down list box, select “9” as the message version you will use. The message versions that are available depend on which message (transaction) you selected.

The Create New Translation Object dialogue should now look like this:



Function: Import

Data Source: GENTRAN Standards

Input Format

Type: Positional ☐ Use Standards

Agency:

Version:

Transaction:

Release:

Output Format

Type: EDI ☑ Use Standards

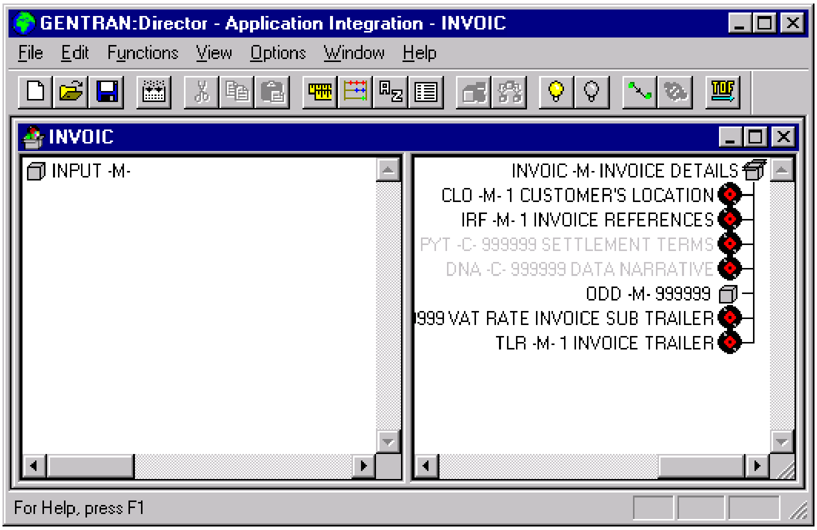
Agency: [A] Tradacoms

Version: [ANA001] TRADACOMS UNITED KINC

Transaction: [INVOIC] INVOICE DETAILS

Release: 9

11. Click OK to generate the new map (this may take a few seconds). The new map will be displayed in the Gentran:Director - Application Integration Window.



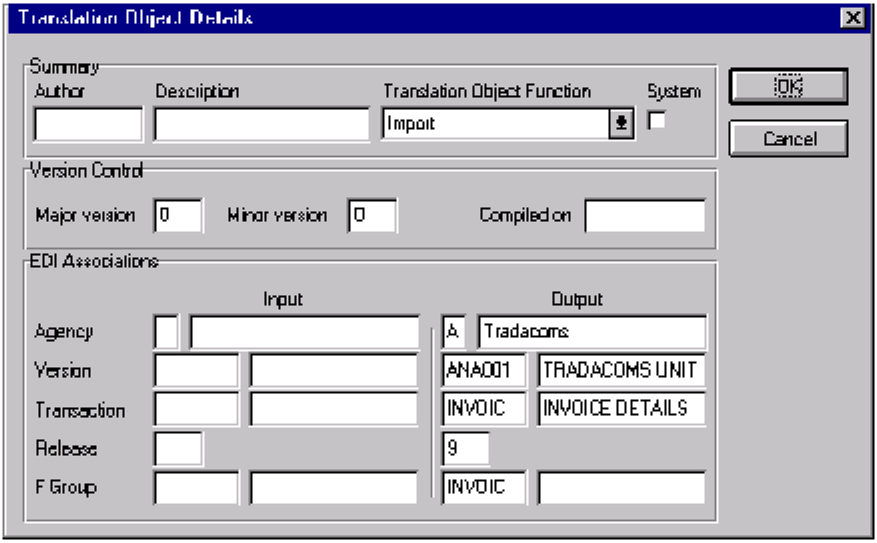
INPUT -M-

INVOIC -M- INVOICE DETAILS
CLO -M- 1 CUSTOMER'S LOCATION
IRF -M- 1 INVOICE REFERENCES
PYT -C- 999999 SETTLEMENT TERMS
DNA -C- 999999 DATA NARRATIVE
ODD -M- 999999
999 VAT RATE INVOICE SUB TRAILER
TLR -M- 1 INVOICE TRAILER

After you create a new map, it is important to define the details of the map and save it. The Save function writes (saves) the map with current focus to disk.

We recommend that you save your maps as often as possible, to mitigate the possible loss of additions or modifications in the event of a hardware or system error.

12. Select Save from the File menu to display the Translation Object Details dialogue.



Summary

Author:

Description:

Translation Object Function: Import

System: □

Version Control

Major version: 0

Minor version: 0

Compiled on:

EDI Associations

	Input		Output	
Agency			A	Tradacoms
Version			ANA001	TRADACOMS UNIT
Transaction			INVOIC	INVOICE DETAILS
Release			9	
F Group			INVOIC	

13. In the Author field, type your name. This field typically contains the name of the person who created the map.

14. In the Description field, type “PET ANA001 9 INVOIC Import”. This field must be unique because it is used by the system to identify the map. We recommend that you use the following identifying characteristics: 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 (e.g., Import, Export, etc.). For example, “PET ANA001 9 INVOIC Import” is the description of a map used with partner Pet Zone, for a TRADACOMS version ANA001 invoice (INVOIC) (release 9), that is sent outbound (Import).

15. Verify that the type of map shown in the Translation Object Function drop-down list box is “Import”.

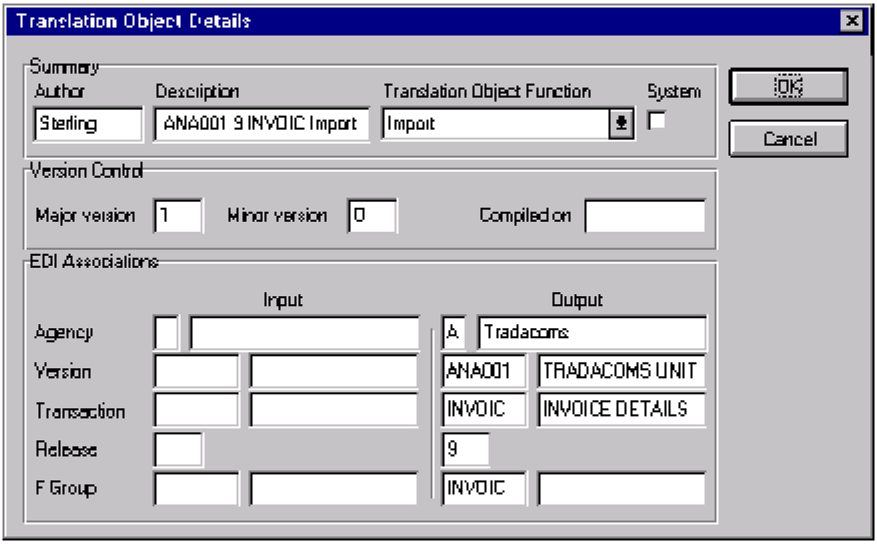
16. Complete the Major and Minor fields to designate a version number for this translation object. Type “1” as the version number in the Major field and accept the “0” as the release number in the Minor field.

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

17. The Input and Output fields (Agency, Version, Transaction, Release, and F Group) contain information only if the Input and/or Output sides of the map are of EDI format. For the tutorial, *do not* change the information in these fields.

The system allows you to change the information in the Input and Output fields, but modifying this information will not alter the content of the map. An example of when you may wish to alter these fields is if you want the map to reflect a standard version that is not loaded on your system. You can change the version on this dialogue, and then physically alter the map to be compliant with that version.

The Translation Object Details dialogue should now look like this:



Summary

Author: Sterling

Description: ANA001 9 INVOIC Import

Translation Object Function: Import

System: □

Version Control

Major version: 1 Minor version: 0 Compiled on:

EDI Associations

	Input		Output	
Agency			A	Tradacoms
Version			ANA001	TRADACOMS UNIT
Transaction			INVOIC	INVOICE DETAILS
Release			9	
F Group			INVOIC	

18. Click OK to save the translation object details and access the Save As dialogue.

19. The “Save in” drop-down list box should reflect the drive\folder where Gentran:Director is installed. The default is GENTRAN Director\Maps. If the current subfolder is not the correct one, select the appropriate subfolder.

20. In the “File name” field, type “PET_TINV.MAP” as the name of the map. You allocate a filename to a map when you save the map for the first time or use the Save As function to save a copy of the map under a different name. The name of the map must be one to eight unique characters that define the map in a meaningful way. For example, “PET_TINV.MAP” is the name of a map for an TRADACOMS invoice (INVOIC) for partner Pet Zone.

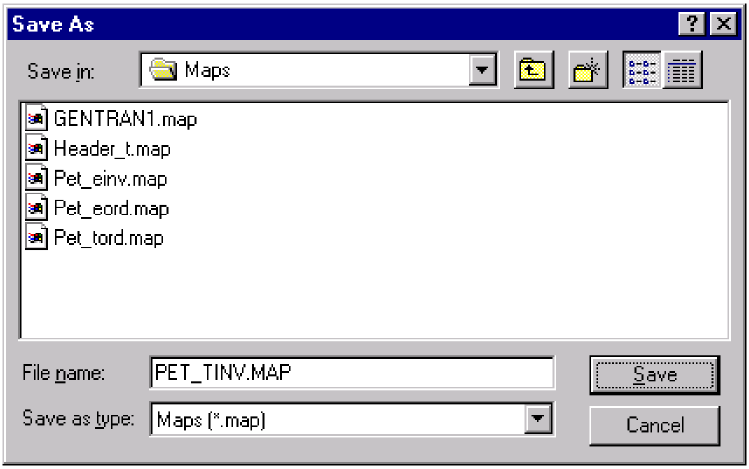
    The default file extension for maps is .MAP. Using this extension distinguishes map files from other types of files, such as translation objects. The naming conventions in Gentran:Director are as follows:

    - The source map (map created in Gentran:Director) should be named using the file extension **.MAP** (e.g., PET_TINV.MAP).
    - The compiled translation object (translation object that is the result of compiling the source map) should be named using the same filename that was allocated to the source map, *but* with the file extension **.TPL** (e.g., PET_TINV.TPL).

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

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

The Save As dialogue should now look like this:



GENTRAN1.map
Header_t.map
Pet_einv.map
Pet_eord.map
Pet_tord.map

File name: PET_TINV.MAP

Save as type: Maps (*.map)

21. Click SAVE to save the map.

After you complete each section of this tutorial or when you make any changes to the map, you should save the map by selecting Save from the File menu.

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

Activating EDI Map Components

The map components for an EDI File are the following:

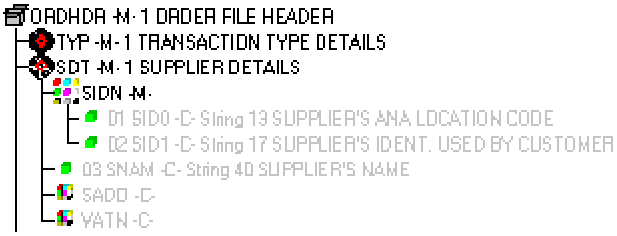
EDI File
Segment
Loop (Group)
Composite Element
Element

The map components for a Positional File are the following:

Positional File
Record
Loop (Group)
Field

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

When Gentran:Director generates the EDI side(s) of the map, the system includes all the groups, segments, composites, and elements that are defined by the standards agency for the version of the message you selected. The system activates all the groups, segments, composites, and elements that are defined as “mandatory” (must be present) by the standard. The system does not enable you to deactivate the mandatory groups, segments, composites, and elements. By default, Gentran:Director displays active map components with a black fount, and inactive map components with a grey fount (dimmed). In the example below, the ORDHDR group, TYP segment, SDT segment, and the SIDN composite are mandatory, so they are already activated by the system. You also need to use the SDT SIDN SID0 element (a conditional element), so you will need to activate it.



```
ORDHDR -M- 1 ORDER FILE HEADER
  TYP -M- 1 TRANSACTION TYPE DETAILS
  SDT -M- 1 SUPPLIER DETAILS
    SIDN -M-
      01 SID0 -C- String 13 SUPPLIER'S ANA LOCATION CODE
      02 SID1 -C- String 17 SUPPLIER'S IDENT. USED BY CUSTOMER
    03 SNAM -C- String 40 SUPPLIER'S NAME
    SADD -C-
    VATN -C-
```

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

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

1. Double-click the groups, segments, and composites that are listed in the table below to open them. These are the segments and elements that you will need to activate. The composite (Comp.) column is provided for your information, so you will know where to locate the appropriate element(s).

Opening all the necessary groups, segments, and composites before you start to activate allows you to scroll down the map and activate the segments and elements sequentially.

Group	Segment	Group	Segment	Comp.	Element
	CLO (already active)			CLOC	CLO1
					CNAM
				CADD	CAD0 CAD1 CAD2 CAD3 CAD4
	PYT				PAYT
				PAYY	PAY2
	DNA			GNAR	GNA0 GNA1 GNA2 GNA3
ODD (already active)	ODD:2			ORNO	ORN0 ORN2
				DELN	DEL0 DEL1
		ILD (already active)	ILD:2	SPRO	SPR1
				CPRO	CPR1
				QTYI	QTY0 CD04:2
				AUCT	CD04:3
				TDES	TDE0

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

2. Click the Activate button on the Main Toolbar.
3. Click each inactive segment that you need to use. This will activate the segments.

If you accidentally click a segment that you did not mean to activate, click the segment with the right mouse button to access the shortcut menu. Select Deactivate from the shortcut menu.

4. Click each inactive element that you need to use. This will activate the elements.

If you accidentally click an element that you did not mean to activate, click the element with the right mouse button to access the shortcut menu. Select Deactivate from the shortcut menu.

5. After you activate all the necessary groups, segments, and elements, click the Activate button on the Main Toolbar to turn activation mode off.

Verifying EDI Delimiters

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

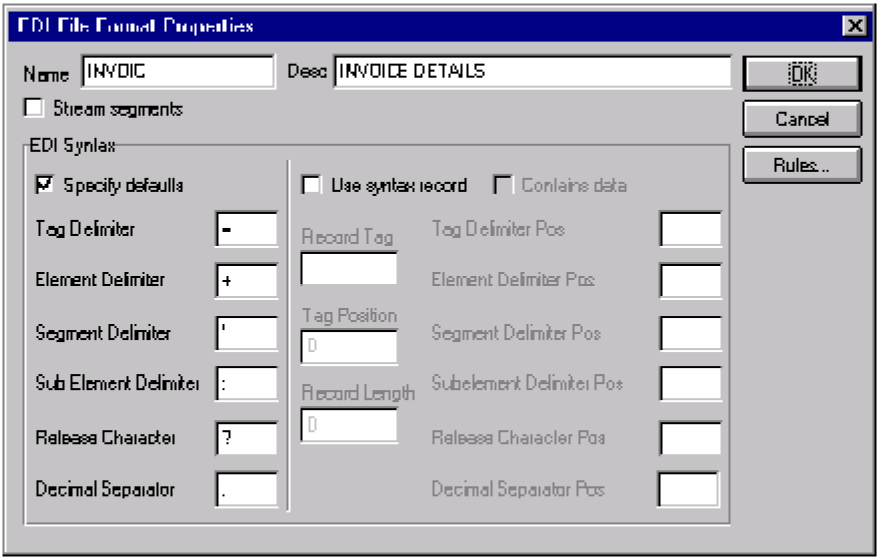
STOP

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

Complete the following steps to verify EDI delimiters:

1. Click the INVOIC INVOICE DETAILS file icon with the right mouse button to access the shortcut menu.

Select Properties... from the shortcut menu to display the EDI File Format Properties dialogue.


EDI File Format Properties

Name INVOIC Desc INVOICE DETAILS

□ Stream segments

EDI Syntax

☑ Specify defaults

Tag Delimiter =

Element Delimiter +

Segment Delimiter '

Sub Element Delimiter :

Release Character ?

Decimal Separator .

□ Use syntax record □ Contains data

Record Tag

Tag Position 0

Record Length 0

Tag Delimiter Pos

Element Delimiter Pos

Segment Delimiter Pos

Subelement Delimiter Pos

Release Character Pos

Decimal Separator Pos

2. In the EDI Syntax section, verify that the Specify defaults check box is selected.

3. Verify the required delimiters for the TRADACOMS standard. If the delimiters differ from the defaults in the table below, type either the character or the hexadecimal value in the correct field. The default delimiters for TRADACOMS are the following:

Field	TRADACOMS Default Character	TRADACOMS Default Hex Value
Tag Delimiter	= (equal)	0x3D
Element Delimiter	+ (plus)	0x2B
Segment Delimiter	' (single quote)	0x27
Sub Element Delimiter	: (colon)	0x3A
Release Character	? (question mark)	0x3F
Decimal Separator	. (full stop)	0x2E

4. Click OK to exit the EDI File Format Properties dialogue.

Now you can define your application file layout to Gentran:Director.

Defining the Rest of Your Application File Format

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

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

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

Copying the HEADER Record

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

Complete the following steps to copy the HEADER record from the system import map to the import map:

1. Open the HEADER.MAP system import map, if it is not already open.
2. Highlight the HEADER record.
3. Click Copy on the Main Toolbar.
4. Open the PET_TINV.MAP import map, if it is not already open.
5. Highlight the INPUT positional file icon that you want the copied record pasted *after.*
6. Click Paste on the Main Toolbar to paste the contents of the Clipboard.

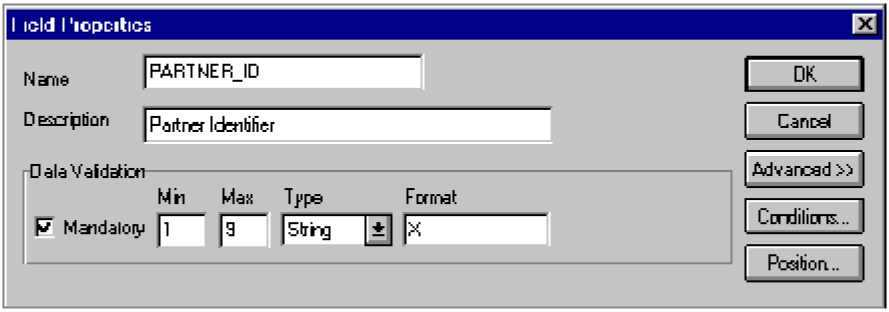
   The HEADER record is pasted as a subordinate map component to the INPUT positional file icon.

Removing the HEADER Standard Rules

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

Complete the following steps to remove the standard rules from the HEADER record in the import map:

1. Double-click the PARTNER_ID field in the HEADER record to display the Field Properties dialogue.



I ield I'ropcitics

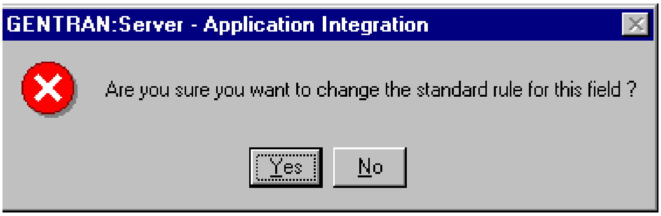
Name: PARTNER_ID

Description: Partner Identifier

Data Validation

☑ Mandatory Min: 1 Max: 9 Type: String Format: X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. In the Advanced section, click the Use Standard Rule check box to remove the standard rule.
4. The system asks you confirm the removal of the standard rule from the field.



GENTRAN:Server - Application Integration

Are you sure you want to change the standard rule for this field ?

5. Click YES to remove the standard rule.
6. Click OK to save the change to the field.
7. Repeat steps 1 – 6 for the APPL_ID and APPL_ALIAS_VALUE fields.

Creating the INVBEGIN Record

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

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

Complete the following steps to create the INVBEGIN record:

1. Highlight the HEADER record and select Insert from the Edit menu. Select Record from the cascading menu to display the Positional Record Properties dialogue.

   Remember to select "Insert" from the Edit menu when you want to create equal records/groups and "Create Sub" from the Edit menu to create subordinate groups/records/fields.

2. In the Name field, type "INVBEGIN" as the record identification code.

   STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

3. In the Desc field, type "Beginning Record" as the description of the record. This field is used to provide a brief explanation of the record that will allow you to differentiate it from similar records.

4. In the Tag field, type "BEG" as the record identification code <TAG>. For example, a record is recognised by the system as the following:

   <TAG>[Field_1][Field_2]

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

   You must define the record tag on the Positional Record Properties dialogue for outbound maps, instead of defining fields with the purpose of explicitly containing the record tag.

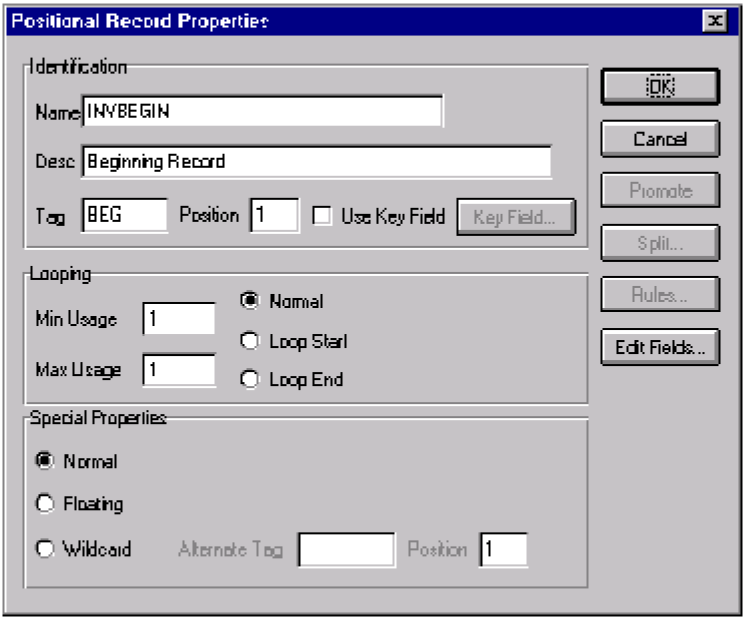
5. In the Position field, accept the default of "1" for the starting column position of the tag in the data record.

6. In the Min Usage field, type "1" to make the record mandatory. This field contains the minimum number of times the record must repeat.

If the Min Usage field contains “0” (zero), the record is conditional. If the Min Usage field contains “1” or greater, the record is mandatory.

Accept the default of “1” in the Max Usage field so the beginning record only occurs once (not a looping structure).

The Positional Record Properties dialogue should now look like this:



Identification

Name: INVBEGIN

Desc: Beginning Record

Tag: BEG Position: 1 □ Use Key Field

Looping

Min Usage: 1

Max Usage: 1

● Normal

○ Loop Start

○ Loop End

Special Properties

● Normal

○ Floating

○ Wildcard Alternate Tag: Position: 1

7. Click OK to create the INVBEGIN record.

Creating the INVCOMMENT Group

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

Complete the following steps to create the INVCommentGroup at the same level as the INVBEGIN record:

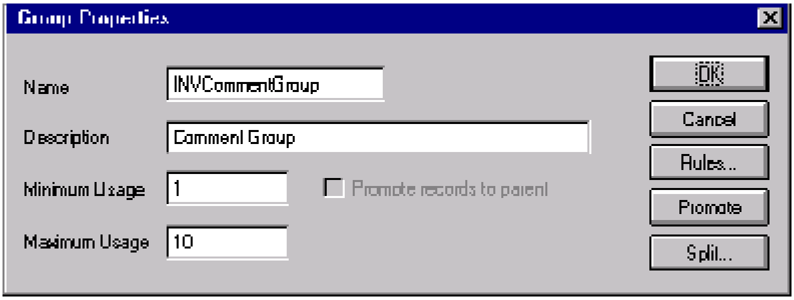
1. Highlight the INVBEGIN record and select Insert from the Edit menu. Select Group from the cascading menu.

   The Group Properties dialogue is displayed.

2. In the Name field, type “INVCommentGroup” to indicate that this group will contain all of the comment records. This is the segment loop identification code.

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

3. In the Description field, type “Comment Group” as the loop description. This field is used to provide a brief explanation of the loop.
4. In the Minimum Usage field, type “1” to make the loop mandatory. This is the minimum number of times the loop must be repeated. For a conditional loop, the minimum usage should always be “0” (zero). For a mandatory loop, the minimum usage should be “1” or greater.
5. In the Maximum Usage field, type “10”. This indicates that this group of records will occur up to ten times. The Group Properties dialogue should now look like this:



Name: INVCommentGroup

Description: Comment Group

Minimum Usage: 1

□ Promote records to parent

Maximum Usage: 10

6. Click OK to create the INVCommentGroup.

Creating the INVCOMMENT Record

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

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

Complete the following steps to create the INVCOMMENT1 record:

1. Click the INVCommentGroup to highlight it and select Create Sub from the Edit menu. Select Record from the cascading menu.

   The Positional Record Properties dialogue is displayed.

2. In the Name field, type “INVCOMMENT” as the record identification code.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

3. In the Desc field, type “Invoice Comment Record 1” as the description of the record. This field is used to provide a brief explanation of the record that will allow you to differentiate it from similar records.
4. In the Tag field, type “CMT” as the record identification code <TAG>. For example, a record is recognised by the system as the following:

   <TAG>[Field_1][Field_2]

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

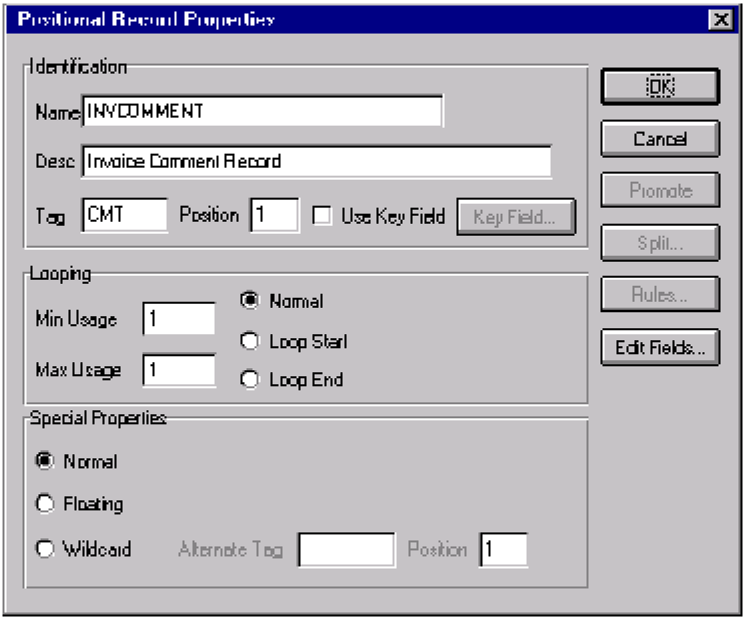
   You must define the record tag on the Positional Record Properties dialogue for outbound maps, instead of defining fields with the purpose of explicitly containing the record tag.

5. In the Position field, accept the default of “1” for the starting column position of the tag in the data record.
6. In the Min Usage field, type “1” to make the record mandatory. This field contains the minimum number of times the record must repeat.

If the Min Usage field contains “0” (zero), the record is conditional. If the Min Usage field contains “1” or greater, the record is mandatory”

Accept the default of “1” in the Max Usage field so this record only occurs once (not a looping structure).

The Positional Record Properties dialogue should now look like this:



Identification

Name INVCOMMENT

Desc Invoice Comment Record

Tag CMT Position 1 □ Use Key Field

Looping

Min Usage 1

Max Usage 1

● Normal

○ Loop Start

○ Loop End

Special Properties

● Normal

○ Floating

○ Wildcard Alternate Tag Position 1

7. Click OK to create the INVCOMMENT record.

Creating the INVDETAIL Record

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

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

Complete the following steps to create the INVDETAIL record:

1. Highlight the INVCommentGroup and select Insert from the Edit menu. Select Record from the cascading menu to display the Positional Record Properties dialogue.
2. In the Name field, type “INVDETAIL” the record identification code.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

3. In the Desc field, type “Detail Record” as a description of the record. This field is used to provide a brief explanation of the record that will allow you to differentiate it from similar records.

4. In the Tag field, type “DET” as the record identification code <TAG>. For example, a record is recognised by the system as the following:

   <TAG>[Field_1][Field_2]

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

   You must define the record tag on the Positional Record Properties dialogue for outbound maps, instead of defining fields with the purpose of explicitly containing the record tag.

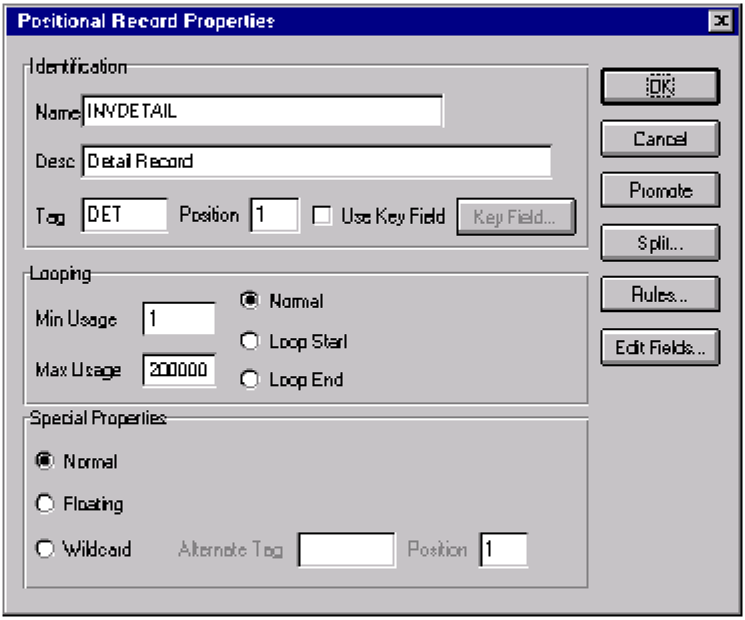
5. In the Position field, accept the default of “1” as the starting column position of the tag in the data record.

6. In the Min Usage field, type “1” to make the record mandatory. This field contains the minimum number of times the record must repeat.

   If the Min Usage field contains “0” (zero), the record is conditional. If the Min Usage field contains “1” or greater, the record is mandatory.

7. In the Max Usage field, type “200000” as the maximum number of times the record can repeat.

The Positional Record Properties dialogue should now look like this:



Identification

Name: INVDETAIL

Desc: Detail Record

Tag: DET Position: 1 ☐ Use Key Field

Looping

Min Usage: 1

Max Usage: 200000

● Normal ○ Loop Start ○ Loop End

Special Properties

● Normal

○ Floating

○ Wildcard Alternate Tag: Position: 1

8. Click OK to create the INVDETAIL record.

Creating the TEMPVATSTL Record

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

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

Complete the following steps to create the TEMPVATSTL record:

1. Highlight the INVDETAIL record and select Insert from the Edit menu. Select Record from the cascading menu to display the Positional Record Properties dialogue.

2. In the Name field, type “TEMPVATSTL” as the record identification code.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

3. In the Desc field, type “Temporary Storage VAT Subtotals” as a description of the record. This field is used to provide a brief explanation of the record that will allow you to differentiate it from similar records.
4. In the Tag field, type “TMP” as the record identification code <TAG>. For example, a record is recognised by the system as the following:

   <TAG>[Field_1][Field_2]

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

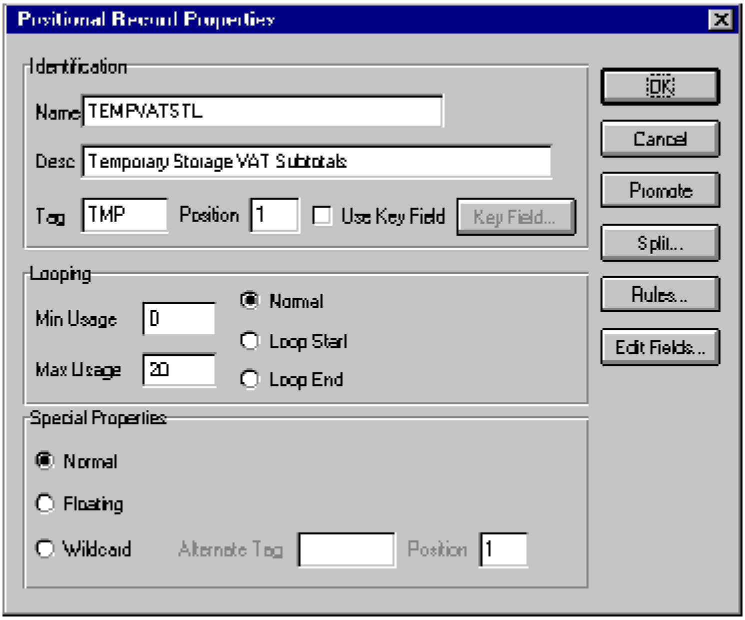
   You must define the record tag on the Positional Record Properties dialogue for outbound maps, instead of defining fields with the purpose of explicitly containing the record tag.

5. In the Position field, accept the default of “1” as the starting column position of the tag in the data record.
6. In the Min Usage field, accept the default of “0” so the record will be conditional. This field contains the minimum number of times the record must repeat.

   If the Min Usage field contains “0” (zero), the record is conditional. If the Min Usage field contains “1” or greater, the record is mandatory.

7. In the Max Usage field, type “20” as the maximum number of times the record can repeat.

The Positional Record Properties dialogue should now look like this:



Identification

Name TEMPVATSTL

Desc Temporary Storage VAT Subtotals

Tag TMP Position 1 □ Use Key Field

Looping

Min Usage 0

Max Usage 20

● Normal

○ Loop Start

○ Loop End

Special Properties

● Normal

○ Floating

○ Wildcard Alternate Tag Position 1

8. Click OK to create the TEMPVATSTL record.

Creating the INVSUMMARY Record

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

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

Complete the following steps to create the INVSUMMARY record:

1. Highlight the INVDETAIL record and select Insert from the Edit menu. Select Record from the cascading menu to display the Positional Record Properties dialogue.
2. In the Name field, type “INVSUMMARY” the record identification code.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

3. In the Desc field, type “Summary Record” as a description of the record. This field is used to provide a brief explanation of the record that will allow you to differentiate it from similar records.
4. In the Tag field, type “SUM” as the record identification code <TAG>. For example, a record is recognised by the system as the following:

   <TAG>[Field_1][Field_2]

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

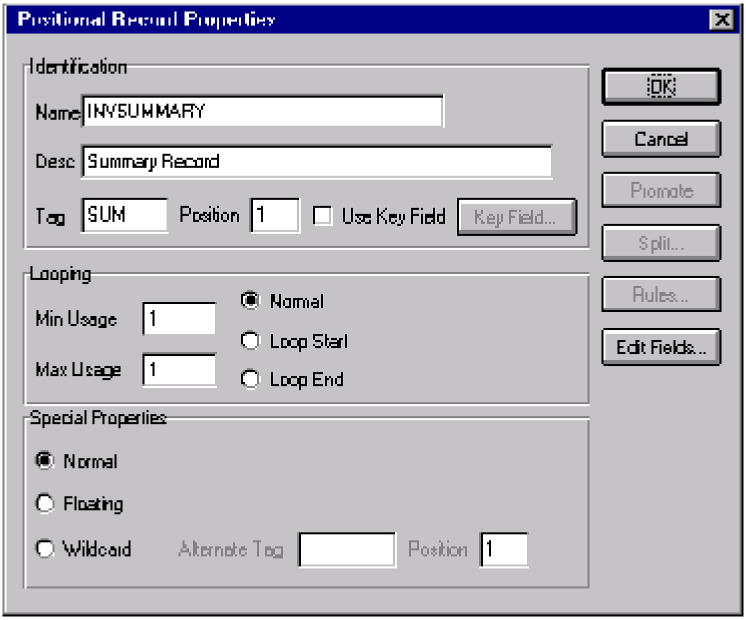
You must define the record tag on the Positional Record Properties dialogue for outbound maps, instead of defining fields with the purpose of explicitly containing the record tag.

5. In the Position field, accept the default of “1” as the starting column position of the tag in the data record.
6. In the Min Usage field, type “1” to make the record mandatory. This field contains the minimum number of times the record must repeat.

If the Min Usage field contains “0” (zero), the record is conditional. If the Min Usage field contains “1” or greater, the record is mandatory.

7. Accept the default of “1” in the Max Usage field. This is the maximum number of times the record can repeat.

The Positional Record Properties dialogue should now look like this:



Positional Record Properties

Identification

Name: INVSUMMARY

Desc: Summary Record

Tag: SUM

Position: 1

□ Use Key Field

Looping

Min Usage: 1

Max Usage: 1

● Normal ○ Loop Start ○ Loop End

Special Properties

● Normal ○ Floating ○ Wildcard

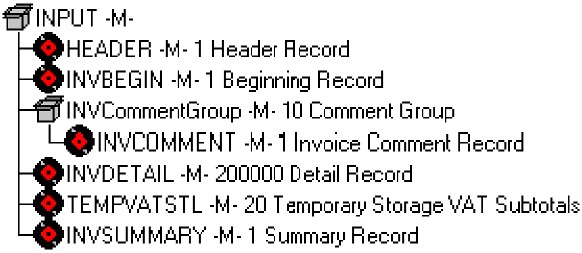
Alternate Tag:

Position: 1

8. Click OK to create the INVSUMMARY record.

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

The application side of the map (input) should now look like this:



```
INPUT -M-
├HEADER -M- 1 Header Record
├INVBEGIN -M- 1 Beginning Record
├INVCommentGroup -M- 10 Comment Group
│ └INVCOMMENT -M- 1 Invoice Comment Record
├INVDETAIL -M- 200000 Detail Record
├TEMPVATSTL -M- 20 Temporary Storage VAT Subtotals
└INVSUMMARY -M- 1 Summary Record
```

Creating the Application Fields

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

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

The application fields that you need to create are listed under their respective records in the following table. The name, mandatory status, description, data type, and field length for each field is given in the table.

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
Header Record (HEADER) 1 per message					
PARTNER_ID	X	Partner Identifier	String	1	9
APPL_ID		Application ID	String	0	6
APPL_ALIAS_VALUE		Application Alias Value	String	0	6
Beginning Record (INVBEGIN) 1 per message					
BILLTOACCTBEG	X	Bill to Acct Number	String	1	9
DOCTYPBEG	X	Document Type	String	1	6
INVNUMBEG	X	Invoice Number	String	1	10
INVDATE	X	Invoice Date	Dt/Tm (YYMMDD)	6	6
PONUM		Purchase Order Number	String	0	20
PODATE		Purchase Order Date	Dt/Tm (YYMMDD)	6	6
SHIPTONAME		Ship To Name	String	0	35
SHIPTOADD1		Ship To Address Line 1	String	0	35
SHIPTOADD2		Ship To Address Line2	String	0	35
SHIPTOCITY		Ship To City	String	0	35
SHIPTOSTATE		Ship To State	String	0	35
SHIPTOPOSTALCODE		Ship To Postal Code	String	0	8
SHIPTOLOC		Ship To Store Location	String	0	8
BOLNUM		Bill of Lading Number	String	0	12
SHIPDATE		Ship Date	Dt/Tm (YYMMDD)	6	6
TERMSNETDUE		Terms New Due Days	Number (R0)	0	3
TERMSTEXT		Payment Terms Text	String	0	40
Comment Group (INVCommentGroup) 10 per message					
Invoice Comment Record (INVCOMMENT) 1 per message					
BILLTOACCTCMT	X	Bill To Acct Number	String	1	9
DOCTYPCMT	X	Document Type	String	1	6
INVNUMCMT	X	Invoice Number	String	1	10

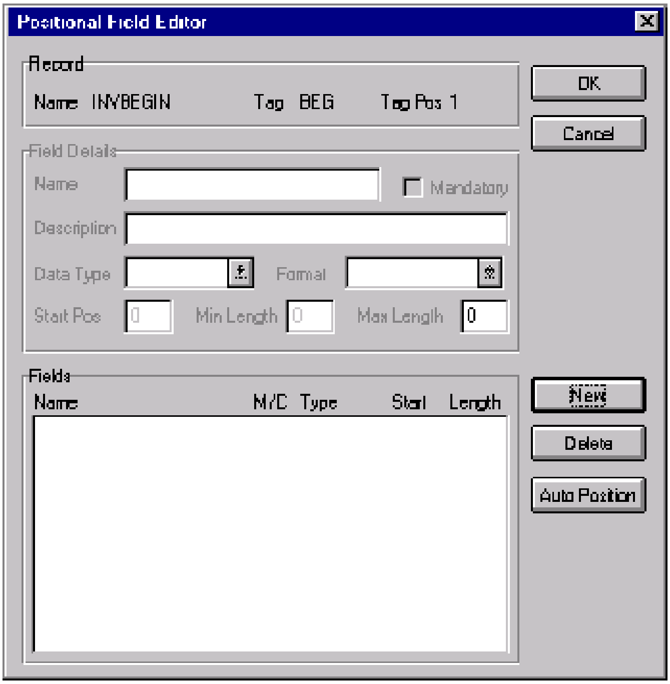
Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
MESSAGETEXT1		Message Text Line 1	String	0	40
MESSAGETEXT2		Message Text Line 2	String	0	40
MESSAGETEXT3		Message Text Line 3	String	0	40
MESSAGETEXT4		Message Text Line 4	String	0	40
Detail Record (INVDETAIL) up to 200,000 per message					
BILLTOACCTDET	X	Bill To Acct Number	String	1	9
DOCTYPDET	X	Document Type	String	1	6
INVNUMDET	X	Invoice Number	String	1	10
ITEMNUM	X	Item Number (Our Item)	Number (R0)	1	6
QTYSHIP	X	Quantity Shipped	Number (R0)	1	5
UNITPRICE	X	Unit Price	Number (R2)	1	7
UOM	X	Unit of Measure	String	1	4
CUSTPROCODE	X	Customer Product Code	String	1	10
UPCCODE	X	UPC Code	String	1	12
ITEMDESC		Item Description	String	0	30
VATCODE	X	VAT Rate Code	String	1	2
TEMPVATRATE		Temporary Storage VAT Percentage Rate	String	0	10
TEMPLEXC		Temporary Line Extended Cost	Number (R4)	0	14
Temporary Storage VAT Record (TEMPVATSTL) up to 10 per message					
BILLTOACCTTS	X	Bill To Acct Number	String	1	9
DOCTYPTS	X	Document Type	String	1	6
INVNUMTS	X	Invoice Number	String	1	10
SEQA		Sequence Number of STL	Number (R0)	0	10
VATRATE		VAT Rate	Number (R3)	0	6
VATRATECODE		VAT Code	String	0	1
NUMLINES		Number of Lines for VAT Code	Number (R0)	0	10
LVLA		Line Subtotal Amount (Before VAT)	Number (R2)	0	12
VATA		VAT Amount Payable	Number (R2)	0	12
Summary Record (INVSUMMARY) 1 per message					
BILLTOACCTSUM	X	Bill To Acct Number	String	1	9
DOCTYPSUM	X	Document Type	String	1	6

Field Name	Mand (x=Yes)	Description	Data Type	Min Length	Max Length
INVNUMSUM	X	Invoice Number	String	1	10
TOTALAMOUNT	X	Total Monetary Value Summary	Number (R2)	1	10

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

Complete the following steps to create the application fields for the INVBEGIN record:

1. Click the INVBEGIN application record with the right mouse button to access the shortcut menu.
2. Select Edit Fields... from the shortcut menu to access the Positional Field Editor dialogue.



Name INVBEGIN Tag BEG Tag Pos 1

Name

Start Pos 0 Min Length 0 Max Length 0

3. If the field you are creating is the first field in the record, just click NEW. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.

4. In the Name field, type “BILLTOACCTBEG”, which is the name of the first field in the INVBEGIN record. Each application field must have a unique name. It is useful to tag the end of the fields that occur in multiple records with a suffix that identifies the record that contains it. For example, you will use “BEG” for the Beginning record.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

5. Click the Mandatory check box to identify it as a required field.
6. In the Description field, type “Bill To Account Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7. From the Data Type drop-down list box, select “String” as the type of the field, to specify that this field must contain alphanumeric data.

If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos field. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.

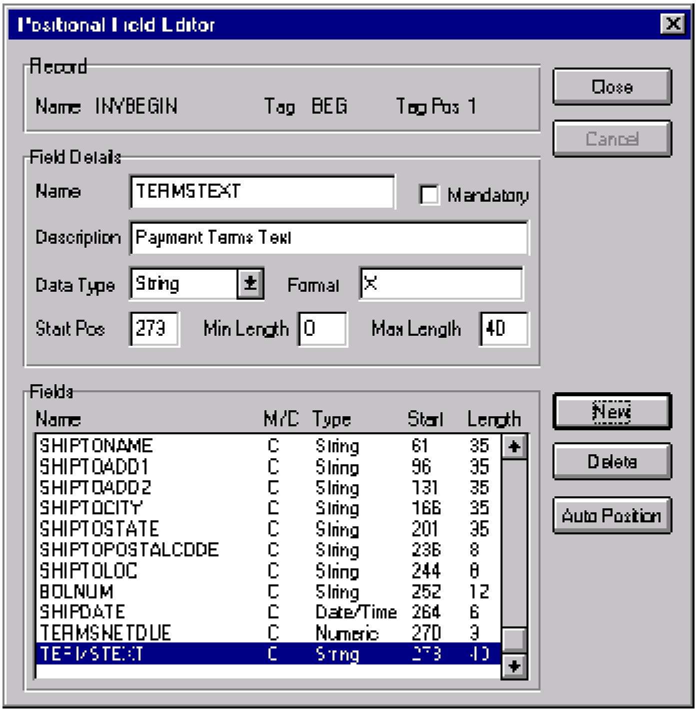
In this tutorial, you will use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see step 12 below for more information on the Auto Position function.

8. If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format drop-down list box.
9. In the Min Length field, type “1” because this is the minimum number of characters that is acceptable for this field. If, according to the table starting on page 150, the minimum length of the field is “0”, then just accept that default in the Min Length field.
10. In the Max Length field, type “9” as the maximum length of the field.
11. To add the field and to create the next field positioned after it, press **ENTER**. A new field with blank values ready for you to identify will be positioned below the field.
12. Create the rest of the fields according to the record layout table starting on page 151.

13. After you add the last field, click AUTO POSITION to automatically position the fields in the record. The Auto Position function will automatically calculate the start position in the record of each field based on the criteria that each field is positioned directly after the previous field and is of the length specified in the Max Length field. Click YES to acknowledge the warning message that fields will be sequenced in order.

STOP **Please note that it is only valid to use the Auto Position function if you define a record tag on the Positional Record Properties dialogue, and if you define every field in the record in the sequence that each field occurs, as you are doing in this tutorial.**

The Positional Field Editor dialogue should now look like the following (only the last eleven fields in the record are currently visible in the Fields list box):



Positional Field Editor

Record

Name INVBEGIN Tag BEG Tag Pos 1

Field Details

Name TERMSTEXT ☐ Mandatory

Description Payment Terms Text

Data Type String Format X

Start Pos 273 Min Length 0 Max Length 40

Fields

Name	M/C	Type	Start	Length
SHIPTONAME	C	String	61	35
SHIPTOADD1	C	String	96	35
SHIPTOADD2	C	String	131	35
SHIPTOCITY	C	String	166	35
SHIPTOSTATE	C	String	201	35
SHIPTOPOSTALCODE	C	String	236	8
SHIPTOLOC	C	String	244	8
BOLNUM	C	String	252	12
SHIPDATE	C	Date/Time	264	6
TERMSNETDUE	C	Numeric	270	3
TERMSTEXT	C	String	273	40

14. When you are completely finished adding fields to the INVBEGIN record, click CLOSE.

Complete the following steps to create the application fields for the INVCOMMENT record:

1. Click the INVCOMMENT application record with the right mouse button to access the shortcut menu.
2. Select Edit Fields... from the shortcut menu to access the Positional Field Editor dialogue.
3. If the field you are creating is the first field in the record, just click NEW. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4. In the Name field, type “BILLTOACCTCMT”, which is the name of the first field in the INVCOMMENT record. Each application field must have a unique name. It is useful to tag the end of the fields that occur in multiple records with a suffix that identifies the record that contains it. For example, you will use “CMT” for the first comment record.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

5. Click the Mandatory check box to identify it as a required field.
6. In the Description field, type “Bill To Account Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7. From the Data Type drop-down list box, select “String” as the type of the field, to specify that this field must contain alphanumeric data.

If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos field. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.

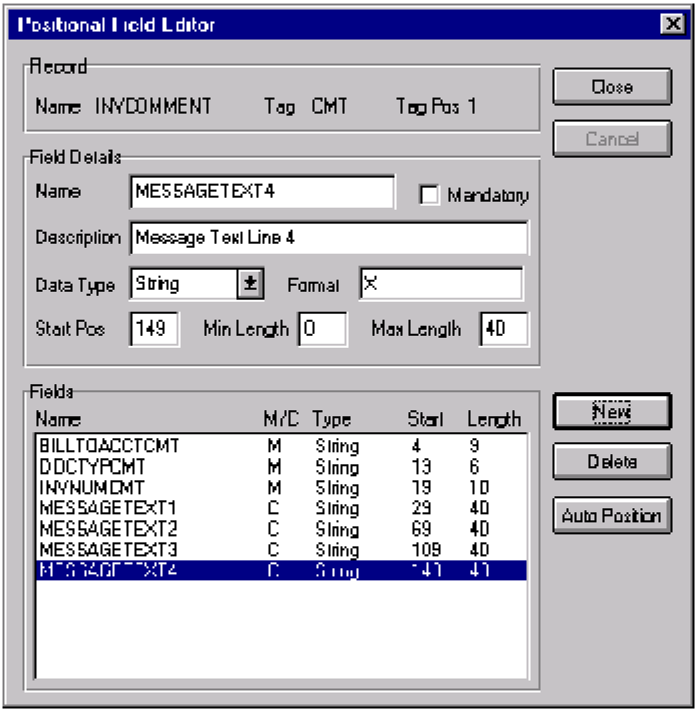
In this tutorial, you will use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see step 12 below for more information on the Auto Position function.

8. If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format drop-down list box.

9. In the Min Length field, type “1” because this is the minimum number of characters that is acceptable for this field. If, according to the table starting on page 150, the minimum length of the field is “0”, then just accept that default in the Min Length field.
10. In the Max Length field, type “9” as the maximum length of the field.
11. To add the field and to create the next field positioned after it, press **ENTER**. A new field with blank values ready for you to identify will be positioned below the field.
12. Create the rest of the fields according to the record layout table starting on page 151.
13. .After adding the last field, click AUTO POSITION to automatically position the fields in the record. The Auto Position function will automatically calculate the start position in the record of each field based on the criteria that each field is positioned directly after the previous field and is of the length specified in the Max Length field. Click YES to acknowledge the warning message that fields will be sequenced in order.

STOP **Please note that it is only valid to use the Auto Position function if you define a record tag on the Positional Record Properties dialogue, and if you define every field in the record in the sequence that each field occurs, as you are doing in this tutorial.**

The Positional Field Editor dialogue should now look like the following:



Record

Name INVCOMMENT Tag CMT Tag Pos 1

Field Details

Name: MESSAGETEXT4 ☐ Mandatory

Description: Message Text Line 4

Data Type: String Format: X

Start Pos: 149 Min Length: 0 Max Length: 40

Fields

Name	M/C	Type	Start	Length
BILLTOACCTCMT	M	String	4	9
DOCTYPCMT	M	String	13	6
INVNUMCMT	M	String	19	10
MESSAGETEXT1	C	String	29	40
MESSAGETEXT2	C	String	69	40
MESSAGETEXT3	C	String	109	40
MESSAGETEXT4	C	String	149	40

14. When you are completely finished adding fields to the INVCOMMENT record, click CLOSE.

Complete the following steps to create the application fields for the INVDETAIL record:

1. Click the INVDETAIL application record with the right mouse button to access the shortcut menu.

2. Select Edit Fields... from the shortcut menu to access the Positional Field Editor dialogue.

3. If the field you are creating is the first field in the record, just click NEW. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.

4. In the Name field, type “BILLTOACCTDET”, which is the name of the first field in the INVDETAIL record. Each application field must have a unique name. It is useful to tag the end of the fields that occur in

multiple records with a suffix that identifies the record that contains it. For example, you will use "DET" for the detail record.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

5. Click the Mandatory check box to identify it as a required field.
6. In the Description field, type "Bill To Account Number" as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7. From the Data Type drop-down list box, select "String" as the type of the field, to specify that this field must contain alphanumeric data.

If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos field. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.

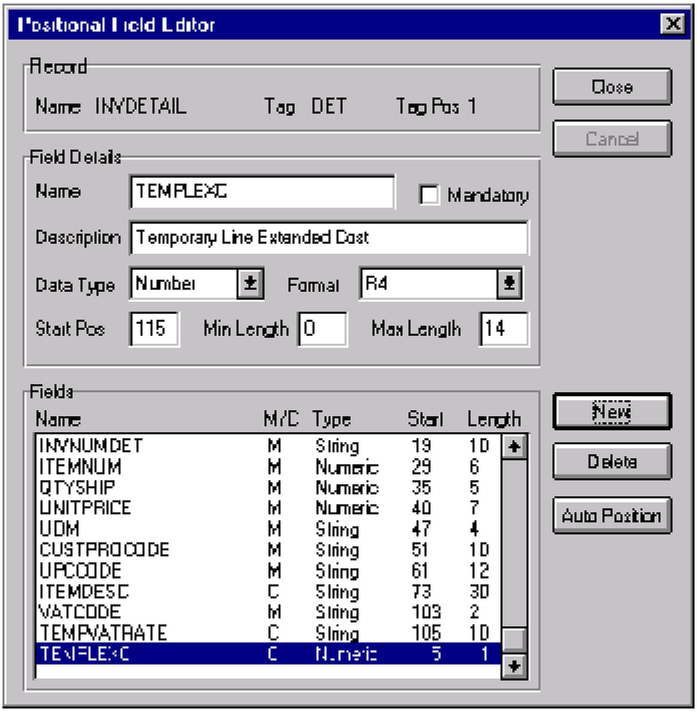
In this tutorial, you will use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see step 12 below for more information on the Auto Position function.

8. If the Data Type is "Number" or "Dt/Tm", select the appropriate format from the Format drop-down list box.
9. In the Min Length field, type "1" because this is the minimum number of characters that is acceptable for this field. If, according to the table starting on page 150, the minimum length of the field is "0", then just accept that default in the Min Length field.
10. In the Max Length field, type "9" as the maximum length of the field.
11. To add the field and to create the next field positioned after it, press **ENTER**. A new field with blank values ready for you to identify will be positioned below the field.
12. Create the rest of the fields according to the record layout table starting on page 151.
13. After adding the last field, click AUTO POSITION to automatically position the fields in the record. The Auto Position function will

automatically calculate the start position in the record of each field based on the criteria that each field is positioned directly after the previous field and is of the length specified in the Max Length field. Click YES to acknowledge the warning message that fields will be sequenced in order.

STOP **Please note that it is only valid to use the Auto Position function if you define a record tag on the Positional Record Properties dialogue, and if you define every field in the record in the sequence that each field occurs, as you are doing in this tutorial.**

The Positional Field Editor dialogue should now look like the following (only the last eleven fields in the record are currently visible in the Fields list box):



Record

Name INVDETAIL Tag DET Tag Pos 1

Field Details

Name TEMPLEXC ☐ Mandatory

Description Temporary Line Extended Cost

Data Type Number Format R4

Start Pos 115 Min Length 0 Max Length 14

Fields

Name	M/C	Type	Start	Length
INVNUMDET	M	String	19	10
ITEMNUM	M	Numeric	29	6
QTYSHIP	M	Numeric	35	5
UNITPRICE	M	Numeric	40	7
UOM	M	String	47	4
CUSTPRODCODE	M	String	51	10
UPCCODE	M	String	61	12
ITEMDESC	C	String	73	30
VATCODE	M	String	103	2
TEMPVATRATE	C	String	105	10
TEMPLEXC	C	Numeric	5	1

14. When you are completely finished adding fields to the INVDETAIL record, click CLOSE.

Complete the following steps to create the application fields for the TEMPVATSTL record:

1. Click the TEMPVATSTL application record with the right mouse button to access the shortcut menu.
2. Select Edit Fields... from the shortcut menu to access the Positional Field Editor dialogue.
3. If the field you are creating is the first field in the record, just click NEW. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.
4. In the Name field, type “BILLTOACCTTS”, which is the name of the first field in the TEMPVATSTL record. Each application field must have a unique name. It is useful to tag the end of the fields that occur in multiple records with a suffix that identifies the record that contains it. For example, you will use “TS” for the temporary storage record.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

5. Click the Mandatory check box to identify it as a required field.
6. In the Description field, type “Bill To Account Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7. From the Data Type drop-down list box, select “String” as the type of the field, to specify that this field must contain alphanumeric data.

If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos field. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.

In this tutorial, you will use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see step 12 below for more information on the Auto Position function.

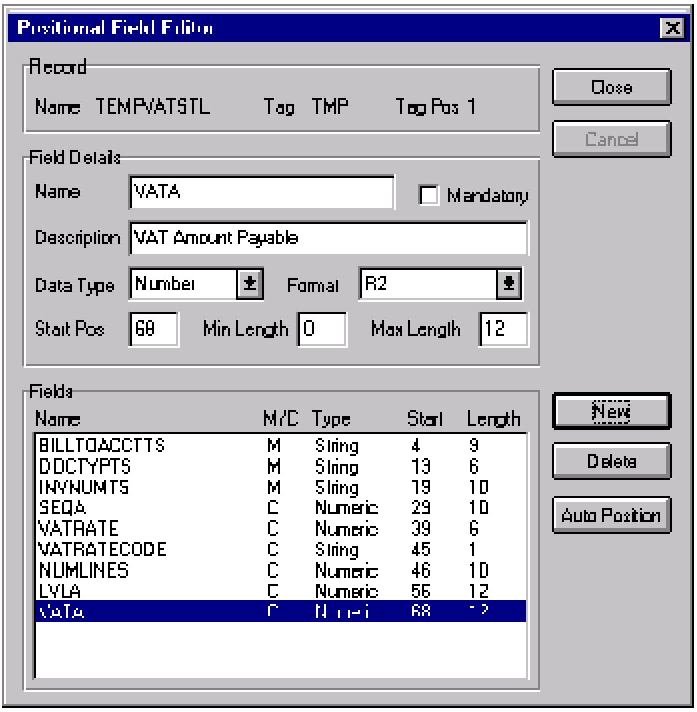
8. If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format drop-down list box.

9. In the Min Length field, type “1” because this is the minimum number of characters that is acceptable for this field. If, according to the table starting on page 150, the minimum length of the field is “0”, then just accept that default in the Min Length field.
10. In the Max Length field, type “9” as the maximum length of the field.
11. To add the field and to create the next field positioned after it, press **ENTER**. A new field with blank values ready for you to identify will be positioned below the field.
12. Create the rest of the fields according to the record layout table starting on page 151.
13. After adding the last field, click AUTO POSITION to automatically position the fields in the record. The Auto Position function will automatically calculate the start position in the record of each field based on the criteria that each field is positioned directly after the previous field and is of the length specified in the Max Length field. Click YES to acknowledge the warning message that fields will be sequenced in order.

STOP

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

The Positional Field Editor dialogue should now look like the following:



Record

Name TEMPVATSTL Tag TMP Tag Pos 1

Field Details

Name: VATA ☐ Mandatory

Description: VAT Amount Payable

Data Type: Number Format: R2

Start Pos: 68 Min Length: 0 Max Length: 12

Fields

Name	M/C	Type	Start	Length
BILLTOACCTTS	M	String	4	9
DOCTYPTS	M	String	13	6
INVNUMTS	M	String	19	10
SEQA	C	Numeric	29	10
VATRATE	C	Numeric	39	6
VATRATECODE	C	String	45	1
NUMLINES	C	Numeric	46	10
LVLA	C	Numeric	56	12
VATA	C	Numeric	68	12

14. When you are completely finished adding fields to the TEMPVATSTL record, click CLOSE.

Complete the following steps to create the application fields for the INVSUMMARY record:

1. Click the INVSUMMARY application record with the right mouse button to access the shortcut menu.

2. Select Edit Fields... from the shortcut menu to access the Positional Field Editor dialogue.

3. If the field you are creating is the first field in the record, just click NEW. A highlight bar is displayed in the Fields section where the new field is positioned. You need to fill in the field values in the Field Details section.

4. In the Name field, type “BILLTOACCTSUM”, which is the name of the first field in the INVSUMMARY record. Each application field must have a unique name. It is useful to tag the end of the fields that occur in

multiple records with a suffix that identifies the record that contains it. For example, you will use “SUM” for the summary record.

STOP **Do not use spaces or hyphens (-) in the Name field. You can use the underscore (_) to separate words.**

5. Click the Mandatory check box to identify it as a required field.
6. In the Description field, type “Bill To Account Number” as the description of the field. The description is used to provide a brief explanation of the field that will allow you to differentiate it from similar fields.
7. From the Data Type drop-down list box, select “String” as the type of the field, to specify that this field must contain alphanumeric data.

If you wanted to indicate the exact position of the field in the record, you would type the starting position of the field in the Start Pos field. You will want to specify field start positions if, for example, you will only be using a few fields but you want them positioned exactly in the record.

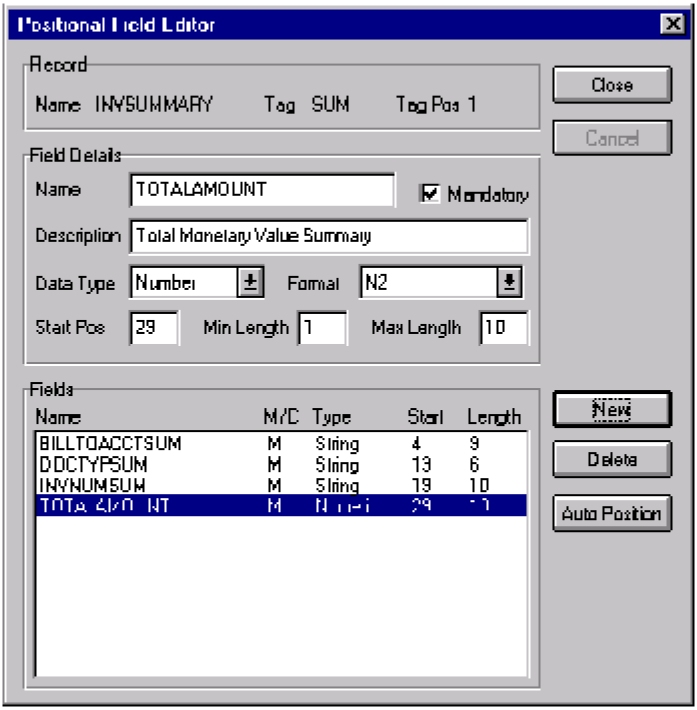
In this tutorial, you will use the alternative to specifying the start position of each field, which is to add the fields sequentially in the record and then use the Auto Position function. Please see step 12 below for more information on the Auto Position function.

8. If the Data Type is “Number” or “Dt/Tm”, select the appropriate format from the Format drop-down list box.
9. In the Min Length field, type “1” because this is the minimum number of characters that is acceptable for this field. If, according to the table starting on page 150, the minimum length of the field is “0”, then just accept that default in the Min Length field.
10. In the Max Length field, type “9” as the maximum length of the field.
11. To add the field and to create the next field positioned after it, press **ENTER**. A new field with blank values ready for you to identify will be positioned below the field.
12. Create the rest of the fields according to the record layout table starting on page 151.
13. After adding the last field, click AUTO POSITION to automatically position the fields in the record. The Auto Position function will automatically calculate the start position in the record of each field

based on the criteria that each field is positioned directly after the previous field and is of the length specified in the Max Length field. Click YES to acknowledge the warning message that fields will be sequenced in order.

STOP **Please note that it is only valid to use the Auto Position function if you define a record tag on the Positional Record Properties dialogue, and if you define every field in the record in the sequence that each field occurs, as you are doing in this tutorial.**

The Positional Field Editor dialogue should now look like the following:



Record

Name INVSUMMARY Tag SUM Tag Pos 1

Field Details

Name TOTALAMOUNT ☑ Mandatory

Description Total Monetary Value Summary

Data Type Number Format N2

Start Pos 29 Min Length 1 Max Length 10

Fields

Name	M/C	Type	Start	Length
BILLTOACCTSUM	M	String	4	9
DOCTYPSUM	M	String	13	6
INVNUMSUM	M	String	19	10
TOTALAMOUNT	M	Number	29	10

14. When you are completely finished adding fields to the INVSUMMARY record, click CLOSE.

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

Mapping Your Data

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

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

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

HEADER Record

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

INVBEGIN Record

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

BILLTOACCTBEG

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

DOCTYPBEG

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

INVNUMBEG

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

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

Also, we recommend that you set up a Document Name for each map, to make the identification of a message created by this map easier in Gentran:Director. The Document Name allows you to differentiate between messages in the document browsers in Gentran:Director.

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

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

Changing the max use of the IRF segment

Complete the following steps to make the IRF a nonrepeating segment:

1. Click the IRF segment with the right mouse button to access the shortcut menu.
2. Select Properties... from the shortcut menu to access the EDI Segment Properties dialogue.
3. In the Max Usage field, type "1" to make the IRF a nonrepeating segment. This field currently contains "999999".
4. Click OK to accept the change to the IRF segment.

Mapping the invoice number

Complete the following steps to map the invoice number:

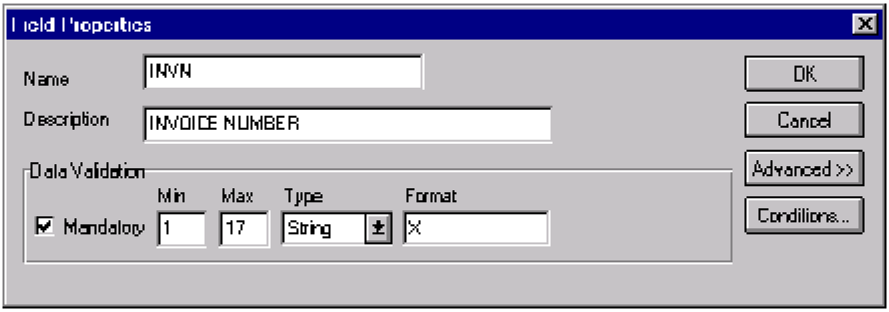
1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field INVNUMBEG in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow": LINK ⇨

3. Click the IRF INVN element (Invoice Number).
4. A line is displayed, visually illustrating the link between the INVNUMBEG field and the IRF INVN element.

Setting up the document name

Complete the following steps to set up the Document Name for this invoice:

1. Double-click the IRF INVN element (Invoice Number) to display the Field Properties dialogue:



Field Properties

Name: INVN

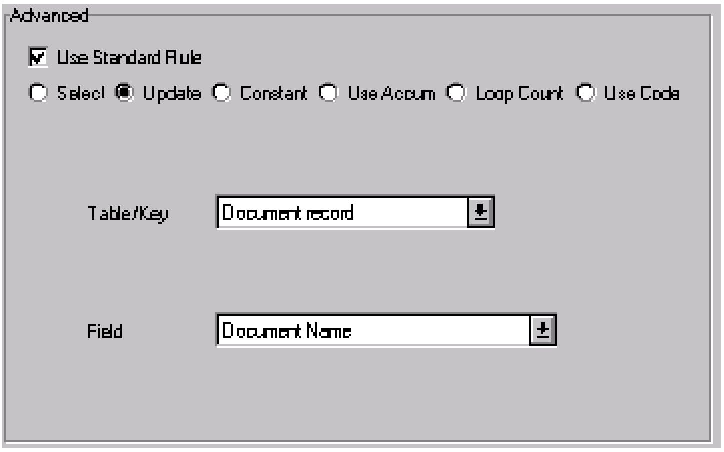
Description: INVOICE NUMBER

Data Validation

☑ Mandatory — Min: 1 — Max: 17 — Type: String — Format: X

2. If the Advanced section of the Field Properties dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. In the Advanced section, click the Use Standard Rule check box, to display the rule options.
4. Click the Update radio button.
5. From the Table/Key drop-down list box, choose “Document record”. This indicates that you are updating the document record in Gentran:Director.
6. From the Field drop-down list box, choose “Document Name.” This indicates that you are updating the Name field in the document browsers with the contents of this element.

The Advanced section of the dialogue should now look like this:



Advanced

☑ Use Standard Rule

○ Select ● Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table/Key: Document record

Field: Document Name

7. Click OK to set up the document name.

INVDATE

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

Mapping the invoice and tax-point dates

Complete the following steps to map the invoice and tax-point dates:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field INVDATE in the INVBEGIN record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the IRF IVDT element (Date of Invoice).
4. A line is displayed, visually illustrating the link between the INVDATE field and the IRF IVDT element.
5. Click the application field INVDATE in the INVBEGIN record again. When you click the field, the cursor changes to a “Link Arrow”.
6. Click the IRF TXDT element (Tax-Point Date).
7. A line is displayed, visually illustrating the link between the INVDATE field and the IRF TXDT element.

PONUM

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

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

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

Changing the maximum usage of the ODD group

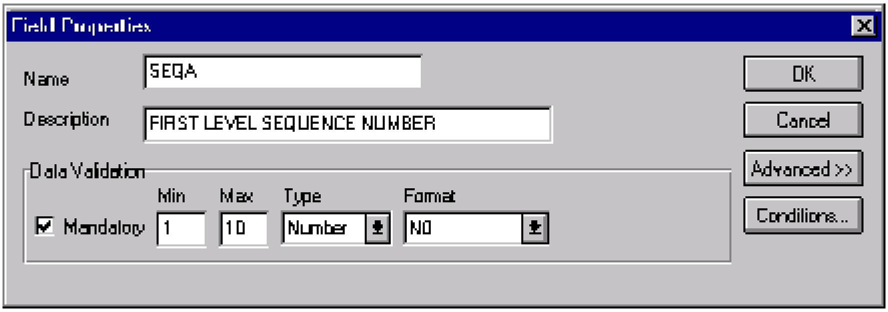
Complete the following steps to make the ODD a nonrepeating group:

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

Using an extended rule to specify the PYT SEQA

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

1. Double-click the PYT SEQA element (First Level Sequence Number) to access the Field Properties dialogue.



Field Properties

Name: SEQA

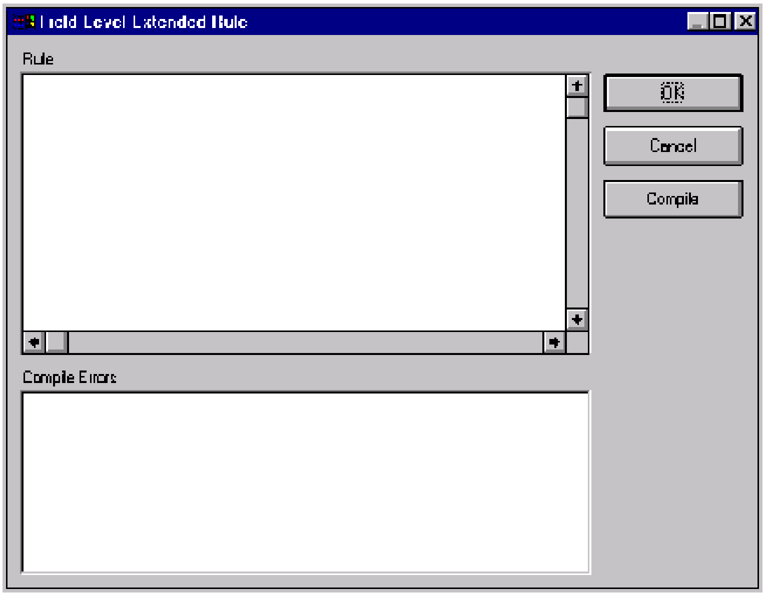
Description: FIRST LEVEL SEQUENCE NUMBER

Data Validation

☑ Mandatory — Min: 1 — Max: 10 — Type: Number — Format: N0

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.

3. Click RULE... to display the Field Level Extended Rule dialogue.



4. In the Rule list box, type the following:

   #SEQA = 1;

   The “#SEQA = 1” sets the first level sequence number for the PYT segment to one. The system will execute this rule when it processes the PYT SEQA element.

5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

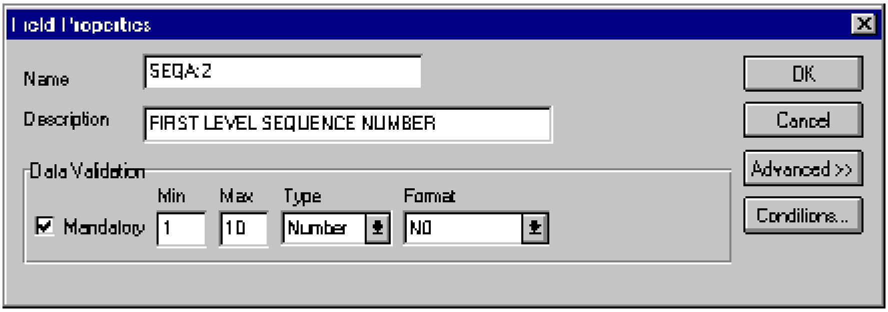
The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list box.

6. Click OK to add the extended rule to the PYT SEQA element.
7. Click CLOSE to exit the Field Properties dialogue.

Using an extended rule to specify the DNA SEQA:2

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

1. Double-click the DNA SEQA:2 element (First Level Sequence Number) to access the Field Properties dialogue.



Field Properties

Name: SEQA:2

Description: FIRST LEVEL SEQUENCE NUMBER

Data Validation

	Min	Max	Type	Format
☑ Mandatory	1	10	Number	N0

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. Click RULE... to display the Field Level Extended Rule dialogue.
4. In the Editor list box, type the following:

   #SEQA:2 = 1;

   The “#SEQA:2 = 1” sets the first level sequence number for the DNA segment to one. The system will execute this rule when it processes the DNA SEQA:2 element.
5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

   The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list box.
6. Click OK to add the extended rule to the DNA SEQA:2 element.
7. Click CLOSE to exit the Field Properties dialogue.

Using an extended rule to specify the ODD:2 SEQA

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

1. Double-click the ODD:2 SEQA element (First Level Sequence Number) to access the Field Properties dialogue.
2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. Click RULE... to display the Field Level Extended Rule dialogue.
4. In the Rule list box, type the following:

   #SEQA = 1;

   The “#SEQA = 1” sets the first level sequence number for the ODD:2 segment to one. The system will execute this rule when it processes the ODD:2 SEQA element.
5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

   This will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Errors list box.
6. Click OK to add the extended rule to the ODD:2 SEQA element.
7. Click CLOSE to exit the Field Properties dialogue.

Using an extended rule to specify the ILD:2 SEQA

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

1. Double-click the ILD:2 SEQA element (First Level Sequence Number) to access the Field Properties dialogue.
2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. Click RULE... to display the Field Level Extended Rule dialogue.

4. In the Rule list box, type the following:

   #SEQA = 1;

   The “#SEQA = 1” sets the first level sequence number for the ILD:2 segment to one. The system will execute this rule when it processes the ILD:2 SEQA element.

5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

   The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list box.

6. Click OK to add the extended rule to the ILD:2 SEQA element.
7. Click CLOSE to exit the Field Properties dialogue.

Mapping the order number

Complete the following steps to map the customer’s order number:

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

PODATE

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

Mapping the date order placed by customer

Complete the following steps to map the date order placed by customer (purchase order date):

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

3. Click the ODD:2 ORNO ORN2 element (Date Order Placed by Customer).
4. A line is displayed, visually illustrating the link between the PODATE field and the ODD:2 ORNO ORN2 element.

SHIPTONAME

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

Complete the following steps to map the customer's name:

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

SHIPTOADD1

Complete the following steps to map the customer's address line 1:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field SHIPTOADD1 in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow".
3. Click the CLO CADD CAD0 element (Customer's Address Line 1).
4. A line is displayed, visually illustrating the link between the SHIPTOADD1 field and the CLO CADD CAD0 element.

SHIPTOADD2

Complete the following steps to map the customer's address line 2:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field SHIPTOADD2 in the INVBEGIN record. When you click the field, the cursor changes to a "Link Arrow".
3. Click the CLO CADD CAD1 element (Customer's Address Line 2).
4. A line is displayed, visually illustrating the link between the SHIPTOADD2 field and the CLO CADD CAD1 element.

SHIPTOCITY

Complete the following steps to map the customer’s address line 3:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field SHIPTOCITY in the INVBEGIN record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the CLO CADD CAD2 element (Customer’s Address Line 3).
4. A line is displayed, visually illustrating the link between the SHIPTOCITY field and the CLO CADD CAD2 element.

SHIPTOSTATE

Complete the following steps to map the customer’s address line 4:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field SHIPTOSTATE in the INVBEGIN record. When you click the field, the cursor changes to a “Link Arrow”.
3. Click the CLO CADD CAD3 element (Customer’s Address Line 4).
4. A line is displayed, visually illustrating the link between the SHIPTOSTATE field and the CLO CADD CAD3 element.

SHIPTOPOSTAL CODE

Complete the following steps to map the customer’s post code:

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

SHIPTOLOC

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

Complete the following steps to map the customer’s own location code:

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

3. Click the CLO CLOC CLO1 element (Customer’s Own Location Code).
4. A line is displayed, visually illustrating the link between the SHIPTOLOC field and the CLO CLOC CLO1 element.

BOLNUM

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

Mapping the delivery note number

Complete the following steps to map the delivery note (bill of lading) number:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field BOLNUM in the INVBEGIN record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the ODD:2 DELN DEL0 element (Delivery Note Number).
4. A line is displayed, visually illustrating the link between the BOLNUM field and the ODD:2 DELN DEL0 element.

SHIPDATE

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

Mapping the date of despatch by supplier

Complete the following steps to map the date of despatch by supplier (ship date):

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

TERMSNETDUE

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

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

Changing the max use of the PYT segment

Complete the following steps to make the PYT a nonrepeating segment:

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

Mapping the number of days

Complete the following steps to map the number of days (terms net due days):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field TERMSNETDUE in the INVBEGIN record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the PYT PAYY PAY2 element (Number of Days).
4. A line is displayed, visually illustrating the link between the TERMSNETDUE field and the PYT PAYY PAY2 element.

TERMSTEXT

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

Mapping the terms of payment

Complete the following steps to map the terms of payment (terms text):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field TERMSTEXT in the INVBEGIN record. When you click the field, the cursor changes to a “Link Arrow”.
3. Click the PYT PAYT element (Terms of Payment).
4. A line is displayed, visually illustrating the link between the TERMSTEXT field and the PYT PAYT element.

INVComment Group

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

INVCOMMENT Record

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

BILLTOACCTCMT

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

DOCTYPCMT

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

INVNUMCMT

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

MESSAGETEXT1

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

Mapping the general narrative line 1

Complete the following steps to map the general narrative line 1 (message text 1):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field MESSAGETEXT1 in the INVCOMMENT record. When you click the field, the cursor changes to a "Link Arrow": LINK ⇨
3. Click the DNA GNAR GNA0 element (General Narrative Line 1).
4. A line is displayed, visually illustrating the link between the MESSAGETEXT1 field and the DNA GNAR GNA0 element.

MESSAGETEXT2

Complete the following steps to map the general narrative line 2 (message text 2):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.

2. Click the application field MESSAGETEXT2 in the INVCOMMENT record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨

3. Click the DNA GNAR GNA1 element (General Narrative Line 2).

4. A line is displayed, visually illustrating the link between the MESSAGETEXT2 field and the DNA GNAR GNA1 element.

MESSAGETEXT3

Complete the following steps to map the general narrative line 3 (message text 3):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.

2. Click the application field MESSAGETEXT3 in the INVCOMMENT record. When you click the field, the cursor changes to a “Link Arrow”.

3. Click the DNA GNAR GNA2 element (General Narrative Line 3).

4. A line is displayed, visually illustrating the link between the MESSAGETEXT3 field and the DNA GNAR GNA2 element.

MESSAGETEXT4

Complete the following steps to map the general narrative line 4 (message text 4):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.

2. Click the application field MESSAGETEXT4 in the INVCOMMENT record. When you click the field, the cursor changes to a “Link Arrow”.

3. Click the DNA GNAR GNA3 element (General Narrative Line 4).

4. A line is displayed, visually illustrating the link between the MESSAGETEXT4 field and the DNA GNAR GNA3 element.

INVDETAIL Record

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

BILLTOACCTDET

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

DOCTYPDET

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

INVNUMDET

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

ITEMNUM

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

Mapping the second level sequence number

Complete the following steps to map the second level sequence number (item number):

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

QTYSHIP

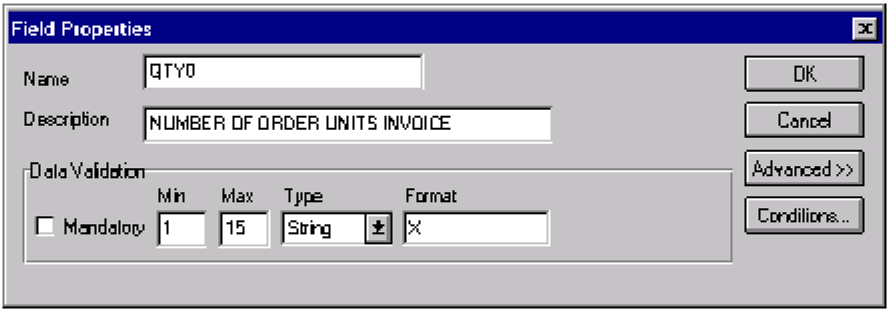
On the EDI side of the map, the quantity shipped information is contained in the ILD:2 QTYI QTY0 element (Number of Order Units Invoiced). The ILD:2 QTYI QTY0 element is defined as a string-type element in the standard. However, you have determined that you will send your partner a number-type element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

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

Changing the properties of the ILD:2 QTYI QTY0 element

Complete the following steps to change the type and format of the ILD:2 QTYI QTY0 element:

1. Double-click the ILD:2 QTYI QTY0 element (Number of Order Units Invoiced) to display the Field Properties dialogue.



Name: QTY0

Description: NUMBER OF ORDER UNITS INVOICE

Data Validation

☐ Mandatory

Min: 1

Max: 15

Type: String

Format: X

2. From the Type drop-down list box, select “Number” to indicate that this is a number element. The system will prompt you with a message that changing the field’s data type will invalidate its format. Click YES to acknowledge this message and proceed.
3. From the Format drop-down list box, select “N0” to indicate that this element will be formatted with an implicit decimal point and no decimal place.

Mapping the number of order units invoiced

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

Complete the following steps to map the quantity shipped:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field QTYSHIP in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the ILD:2 QTYI QTY0 element (Number of Order Units Invoiced).
4. A line is displayed, visually illustrating the link between the QTYSHIP field and the ILD:2 QTYI QTY0 element.

UNITPRICE

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

Complete the following steps to map the cost price (excluding VAT):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field UNITPRICE in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the ILD:2 AUCT AUC0 element (Cost Price (Excluding VAT)).
4. A line is displayed, visually illustrating the link between the UNITPRICE field and the ILD:2 AUCT AUC0 element.

UOM

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

Complete the following steps to map the unit of measure:

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field UOM in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow”.
3. Click the ILD:2 QTYI CD04:2 element (Measure Indicator).
4. A line is displayed, visually illustrating the link between the UOM field and the ILD:2 QTYI CD04:2 element.
5. Again, click the application field UOM in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow”.
6. Click the ILD:2 AUCT CD04:3 element (Measure Indicator).
7. A line is displayed, visually illustrating the link between the UOM field and the ILD:2 AUCT CD04:3 element.

CUSTPROCODE

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

Complete the following steps to map the customer’s item code (customer product code):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.

2. Click the application field CUSTPROCODE in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the ILD:2 CPRO CPR1 element (Customer’s Item Code).
4. A line is displayed, visually illustrating the link between the CUSTPROCODE field and the ILD:2 CPRO CPR1 element.

UPCCODE

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

Complete the following steps to map the supplier’s code for the traded unit (UPC code):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field UPCCODE in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow”.
3. Click the ILD:2 SPRO CPR1 element (Supplier’s Code for the Traded Unit).
4. A line is displayed, visually illustrating the link between the UPCCODE field and the ILD:2 SPRO CPR1 element.

ITEMDESC

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

Complete the following steps to map the traded unit description (item description):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field ITEMDESC in the INVDETAIL record. When you click the field, the cursor changes to a “Link Arrow”.
3. Click the ILD:2 TDES TDE0 element (Traded Unit Descr. Line 1).
4. A line is displayed, visually illustrating the link between the ITEMDESC field and the ILD:2 TDES TDE0 element.

VATCODE

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

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

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

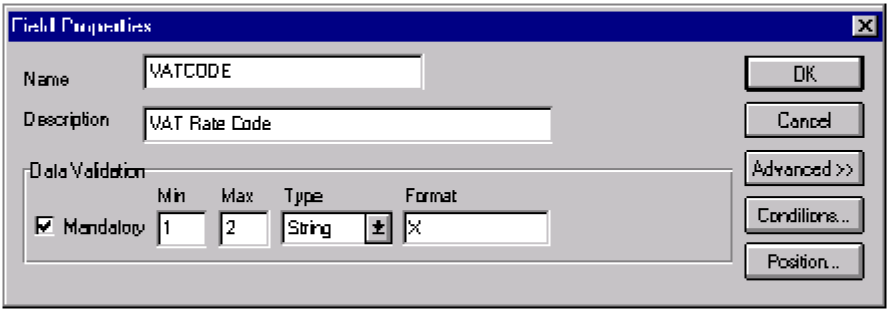
You do not need to create this lookup table (INVATCO.LKP). It is included with the tutorial data that is provided when you install Gentran:Director.

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

Mapping the VAT rate

Complete the following steps to look up the appropriate VAT rate for the specified VAT code:

1. Double-click the VATCODE field (in the INVDETAIL record). The Field Properties dialogue is displayed.



Name: VATCODE

Description: VAT Rate Code

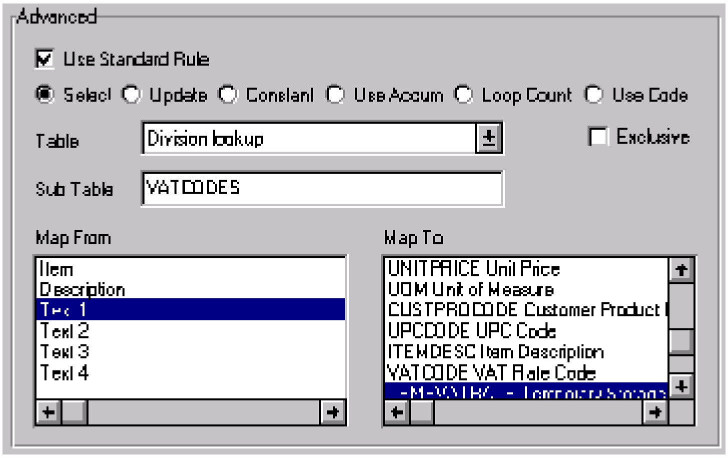
Data Validation

☑ Mandatory Min: 1 Max: 2 Type: String Format: X

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. In the Advanced section, click the Use Standard Rule check box to display the rule options.
4. Verify that the Select radio button is selected.

5. From the Table drop-down list box, select “Division lookup”. This indicates that the key is the partner lookup table name for the <Internal System User> partner and enables you to map from any field on the Lookup Entry dialogue of the Partner Editor.
6. In the Sub Table field, type “VATCODES”. This is the name of the lookup table you imported.
7. From the Map From list box, choose “Text 1”. This is the field on the Lookup Entry dialogue of the Partner Editor from which the contents will be mapped.
8. From the Map To list box, choose “TEMPVATRATE”. This is the field to which the information from the Partner Editor will be mapped.

   The Advanced section of the Field Properties dialogue should now look like this:



Advanced

☑ Use Standard Rule

● Select ○ Update ○ Constant ○ Use Accum ○ Loop Count ○ Use Code

Table: Division lookup

□ Exclusive

Sub Table: VATCODES

Map From:
Item
Description
Text 1
Text 2
Text 3
Text 4

Map To:
UNITPRICE Unit Price
UOM Unit of Measure
CUSTPRODCODE Customer Product I
UPCCODE UPC Code
ITEMDESC Item Description
VATCODE VAT Rate Code
TEMPVATRA - Temporary Storage

9. Click OK to map from the lookup table to the TEMPVATRATE field.

Mapping the VAT rate category code

Complete the following steps to map the VAT rate category code (VAT code):

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

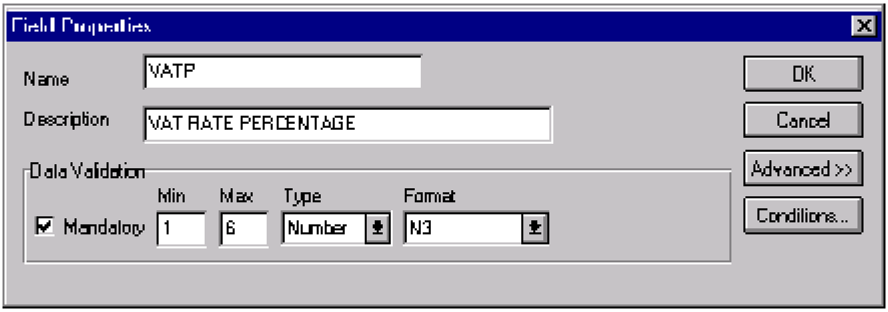
TEMPVATRATE

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

Using an extended rule to convert the type of the TEMPVATRATE field and map it

Complete the following steps to create an extended rule to convert the TEMPVATRATE field from a string-type to a number-type field and map the TEMPVATRATE field to the ILD:2 VATP element (VAT rate percentage):

1. Double-click the ILD:2 VATP element (VAT rate percentage) to access the Field Properties dialogue.



Field Properties

Name: VATP

Description: VAT RATE PERCENTAGE

Data Validation

☑ Mandatory Min: 1 Max: 6 Type: Number Format: N3

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.

3. Click RULE... to display the Field Level Extended Rule dialogue.

4. In the Rule list box, type the following:

   #VATP = ATON($INVDETAIL.#TEMPVATRATE);

   The “ATON” is a numerical function that converts strings into real numbers, which enables you to convert a string-type field to a number-type field. The “#VATP = ” maps the resulting real number to the ILD:2 VATP element. The system will execute this rule when it processes the ILD:2 VATP element.

5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile

each rule individually, so that you can verify the accuracy of the rule after you create it.

The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list box.

6. Click OK to add the extended rule to the ILD:2 VATP element.
7. Click CLOSE to exit the Field Properties dialogue.

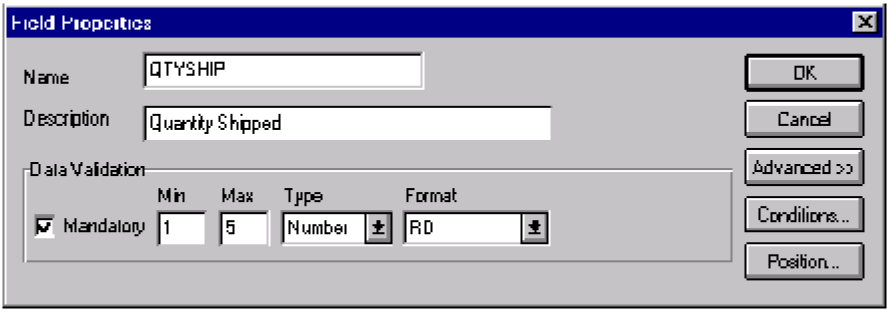
TEMPLEXC

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

Calculating the extended price

Complete the following steps to multiply the quantity shipped for each line item by the unit price to obtain the extended price:

1. Double-click the QTYSHIP field (in the INVDETAIL segment). The Field Properties dialogue is displayed.



Name: QTYSHIP

Description: Quantity Shipped

Data Validation

☑ Mandatory

Min	Max	Type	Format
1	5	Number	RD

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section.
3. In the Advanced section, click the Use Standard Rule check box to display the rule options.
4. Click the Use Accum radio button.
5. Click NEW... to access the Edit Accumulator Entry dialogue to create a new calculation for this element.
6. From the Primary Accumulator drop-down list box, select “0”. This number is a name by which the accumulator will be referred. Before any calculations are performed on an accumulator, its content is “0” (zero).

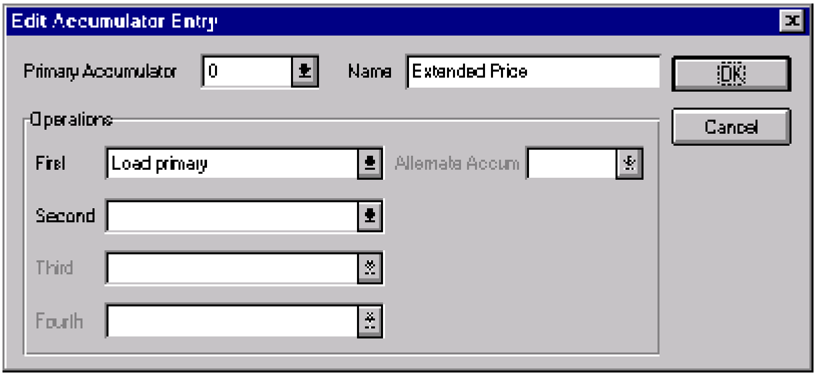
When you use an accumulator, the system adds a new accumulator to the bottom of this drop-down list box.

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

7. In the Name field, type "Extended Price". This is a descriptive alias that will enable you to differentiate what the accumulators you create are used for.
8. From the First drop-down list box, select "Load primary". This is the first operation that the system will perform. This specifies that the system will load the contents of the field into the Primary Accumulator for each iteration of the INVDETAIL segment.

The First field is active only after you select a Primary Accumulator.

The Edit Accumulator Entry dialogue should now look like this:



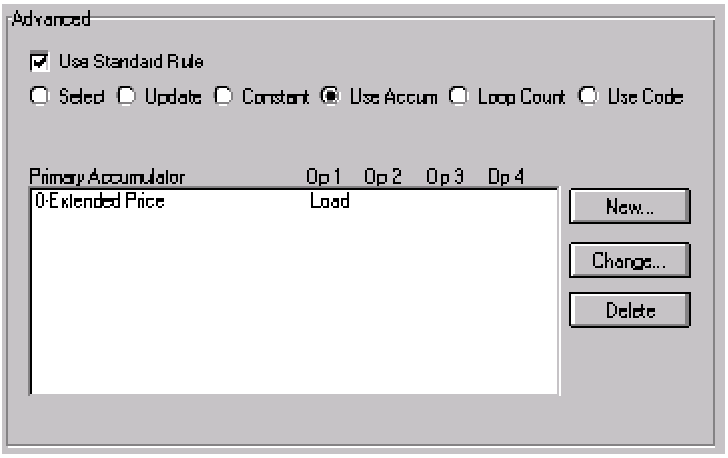
Primary Accumulator: 0

Name: Extended Price

First: Load primary

9. Click OK to add the accumulator.

The Advanced section of the Field Properties dialogue should now look like this:



Advanced

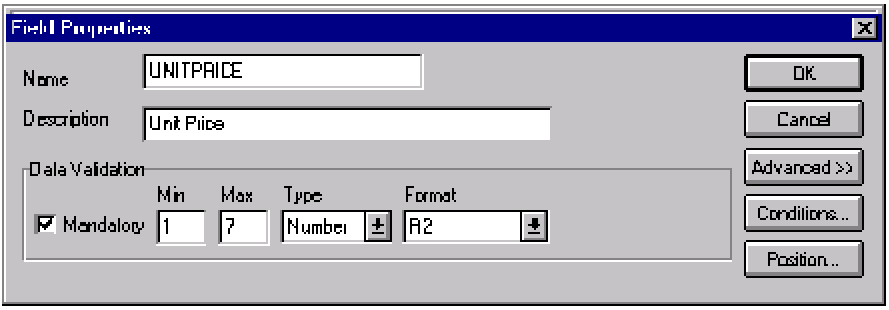
☑ Use Standard Rule

○ Select ○ Update ○ Constant ● Use Accum ○ Loop Count ○ Use Code

Primary Accumulator	Op 1	Op 2	Op 3	Op 4
0-Extended Price	Load			

10. Click OK on the Field Properties dialogue to add the standard rule to the QTYSHIP field.

11. Double-click the UNITPRICE field (in the INVDETAIL segment). The Field Properties dialogue is displayed.



Field Properties

Name: UNITPRICE

Description: Unit Price

Data Validation

☑ Mandatory Min: 1 Max: 7 Type: Number Format: R2

12. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.

13. In the Advanced section, click the Use Standard Rule check box to display the rule options.

14. Click the Use Accum radio button.

15. Click NEW... to access the Edit Accumulator Entry dialogue to create a new calculation for this element.

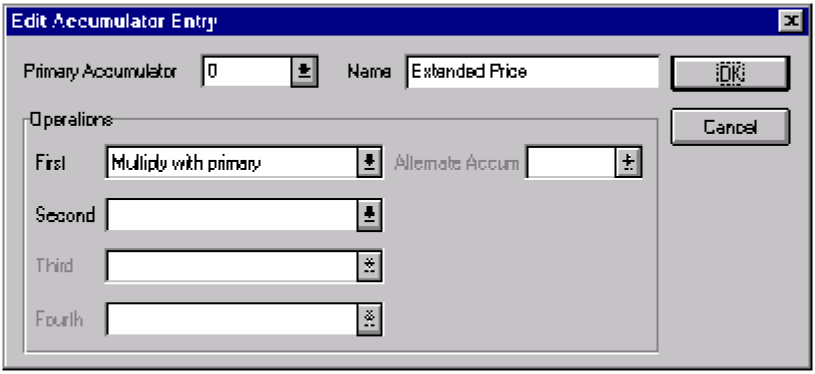
16. From the drop-down list box, select primary accumulator “0”.

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

17. From the First drop-down list box, select "Multiply with primary". This is the first operation that the system will perform. This specifies that the system will multiply the value of the UNITPRICE field with the contents of the primary accumulator, and store the result in the primary accumulator for each iteration of the INVDETAIL segment.

The First field is active only after you select a Primary Accumulator.

The Edit Accumulator Entry dialogue should now look like this:


Edit Accumulator Entry

Primary Accumulator: 0

Name: Extended Price

Operations

First: Multiply with primary

Alternate Accum:

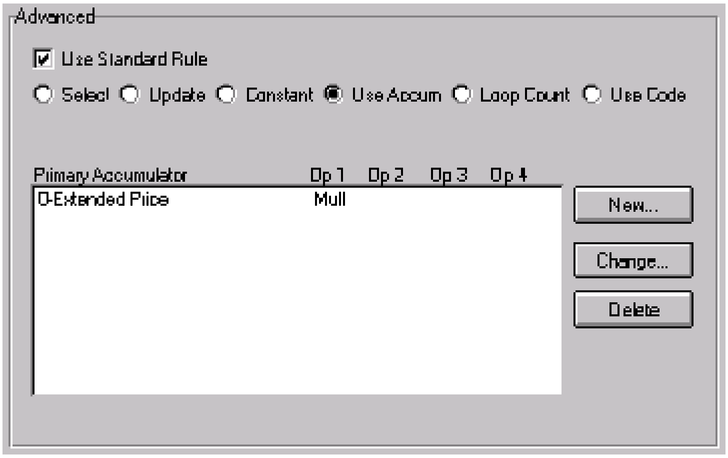
Second:

Third:

Fourth:

18. Click OK to add the accumulator.

The Advanced section of the Field Properties dialogue should now look like this:



Advanced

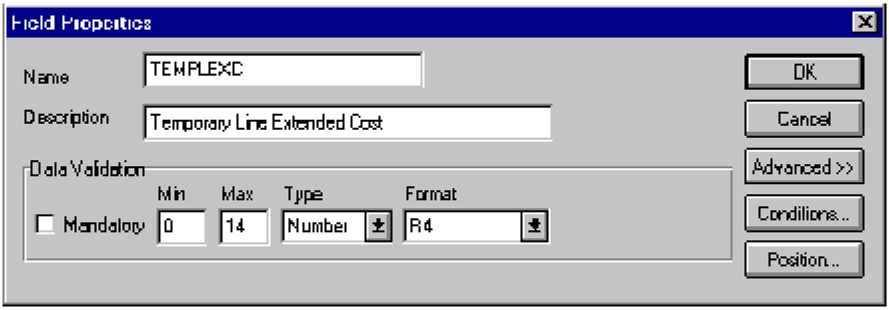
☑ Use Standard Rule

○ Select ○ Update ○ Constant ● Use Accum ○ Loop Count ○ Use Code

Primary Accumulator	Op 1	Op 2	Op 3	Op 4
0-Extended Price	Null			

19. Click OK on the Field Properties dialogue to add the standard rule to the UNITPRICE field.

20. Double-click the TEMPLEXC field (in the INVDETAIL segment). The Field Properties dialogue is displayed.



Field Properties

Name: TEMPLEXC

Description: Temporary Line Extended Cost

Data Validation

☐ Mandatory Min: 0 Max: 14 Type: Number Format: R4

21. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.

22. In the Advanced section, click the Use Standard Rule check box to display the rule options.

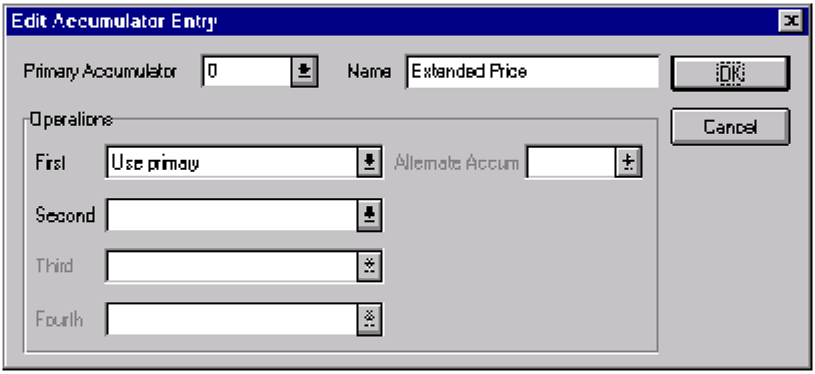
23. Click the Use Accum radio button.

24. Click NEW... to access the Edit Accumulator Entry dialogue to create a new calculation for this element.

25. From the drop-down list box, select primary accumulator “0”.

26. From the First drop-down list box, select “Use primary”. This specifies that the system will load the contents of the primary accumulator into the TEMPLEXC field.

    The Edit Accumulator Entry dialogue should now look like this:



Edit Accumulator Entry

Primary Accumulator: 0

Name: Extended Price

Operations

First: Use primary

Alternate Accum:

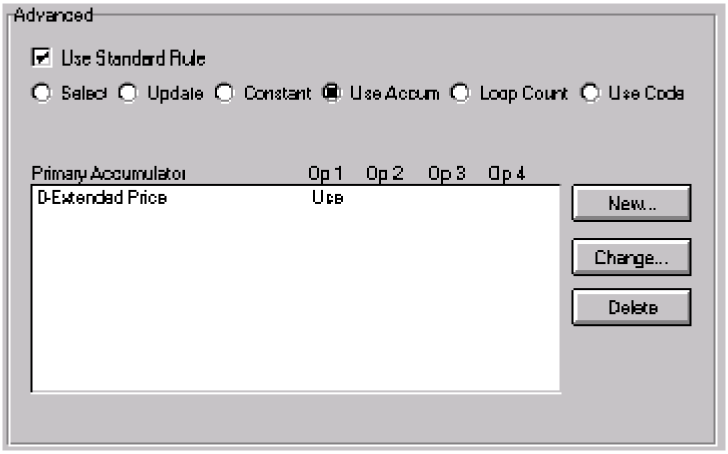
Second:

Third:

Fourth:

27. Click OK to add the accumulator.

    The Advanced section of the Field Properties dialogue should now look like this:



Advanced

☑ Use Standard Rule

○ Select ○ Update ○ Constant ● Use Accum ○ Loop Count ○ Use Code

Primary Accumulator	Op 1	Op 2	Op 3	Op 4
0-Extended Price	Use			

28. Click OK on the Field Properties dialogue to add the standard rule to the TEMPLEXC field. This loads the extended price for the current line item into the TEMPLEXC field.

Mapping the extended line cost (excluding VAT)

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

Complete the following steps to map the extended line cost (excluding VAT):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field TEMPLEXC in the INVDETAIL record. When you click the field, the cursor changes to a "Link Arrow": LINK ⇨
3. Click the ILD:2 LEXC element (Extended Line Cost (Excluding VAT)).
4. A line is displayed, visually illustrating the link between the TEMPLEXC field and the ILD:2 LEXC element.

TEMPVATSTL Record

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

BILLTOACCTTS

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

DOCTYPTS

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

INVNUMTS

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

SEQA

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

Mapping the first level sequence number

Complete the following steps to map the first level sequence number (SEQA):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field SEQA in the TEMPVATSTL record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the STL SEQA:7 element (First Level Sequence Number).
4. A line is displayed, visually illustrating the link between the SEQA field and the STL SEQA:7 element.

VATRATE

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

Mapping the VAT rate percentage

Complete the following steps to map the VAT rate percentage (VATRATE):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field VATRATE in the TEMPVATSTL record. When you click the field, the cursor changes to a “Link Arrow”.
3. Click the STL VATP:2 element (VAT Rate Percentage).
4. A line is displayed, visually illustrating the link between the VATRATE field and the STL VATP:2 element.

VATRATECODE

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

Mapping the VAT rate category code

Complete the following steps to map the VAT rate category code (VATRATECODE):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field VATRATECODE in the TEMPVATSTL record. When you click the field, the cursor changes to a “Link Arrow”.

3. Click the STL CD12:2 element (VAT Rate Category Code).
4. A line is displayed, visually illustrating the link between the VATRATECODE field and the STL CD12:2 element.

NUMLINES

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

Mapping the number of line items

Complete the following steps to map the number of line items (NUMLINES):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field NUMLINES in the TEMPVATSTL record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the STL NRIL element (Number of Line Items).
4. A line is displayed, visually illustrating the link between the NUMLINES field and the STL NRIL element.

LVLA

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

Mapping the line subtotal amount

Complete the following steps to map the line subtotal amount (LVLA):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field LVLA in the TEMPVATSTL record. When you click the field, the cursor changes to a “Link Arrow”.
3. Click the STL LVLA element (Lines Sub-Total (Excluding VAT)).
4. A line is displayed, visually illustrating the link between the LVLA field and the STL LVLA element.

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

VATA

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

Mapping the VAT amount payable

Complete the following steps to map the VAT Amount Payable (VATA):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field VATA in the TEMPVATSTL record. When you click the field, the cursor changes to a "Link Arrow".
3. Click the STL VATA element (VAT Amount Payable).
4. A line is displayed, visually illustrating the link between the VATA field and the STL VATA element.

INVSUMMARY Record

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

BILLTOACCTSUM

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

DOCTYPSUM

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

INVNUMSUM

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

TOTALAMOUNT

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

Mapping the total payable including settlement

Complete the following steps to map the total payable including settlement (total amount):

1. If the Linking function is not currently active, click the Link button on the Main Toolbar to turn on the Linking function.
2. Click the application field TOTALAMOUNT in the INVSUMMARY record. When you click the field, the cursor changes to a “Link Arrow”: LINK ⇨
3. Click the TLR TPSI element (Total Amount Inc Settlement).
4. A line is displayed, visually illustrating the link between the TOTALAMOUNT field and the TLR TPSI element.

Producing the VAT Summary

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

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

The following is an example of an application file that contains six line items that use three different VAT codes: “S” (Standard rate), “E” (Export item), and “Z” (Zero rate).

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

The following are the corresponding VAT percentage rates for the VAT codes used in the above example:

VAT Code	VAT Percentage Rate
S	17.5
E	0
Z	0

From this, you can calculate the required VAT summary information:

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

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

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

The extended rules are indicated with bold typeface (Â is used for a line continuation character).

1. You need to define two temporary variables (“i” and “j”), a variable defined to count the number of line items (“invlimit”), and a variable to count the number of STL segments (“stllimit”) that is initialised to zero.

   ```
   integer i,j,invlimit,stllimit;
   invlimit = count($INVDETAIL[*]);
   stllimit = 0;
   ```

   You need to visit each iteration of the INVDETAIL record.

   ```
   i=1;
   while i <= invlimit do
   begin
   .
   .
   i=i+1;
   end
   ```

2. For each iteration of the INVDETAIL record, you need to determine the VAT code used, and ascertain whether or not a TEMPVATSTL (STL) record exists for that VAT code.

   ```
     j=1;
     while j <= stllimit do
     begin
       if $INVDETAIL[i].#VATCODE =
      Â $TEMPVATSTL[j].#VATRATECODE then
      begin
      .
      .
      end
       j=j+1;
     end
   ```

3. If a TEMPVATSTL record does exist for that VAT code, you need to add the extended cost for that line item into the total extended cost for this STL, and increase the number of line items in this STL by one.

   ```
   begin
   $TEMPVATSTL[j].#LVLA = $TEMPVATSTL[j].#LVLA +
      Â$INVDETAIL[i].#TEMPLEXC;
   $TEMPVATSTL[j].#NUMLINES =
      Â$TEMPVATSTL[j].#NUMLINES + 1;
   break;
   end
   ```

4. If a TEMPVATSTL record does not exist for that VAT code, you need to create an STL for that VAT code, set the sequence number to the line item number, set the VAT rate percentage, set the STL number, and set the total extended cost to be the extended cost of that line item.

   ```
   $TEMPVATSTL[j].#SEQA = j;
   $TEMPVATSTL[j].#VATRATE =
      ATON($INVDETAIL[i].#TEMPVATRATE);
   $TEMPVATSTL[j].#VATRATECODE =
   $INVDETAIL[i].#VATCODE;
   $TEMPVATSTL[j].#LVLA = $INVDETAIL[i].#TEMPLEXC;
   $TEMPVATSTL[j].#NUMLINES = 1;
   ```

5. Finally, you need to calculate the VAT amount that is payable for each STL.

   ```
   j=1;
   while j <= stllimit do
   begin
   $TEMPVATSTL[j].#VATA = ($TEMPVATSTL[j].#LVLA *
      Â$TEMPVATSTL[j].#VATRATE) / 100.0;
   j = j + 1;
   end
   ```

Using an extended rule to perform VAT calculations

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

Complete the following steps to create the extended rule:

1. Click the INPUT group with the right mouse button to access the shortcut menu.

2. From the shortcut menu, select Extended Rules... to display the Loop Level Extended Rules dialogue.
3. Click the On End radio button to specify that the rule is executed when the loop terminates. The system executes this rule when it has concluded its processing of the application file.
4. In the Editor list box, type the following (or copy from the PET_TINV.MAP provided for you with the tutorial data in the GENTRAN Director\TUTORIALS subfolder):

```
integer i,j,invlimit,stllimit;
invlimit = count($INVDETAIL[*]);
stllimit = 0;
i=1;
while i <= invlimit do
begin
  j=1;
  while j <= stllimit do
  begin
    if $INVDETAIL[i].#VATCODE =
$TEMPVATSTL[j].#VATRATECODE then
    begin
      $TEMPVATSTL[j].#LVLA = $TEMPVATSTL[j].#LVLA +
         Â$INVDETAIL[i].#TEMPLEXC;
      $TEMPVATSTL[j].#NUMLINES =
         Â$TEMPVATSTL[j].#NUMLINES + 1;
      break;
    end
    j=j+1;
  end
  if j > stllimit then
  begin
    $TEMPVATSTL[j].#SEQA = j;
    $TEMPVATSTL[j].#VATRATE =
     ÂATON($INVDETAIL[i].#TEMPVATRATE);
    $TEMPVATSTL[j].#VATRATECODE =
     Â$INVDETAIL[i].#VATCODE;
    $TEMPVATSTL[j].#LVLA = $INVDETAIL[i].#TEMPLEXC;
    $TEMPVATSTL[j].#NUMLINES = 1;
    stllimit = j;
  end
  i=i+1;
end
j=1;
while j <= stllimit do
```

```
begin
  $TEMPVATSTL[j].#VATA = ($TEMPVATSTL[j].#LVLA *
    Â$TEMPVATSTL[j].#VATRATE) / 100.0;
  j = j + 1;
end
```

5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

   The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list box.

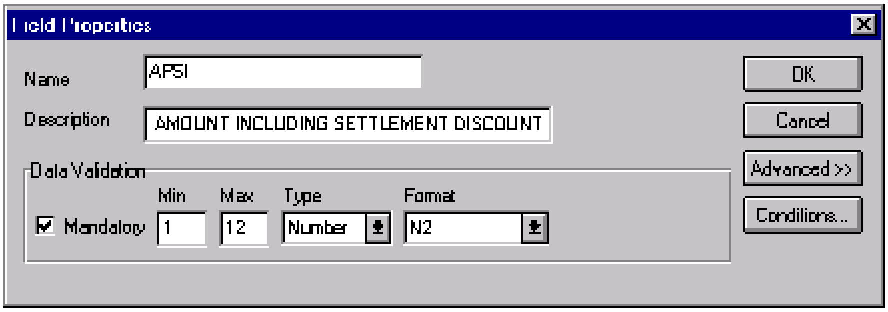
6. Click OK to add the extended rule to the INPUT group (application file).

Using an extended rule to calculate the amount payable for each STL

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

Complete the following steps to create the extended rule:

1. Double-click the STL APSI element (Payable Sub-Total Amount Including Settlement Discount) to access the Field Properties dialogue.



Field Properties

Name: APSI

Description: AMOUNT INCLUDING SETTLEMENT DISCOUNT

Data Validation

☑ Mandatory Min: 1 Max: 12 Type: Number Format: N2

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.

3. Click RULE... to display the Field Level Extended Rule dialogue.

4. In the Rule list box, type the following (or copy from the PET_TINV.MAP provided for you with the tutorial data in the C:\Program Files\Sterling Commerce\GENTRAN Director\TUTORIALS\ subfolder):

   #APSI = #ASDA + #VATA;

   The system will execute this rule when it processes the STL APSI element.

5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

   The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list box.

6. Click OK to add the extended rule to the STL APSI element.

7. Click CLOSE to exit the Field Properties dialogue.

Producing the Invoice Trailer

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

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

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

The extended rules are indicated with bold typeface (Â is used for a line continuation character).

1. You need to define a variable to count the number of TEMPVATSTL records (“limit”), define a temporary variable (“i”) that is set to one, and initialise the TLR LVLT and TLR TVAT elements to zero.

   ```
   integer i,limit;
   limit = count($TEMPVATSTL[*]);
   i=1;
   #LVLT = 0;
   #TVAT = 0;
   ```

2. You need to visit each iteration of the TEMPVATSTL record.

   ```
   while i <= limit do
   begin
   i = i +1;
   end
   ```

3. You need to calculate the total amount (extended cost) of all the line items in the invoice (LVLT) and the total VAT amount payable for all the line items in the invoice (TVAT).

   ```
   #LVLT = #LVLT + $TEMPVATSTL[i].#LVLA;
   #TVAT = #TVAT + $TEMPVATSTL[i].#VATA;
   ```

4. You need to set the NSTL (Number of Total Segments) equal to the number of TEMPVATSTL records.

   ```
   #NSTL = limit;
   ```

5. You need to set both the EVLT (Total Extended Amount Excluding Settlement Discount) and the ASDT (Total Amount Excluding VAT Including Settlement) equal to the LVLT (Lines Total Amount (Excluding VAT)).

   ```
   #EVLT = #LVLT;
   #ASDT = #LVLT;
   ```

6. Finally, you need to set the TPSI equal to the ASDT (Total Amount Excluding VAT Including Settlement) plus the TVAT (Total VAT Amount Payable).

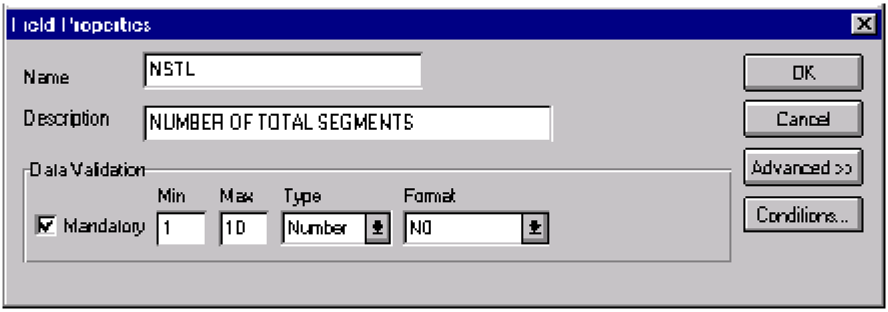
   ```
   #TPSI = #ASDT + #TVAT;
   ```

Using an extended rule to calculate TLR totals

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

Complete the following steps to create the extended rule:

1. Double-click the TLR NSTL element (Number of Total Segments) to access the Field Properties dialogue.



Name: NSTL

Description: NUMBER OF TOTAL SEGMENTS

Data Validation

	Min	Max	Type	Format
☑ Mandatory	1	10	Number	N0

2. If the Advanced section of the dialogue is not currently displayed, click ADVANCED>> to access this section, which enables you to define rules.
3. Click RULE... to display the Field Level Extended Rule dialogue.
4. In the Rule list box, type the following:

```
integer i,limit;
limit = count($TEMPVATSTL[*]);
i=1;
#LVLT = 0;
#TVAT = 0;

while i <= limit do
begin
  #LVLT = #LVLT + $TEMPVATSTL[i].#LVLA;
  #TVAT = #TVAT + $TEMPVATSTL[i].#VATA;
  i=i+1;
end

#NSTL = limit;
#EVLT = #LVLT;
#ASDT = #LVLT;
#TPSI = #ASDT + #TVAT;
```

The system will execute this rule when it processes the TLR NSTL element.

5. Click COMPILE to validate the syntax of the extended rule. Every rule in the map will be compiled when you compile the translation object, after you complete the map. However, the system allows you to compile each rule individually, so that you can verify the accuracy of the rule after you create it.

   The Compile function will compile the rule interactively, and allow you to correct any errors that are generated. Any errors or warnings generated in the compilation process will be displayed in the Compile Errors list box.

6. Click OK to add the extended rule to the TLR NSTL element.
7. Click CLOSE to exit the Field Properties dialogue.

Finalising the Map

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

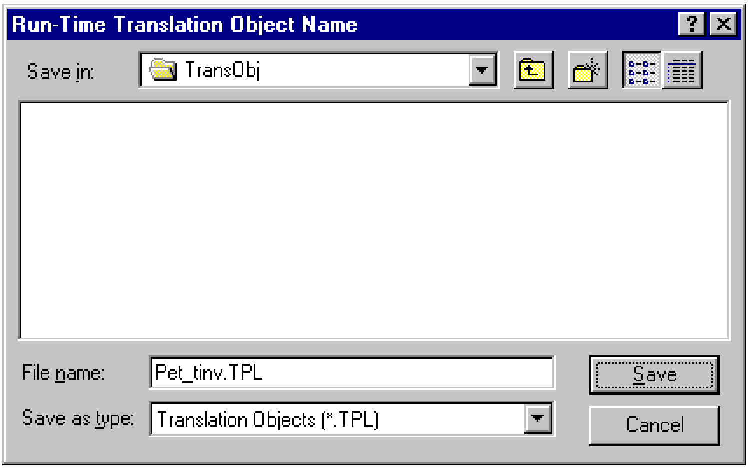
Compiling the Map

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

This translation object will need to be registered with the Gentran:Director system before you can use it. You will use the Compile function after the map is completed and saved.

Complete the following steps to compile a map and generate a translation object:

1. Select Save from the File menu to save the source map, prior to using the Compile function.
2. Select Compile from the File menu to display the Run-Time Translation Object Name dialogue.



Save in: TransObj

File name: Pet_tinv.TPL

Save as type: Translation Objects (*.TPL)

3. The “File name” field will already contain “PET_TINV.TPL”. This is the name of the translation object, using the default .TPL file extension and our recommendation that you name the translation object (.TPL

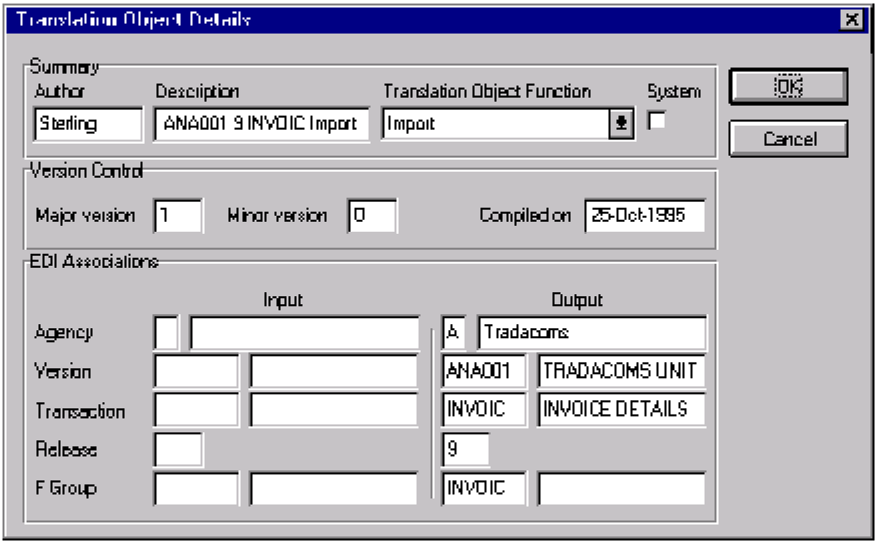
file) the same filename (one to eight characters long) as you named the map (.MAP file). Preserving the same filename (with different file extensions) means that the relationship between the source map and the compiled translation object will remain evident.

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

4. Change the Drive and Folder where the compiled translation object will be stored, if necessary.

STOP **Do *not* store the compiled translation object in the GENTRAN Director\RegTransObj subfolder. This subfolder is reserved for storing a copy of each translation object you register with Gentran:Director.**

5. Click SAVE and the system will compile the map and generate a translation object. The Compile Error dialogue is displayed. Verify that no errors occurred and click OK to exit the dialogue. The date on which the translation object was compiled will be automatically loaded into the Compiled on field on the Translation Object Details dialogue.



Translation Object Details

Summary

Author	Description	Translation Object Function	System
Sterling	ANA001 9 INVOIC Import	Import	□

Version Control

Major version: 1 Minor version: 0 Compiled on: 25-Oct-1995

EDI Associations

	Input		Output	
Agency			A	Tradacoms
Version			ANA001	TRADACOMS UNIT
Transaction			INVOIC	INVOICE DETAILS
Release			9	
F Group			INVOIC	

6. Select Save from the File menu to save the source map with the Compiled on date.

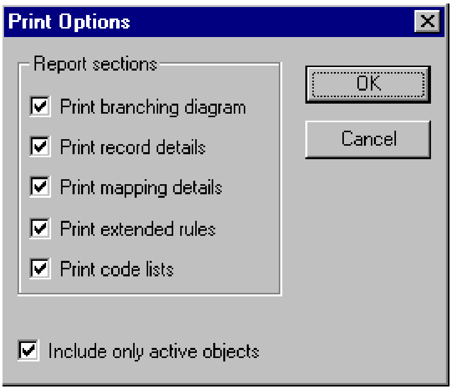
7. You must register this translation object with the Gentran:Director system before you can use it. See the Gentran:Director documentation for information on registering a translation object.

Printing the Report

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

Complete the following steps to print the Gentran:Director report:

1. Select Print from the File menu or the Main Toolbar to access the Print Options dialogue.



☑ Print branching diagram
☑ Print record details
☑ Print mapping details
☑ Print extended rules
☑ Print code lists
☑ Include only active objects

2. Accept the defaults on this dialogue (all options checked), so the report will include all sections and only the activated map components. Click OK and the Print dialogue is displayed.
3. Set the appropriate options. Click OK if you do *not* need to change Setup options, and the mapping report will be printed.

   Click SETUP... if you need to access printer setup information (to select a specific printer, paper orientation, or size and source of paper). The Print Setup dialogue is displayed as follows:
4. Set the appropriate options. Click OK if you do *not* need to change Print Options (dithering, intensity control, or print true type as graphics), and you will return to the Print dialogue. Click OK on the Print dialogue and the mapping report will be printed.

   Click OPTIONS... if you need to access printer setup options information. The Options dialogue is displayed.
5. Set the appropriate options. Click OK to return to the Print Setup dialogue. Click OK on the Print Setup dialogue to return to the Print dialogue. Click OK on the Print dialogue and the report will be printed.

Testing the Translation Object

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

Complete the following steps to test the system import (HEADER_T.TPL) and import (PET_TINV.TPL) translation objects:

1. Register the HEADER_T.TPL and PET_TINV.TPL translation objects with Gentran:Director. Please see the Gentran:Director *User's Guide* for more information on registering a translation object.
2. If you have not already done so, import the PETZONE3.PAR partner relationship into Gentran:Director. Please see the Gentran:Director *User's Guide* for more information on importing a partner.

   The default full path for the partner file is: C:\Program Files\Sterling Commerce\GENTRAN Director\TUTORIAL\PETZONE3.PAR
3. Import the division lookup table (INVATCO.LKP) for the system partner (<Internal System User>) into Gentran:Director. Please see the Gentran:Director *User's Guide* for more information on importing a table.

   The default full path for the lookup table is: C:\Program Files\Sterling Commerce\GENTRAN Director\TUTORIAL\INVATCO.LKP
4. Verify (in Partner Editor) that the PET T ANA001 9 INVOIC Import translation object is selected for the Outbound Relationship. Please see the Gentran:Director *User's Guide* for more information on selecting outbound translation objects.
5. Ask your system administrator to add the system import translation object (HEADER_T.TPL) to the System Configuration program. Please see the *Administration Guide* for more information about how your system administrator will need to modify the system configuration program.
6. Use the Import option in Gentran:Director to process the data file (PET_TINV.TXT) through the translation object. The data file will be located in the TUTORIAL subfolder under the folder where Gentran:Director is installed.

   The default full path for the data file is: C:\Program Files\Sterling Commerce\GENTRAN Director\TUTORIAL\PET_TINV.TXT

Please see the Gentran:Director *User's Guide* for more information on importing files.

7. After the message is translated, it will be located in the Workspace in Gentran:Director. View the EDI data to ensure that the message was translated correctly. Please see the Gentran:Director *User's Guide* for more information on viewing raw EDI data.

GLOSSARY

Glossary

?In Documents This Gentran:Director browser contains a list of documents that were received by the system but failed compliance checking or that do not have an identifiable partner or transaction set.

?Out Documents This Gentran:Director browser contains a list of documents that were imported into the system via unattended processing but are invalid.

acknowledgement This term is used to indicate the ANSI 997 functional acknowledgement, the EDIA 999 acceptance/rejection advice, and the EDIFACT CNTRL document.

activation This function enables you to make map components available for use. The system activates all the groups, segments, composites, and elements that are defined as “mandatory” (must be present) by the standard. The system does not enable you to deactivate the mandatory groups, segments, composites, and elements. When translating data, the system does not process groups, segments, composites, and elements (or records and fields) that are not activated. Therefore, *you* must activate the groups, segments, composites, and elements that are not defined as mandatory by the standard, but that you have determined that you need to use in mapping.

AIAG The Automotive Industry Action Group (AIAG) is the standards-setting group for the automotive industry. The standards form a subset of the ANSI X12 standard.

ANA Article Numbering Association.

ANSI American National Standards Institute. ANSI sets standards for many products and services, such as safety glasses and battery capacities. The ANSI X12 committee is the chief EDI standards-setting organization for the United States.

application file If you are creating an import or export map, you must define your application to Gentran:Director. In Gentran:Director terminology, your application file is also referred to as a *fixed-format file* or a *positional file*. Your application file must contain all the information that you either need to extract from your partner’s document (if the map is inbound) or need to send to your partner (if the map is outbound), so that your system can accurately process the data.

application system Computer systems, outside of EDI, designed to fulfill specific business functions. These include accounting, purchasing, materials control, human resources, shipping, and other systems.

browser A browser is a window that displays a list of items, such as documents (transaction sets), interchanges, translation objects, etc. A document browser displays only documents.

button A button enables you to execute a command. You select a button either by clicking it or by pressing **TAB** until the button you want is selected and then pressing **ENTER**. Unavailable buttons are dimmed, and the currently selected button has a darker border than the other buttons.

check box A check box enables you to select or clear options from a list. You can select as many options from the list as you need. A selected check box contains a checkmark. The labels of unavailable options are dimmed.

click Click indicates that you should position your cursor on top of the designated item and press and release your primary mouse button. The primary mouse button is the button you access with your index finger, regardless of whether you are utilizing right-handed or left-handed mouse operation.

code list tables These are used by EDI standards as repositories for lists of codes. Each EDI standard provides a code list for each element that can be further defined with a code.

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

communications session Everything sent and received to/from one telephone number in one continuous period of connection. This could include sending two or three interchange envelopes to a network, each for a different trading partner.

Communicator This is the Gentran:Director communications software. The Communicator enables you to send, receive, resend, establish communication port definitions, establish communication profiles, establish communication scripts, and view and delete communication sessions.

compile This function compiles the map and creates a translation object. The map that you create using Gentran:Director is a *source map*. When that source map is compiled, the result is a *compiled translation object*. This translation object will need to be registered with the Gentran:Director system before you can use it.

compliant This means that the document conforms to the EDI standards as defined by the translation object.

component data element A simple data element that belongs to a composite data element. Component data elements are also called sub-elements.

composite data element A contains two or more component data elements or sub-elements. Composites are defined by the EDI standards that use them (EDIFACT, TRADACOMS, and certain ANSI X12 standards).

conditions See relational conditions.

constant This standard rule enables you to move a literal constant value to the specified element or field, indicate a qualifying relationship with another element or field, and map the current date or time to the specified element or field.

cross-reference table This table is created in Partner Editor and is used to convert your values to your trading partner's values during outbound processing, or your partner's values to your values during inbound processing.

date/time This type of field or element contains a date or time. If you specify that a field or element will be a date/time type, you must specify exactly how the date or time must be formatted.

deactivation This function enables you to make map components unavailable for use by the system. The system does not enable you to deactivate the mandatory groups, segments, composites, and elements.

default A predefined value – programs use these built-in values *unless* you specifically override them.

dialog box A dialog box (or dialog) contains additional information or options that you need to act on. If you specify any options, you can select a button to execute the command. Some dialog boxes display warnings or messages indicating why a requested task cannot be accomplished. If the dialog box has a title bar, you can move it to another location on your desktop.

document One transaction set containing actual data and treated as a single entity. The amount of data does not affect whether something is a document or not, only the fact that it is treated as a single entity. For example, regardless of whether a purchase order contains one item or ten thousand, if it is one purchase order, it is one document.

If a trading partner sends you a communication containing 10 purchase orders, you have received 10 documents. If the communication contained 15 invoices, you have received 15 documents.

double-click This indicates that you should position your cursor on top of the designated item and quickly press and release your primary mouse button twice. The primary mouse button is the button you access with your index finger, regardless of whether you are utilizing right-handed or left-handed mouse operation.

EANA International Article Numbering Association, Brussels, Belgium.

EDI Electronic Data Interchange (EDI) is the process by which companies can exchange business documents directly from application to application by computer without paper documents being produced.

EDI standard This is the rules for turning a business document into an EDI document.

EDIA The Electronic Data Interchange Association, formerly known as the Transportation Data Coordinating Committee (TDCC).

EDIFACT This is the standards organization of the United Nations Economic and Social Council. The acronym is short for the Electronic Data Interchange for Administration, Commerce, and Transport.

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

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

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

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

There are three levels of envelopes, as follows:

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

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

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

A communications session could easily include a number of interchange envelopes. For this reason, the standards-setting bodies are considering a fourth level of envelope to cover an entire communications session.

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

export map This map defines how to move data from the EDI standard-formatted documents that your partners send you to your application file (flat file definition). An export map is necessary for inbound processing.

export translation object See export map.

extended rules These rules enable you to use a Gentran:Director proprietary programming language to perform virtually any mapping operation you require.

field The smallest piece of information defined in the application file. A field is the application map component that is mapped (linked) to a corresponding EDI element to move data to and from your application file.

fixed-format file See application file.

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

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

group A looping structure that contains related records/segments and/or groups that repeat in sequence until either the group data ends or the maximum number of times that the loop is allowed to repeat is exhausted. If you create a group that is subordinate to another group (a sub-group), this corresponds to a nested looping structure (a loop within a loop). The application (positional) file and the EDI file are both groups and therefore, they is visually represented the same way as other groups and sub-groups in Gentran:Director.

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

import map This map defines how to move data from your application file (flat file definition), which may contain multiple documents, to the EDI standard-formatted documents that your partners expect to receive from you. An import map is necessary for outbound processing.

import translation object See import map.

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

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

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

integer An integer is a number that has an implied decimal point (e.g., “2.01” is formatted as “201”).

interchange The interchange contains all the functional groups of documents (transaction sets) sent from one sender to one receiver in the same transmission.

Interchanges browser This Gentran:Director browser enables you to view a hierarchical list of all the interchanges that were sent or received. It contains status information about the interchanges, (e.g., whether or not a functional acknowledgement was received and the status of that acknowledgement). This browser also enables you to access the documents within the interchanges. The Interchanges browser can always be accessed.

link This function enables you to map a field or element from the Input side of the map to a field or element on the Output side of the map. The link between two map components (hereafter referred to as fields) is visually represented with a line connecting the two fields.

list box A list box displays a list of choices. Scroll bars are provided if there are more choices than can fit in the box.

literal constant These are used by the system as a repository to store information that will be used at a later point in the map. Typically, constants are used in an outbound map to generate a qualifier.

location table This is a table created in Partner Editor that is used to contain address-related information about the partner.

lookup table This is a table created in Partner Editor that is used to select information related to a value in inbound or outbound data.

loop count This standard rule enables you to count the number of times a loop is repeated, if the element or field is part of a loop. If the loop is a nested loop, you can track the current loop *or* the outer loop. For example, if the Y loop is nested within the X loop, and the Y loop has cycled through 15 iterations and the X loop has cycled through 3 iterations, you can choose to count either the “15” (Y loop) or the “3” (X loop).

loop start/loop end Certain EDI standards use Loop Start (LS) and Loop End (LE) segments. LS and LE segments differentiate between two or more loops of the same type. If the transaction contains LS and LE segments and depending on whether your map is inbound or outbound, you will need to define the LS and LE segments for the loops you will be using in the map in one of two different ways.

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

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

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

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

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

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

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

outbound mapping The system translates your application file format to EDI standard formats, so you can send documents to your partners. To translate outbound data, you need to create an import map and a system import map in Gentran:Director.

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

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

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

positional file See application file.

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

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

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

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

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

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

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

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

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

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

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

responsible agency An organization that develops and updates standards for EDI communications. These organizations include ANSI, EDIA, AIAG, UCS, EDIFACT, ODETTE, and VICS.

segment A group of related elements or composite data elements that combine to communicate useful data. Segments are defined by the EDI standards. A segment can occur once or can repeat multiple times. For example, a catalog price segment might consist of elements for item description, volume, price, lead time, etc. By themselves, none of these elements would communicate useful information. Together, they provide the information necessary for someone to tell if the item is what is desired, whether the price is reasonable, etc. A number of segments together form a transaction set.

select This standard rule enables you to select entries from a location table, cross-reference table, partner table, or lookup table (all tables created in the Gentran:Director Partner Editor). You can then map the fields in those tables to one or more fields in the data. The Select function uses the value of the current field to perform the selection.

send This Gentran:Director function enables you to manually start a communications session to send data to your trading partner. Only selected documents are enveloped and sent. If no documents are selected, ALL documents will be sent. Successfully sent documents are moved to the Out Drawer.

set See Transaction Set.

simple mapping See link.

split This function enables you to split (break) a group or repeating segment into two loops. You will typically use this function when you need more than one instance of the same map component that still occurs multiple times.

standard rules These rules give you access to functions that are necessary for mapping operations that are more complex than simple linking, but less involved than extended rules. Each of the standard rules are mutually exclusive (you can use only one on a particular field).

status bar The status bar of an application window defines information about a selection, command, or process, defines Menu Bar items as the user highlights each item, and indicates any current keyboard-initiated modes for typing (e.g., CAP for the “Caps Lock” key or NUM for the “Num Lock” key, etc.).

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

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

system import map The translator uses this map to determine which trading relationship (established in Partner Editor) corresponds to each document in the application file, so the system will know which import map to use to process the document. You need a system import map that is defined in EDIMGR.INI to translate outbound data.

system import translation object See system import map.

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

TDCC See EDIA.

TDF TDF is the Transaction Data File. This file serves as a filter between your document files and the Gentran:Director 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:Director can be exported to a TDF-formatted file using a TDF Export translation object. This data file is then ready to be processed or converted to your internal application files.

third-party network See Network.

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

toolbar This contains buttons that graphically represent commands available in the Menu Bar.

TRADACOMS The U.K. standards for EDI that are published by the Article Numbering Association (UK) LTD.

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

transaction set (document) A business form as defined by the standards. Examples include an ANSI 850 purchase order or an EDIFACT INVOIC (invoice). The standards define each transaction set in terms of the segments and elements that make up the form, the order in which they appear, and the relationships among them. This is also known as “message” in Europe.

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

Inbound Translation Objects:

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

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

Print: This translation object is used to print documents.

Outbound Translation Objects:

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

Print: This translation object is used to print documents.

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

translator This is the engine that processes data for Gentran:Director, the Application Integration subsystem, and the Forms Integration subsystem.

transmission See Communications Session.

transmission chain A path an EDI communication could follow, including one company, one trading partner, and one or more network services.

turnaround document A document into which data elements from the source document have been automatically transferred using a turnaround map.

turnaround map A series of instructions that the system uses to create a turnaround document (a logical response document to the source) from an inbound (source) document, by transferring data from the source document to elements in the target document (translation object).

UCS The Uniform Communications Standard is the standard used by the grocery industry.

update This standard rule enables you to update a specific field in a document record, envelope segment, interchange, group, current partner, or document (if the map side format type is EDI), with the element or field contents.

use accum This standard rule gives you access to a set of numeric variables that you can manipulate via numeric operations, and then transfer to and from fields. This function enables you to add, change, or delete calculations for the element/field, including hash totals (used to accumulate numeric field values, i.e., quantity, price, etc.). This function also enables you to map the accumulated total into a control total field, and use accumulators. Accumulators are used generally for counting the occurrences of a specific element or generating increasing or sequential record or line item numbers.

use code This standard rule enables you to match an element or field against a predefined code table, specify whether or not a compliance error will be generated if the element or field does not contain one of the values in the code table, and store a code’s description in another element or field.

user translation object These translation objects control data entry, importing, exporting, document turnaround, and creating printed reports. These translation objects are created using the Application Integration or Forms Integration subsystems. The Application Integration subsystem enables you to generate import, export, and document turnaround translation objects. The Forms Integration subsystem enables you to generate data entry and print translation objects.

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

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

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

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